Doctoral thesis submitted to
the Faculty of Behavioural and Cultural Studies
Heidelberg University
in partial fulfillment of the requirements of the degree of
Doctor of Philosophy (Dr. phil.)
in Psychology

Title of the publication-based thesis

AI for Cognitive Science: Scaling Bayesian Modeling with Deep Learning

presented by
Lasse Elsemüller, M.Sc.

year of submission
2025

Dean: Prof. Dr. Guido Sprenger
Supervisors: Prof. Dr. Andreas Voß
Asst. Prof. Dr. Stefan T. Radev
Prof. Dr. Andrea Kiesel

Acknowledgments

I am deeply grateful for the privilege of taking on this PhD journey and for the enduring support I received along the way. My first steps into the academic world occurred during my Bachelor's studies at the University of Mannheim, as a student assistant at Edgar Erdfelder's lab. I would like to thank Edgar for constantly encouraging me to consider pursuing a PhD. He never missed an opportunity—even during our meetings for my bachelor's thesis project—to kindly nudge me towards this path. Martin Schnuerch also had a decisive influence on my decision to take on this PhD journey: When discussing potential Master's thesis topics at the intersection of psychology and machine learning, he quickly connected me with Stefan Radev, which turned into a joint project of Edgar, Martin, Stefan, and me. The experiences made during this project fueled my excitement for research at the frontier of methodological advancements. They also convinced me that modern scientific research can and should be approached as a collaborative endeavor where the whole is much greater—and much more fun—than the sum of its parts. I would like to thank Edgar again for continuing his tireless encouragement until I finally started my PhD position at the *Statistical Modeling in Psychology* (SMiP) research training group, and also for encouraging me to consider Heidelberg University as soon as it became apparent that my research interests would be perfectly situated in Andreas Voss' lab.

This brings me to the supervisors of this thesis: First, I would like to thank Andreas Voss for his immediate support when I first expressed my interest in joining his lab, his encouragement in pursuing my methodological research interests, and for always taking the time to share his expertise in evidence accumulation modeling with me. Second, I am also deeply grateful to Stefan Radev for his invaluable guidance in navigating the complexities of deep learning, Bayesian inference, and academic writing during our close collaboration ever since my Master's thesis. Third, I would like to thank Andrea Kiesel, whose impressively broad scope of psychological research interests always sparked inspiring discussions about the evolving relationship between machine learning and psychology.

I quickly learned that the conditions for conducting PhD research can vary greatly, and thus consider myself lucky to have been part of a rich and supportive academic ecosystem. Specifically, I would like to express my gratitude for the great discussions about: evidence accumulation modeling and the everyday challenges of academic research at our lab at Heidelberg University; amortized Bayesian inference, deep learning, and software engineering at the BayesFlow developer team; and computational modeling

in psychology and the broader PhD experience at the SMiP research training group. Special thanks go out to Marvin Schmitt and Lukas Schumacher for the countless hours of enthusiastically nerding out together.

One of the highlights of my PhD journey was my lab visit at the University of Amsterdam. Here, I would like to thank Andrew Heathcote, Dora Matzke, and the whole AMPL team—especially Henrik Godmann and Michelle Donzallaz—for their warm welcome and the many inspiring discussions. I am grateful to the SMiP research training group for making this fantastic experience, as well as many others, possible.

On a personal note, I feel truly privileged by my family's unconditional belief in me and the immense support I received whenever needed. My gratitude extends to my friends in Mannheim and everywhere else, who put the inevitable lows of a PhD journey into perspective and reminded me of celebrating the milestones along the way. Finally, I am deeply grateful to always have Hannah by my side on this journey, from its very first steps nine years ago. I am excited about all the shared journeys yet to come.

Contents

Introduction 1

To attain any assured knowledge about the soul is one of the most difficult things in the world.

—Aristotle, *De Anima*

Reasoning about human cognition—the processes enabling such reasoning in the first place—might be one of the oldest scientific endeavors in human history. Yet, approaching this challenging undertaking with the rigor of a formal, empirically grounded science is a relatively recent development. In modern cognitive science, *cognitive modeling* allows us to move beyond purely theoretical contemplation: By developing formal computational models, we can encode our theoretical understanding of processes such as learning, memory, and decision making. The precise mathematical formulation of these models allows researchers to empirically test theoretical explanations for observed behavior. For example, cognitive models of decision making decompose an individual's observed decision behavior into several cognitive parameters, such as their speed of information uptake or the caution with which they commit to a decision. Through the process of statistical inference, such latent parameters can be estimated based on observed behavior. Thereby, seemingly simple behavioral outcomes of a complex interplay of cognitive processes can inform our theoretical understanding of the latter.

However, this powerful approach faces a fundamental limitation that creates a growing bottleneck for scientific progress. If there is one recurring theme in millennia of reasoning about cognition, it is that even our most sophisticated theories barely scratch the surface of capturing the inherent complexity of the human mind. Thus, as our scientific theories evolve to capture this complexity, so too must our computational models. With growing model complexity, in turn, statistical inference becomes computationally more demanding, often involving intractable high-dimensional integrals. In practice, this forces researchers into a difficult compromise: either spend weeks on computation or, more commonly, retreat to simplified models that fail to express their full theoretical understanding.

An emerging paradigm offers a path forward: *AI for science* seeks to leverage recent breakthroughs in artificial intelligence (AI) research for pushing the frontiers of complexity and scale in scientific discovery (Berens et al., 2023). Highly visible examples of this progress include fundamental advances in protein structure prediction (Abramson et al.,

2024; Jumper et al., 2021), exoplanet detection (Shallue & Vanderburg, 2018), or global weather forecasting (Lam et al., 2023; Price et al., 2025). At the heart of these successes lies the ability of deep neural networks to extract complex patterns from large data sets, enabling the approximation of complex processes.

Within this new paradigm, a recent methodological advancement is particularly promising for overcoming computational bottlenecks in cognitive modeling: *Amortized Bayesian inference* (ABI) employs deep neural networks to accelerate Bayesian inference, a comprehensive framework for statistical inference. In essence, a neural network is trained on vast amounts of simulated data to learn a direct mapping from a given data set to the underlying parameters of a scientific model. Compared to traditional inference methods, ABI offers two unique properties that promise to substantially scale cognitive modeling and expand its possibilities: First, ABI's simulation-based nature bypasses the computationally prohibitive calculations that render complex models mathematically intractable. This enables cognitive modelers to formalize their theoretical understanding without concerns about computational feasibility. Second, traditional inference methods can take hours for processing a single data set (e.g., experimental data of a single participant). This poses a challenge in handling the ever-increasing amounts of data available. In contrast, ABI splits statistical inference into an upfront training phase and subsequent near-real-time inference. This amortizes the initial training cost over multiple analyzed data sets, unlocking the application of complex statistical models to large-scale data of potentially millions of individuals. Thus, in line with the aims of the broader AI for science paradigm, ABI has the potential to advance both the complexity and scale of cognitive modeling.

Realizing this potential, however, requires careful adaptation and rigorous evaluation: In contrast to traditional statistical methods with decades of research on their theoretical guarantees and application-specific properties, ABI is a young and rapidly evolving field. Consequently, readily available ABI-based solutions for common cognitive modeling scenarios might not yet exist, their reliability might not be clear, and their implementation might demand specialized expertise in deep learning and Bayesian inference. This thesis aims to scale cognitive modeling with ABI by addressing three key requirements: (i) Expanding ABI's *capabilities* to fully cover relevant use cases in cognitive modeling, (ii) ensuring *trustworthiness* when applying these emerging deep learning methods in complex applied modeling workflows, and (iii) increasing the *accessibility* for practitioners.

The interdisciplinary nature of this work opens up new perspectives and resources for approaching the obstacles encountered along the way. For example, model misspecification—the unavoidable mismatch between a simplified statistical model and the complexity of the real world—is a fundamental problem in statistical modeling. In the context of ABI, this challenge manifests as a "sim-to-real" transfer problem: the neural network is trained on

clean, simulated data, but must then make accurate predictions for noisy, empirical data. At the same time, as demonstrated throughout this thesis, ABI's interdisciplinary foundation lets us draw on a rich body of AI research that investigates this domain transfer problem in other learning tasks, such as mitigating the impact of unfamiliar environments on autonomous driving systems. Such adaptations of powerful general-purpose advancements for ABI, as well as their critical evaluation in the context of cognitive modeling, represent a fundamental motivation for this thesis that is reflected in all of its publications.

1.1 Outline

Chapter 2 introduces the theoretical foundations that my publications rest upon. Concretely, it provides an overview of the four fields that form the intersection in which my work is located: (i) Modern computational cognitive modeling as a formal approach to capturing the complexity of human cognition, (ii) Bayesian inference as a principled statistical framework for specifying and updating our knowledge about such formal cognitive models, (iii) simulation-based inference as a collection of inference methods for complex models that cannot be tamed by traditional statistical methods, and (iv) ABI as a new frontier of probabilistic modeling that leverages advances in deep learning to substantially accelerate inference for complex models.

Afterwards, the core publications of this thesis are summarized and connected: Chapter 3 develops a novel amortized method for comparing Bayesian hierarchical models and validates it on widely used cognitive models. Chapter 4 proposes a workflow that leverages the unique properties of amortized inference to enable large-scale sensitivity analyses for complex models. Chapter 5 further explores the fundamental issue of robustness by systematically evaluating the potential and limitations of unsupervised domain adaptation for bridging the gap between the simulated and the real world in ABI. As a cumulative dissertation, elements (e.g., figures) of these publications may be incorporated directly without explicit reference. Finally, Chapter 6 broadens the scope with a general discussion of overarching themes, current limitations, and future perspectives for the field.

1.2 Scientific Publications of the Cumulative Dissertation

The core of this dissertation consists of three first-author publications, of which two are published and one is currently under review:

- **Elsemüller, L.**, Schnuerch, M., Bürkner, P. C., & Radev, S. T. (2024). A deep learning method for comparing Bayesian hierarchical models. *Psychological Methods*. Advance online publication. https://doi.org/10.1037/met0000645
  Awarded the *SMiP Best Paper Award 2023 / 2024* by the research training group *Statistical Modeling in Psychology* (SMiP).

- **Elsemüller, L.**, Olischläger, H., Schmitt, M., Bürkner, P. C., Köthe, U., & Radev, S. T. (2024). Sensitivity-aware amortized Bayesian inference. *Transactions on Machine Learning Research*.
  Presented at the *International Conference on Learning Representations* (ICLR) 2025.

- **Elsemüller, L.**, Pratz, V., von Krause, M., Voss, A., Bürkner, P. C., & Radev, S. T. (2025). Does unsupervised domain adaptation improve the robustness of amortized Bayesian inference? A systematic evaluation. Under review at *Transactions on Machine Learning Research*.
  An early version of this work was published at the *Frontiers in Probabilistic Inference: Learning Meets Sampling* workshop at the *International Conference on Learning Representations* (ICLR) 2025.[1]

I additionally contributed to the following publications that are closely related to the contents of this dissertation:

- Radev, S. T., Schmitt, M., Schumacher, L., **Elsemüller, L.**, Pratz, V., Schälte, Y., Köthe, U., & Bürkner, P. C. (2023). BayesFlow: Amortized Bayesian workflows with neural networks. *Journal of Open Source Software 8*(89), 5702. https://doi.org/10.21105/joss.05702

- Liss, J. V., von Krause, M., **Elsemüller, L.**, Hunsmann, E. M., & Lerche, V. (2025). Time to jump: Exploring the distribution of noise in evidence accumulation as a function of time pressure. Under review at *Cognitive Psychology*.

1.3 Acronyms & Notation

The fields of cognitive science, Bayesian inference, and deep learning share a high prevalence of acronyms. Thus, Table 1.1 aims to mitigate potential ambiguity for the reader by providing a reference for the most important acronyms used throughout this thesis. Additionally, Table 1.2 provides an overview of the core notation employed in this thesis.

[1] In machine learning, conference workshops provide a space to discuss ongoing work. Publications in these workshops are non-archival, i.e., they do not prevent subsequent publication of the final version.

Table 1.1: List of central acronyms.

Acronym	Definition	First Mention
AI	Artificial Intelligence	Chapter 1
ABI	Amortized Bayesian Inference	Chapter 1
EAM	Evidence Accumulation Model	Section 2.1
MCMC	Markov Chain Monte Carlo	Section 2.2
ABC	Approximate Bayesian Computation	Section 2.3
NPE	Neural Posterior Estimation	Section 2.4
UDA	Unsupervised Domain Adaptation	Chapter 5
MMD	Maximum Mean Discrepancy	Chapter 5
DANN	Domain-Adversarial Neural Networks	Chapter 5

Table 1.2: Overview of the core notation used throughout this thesis.

	Symbol	Meaning
General	a	A scalar
	$\boldsymbol{a}$	A vector
	$\{\boldsymbol{a}\}$	A set
	$p(\cdot)$	A probability distribution (density or mass function)
	$a \sim p(a)$	Random variable a has probability distribution p
	$\mathbb{E}_{p(\cdot)}[\cdot]$	Expectation with respect to distribution $p(\cdot)$
	$\mathrm{d}a$	An infinitesimal change in a variable a
	m, n, j	Indices for groups, observations, and models, respectively
	M, N, J	Total number of groups, observations, and models, respectively
	$1, 2, \ldots, M$	Integers between 1 and M.
Bayesian Inference	$\boldsymbol{\theta}$	A vector of model parameters
	$\boldsymbol{\eta}$	A vector of group-level (hierarchical) parameters
	$\boldsymbol{x}$	A vector of data
	$\boldsymbol{x}_{\text{obs}}$	A vector of observed (empirical) data
	$p(\boldsymbol{\theta})$	Prior distribution over parameters
	$p(\boldsymbol{x} \mid \boldsymbol{\theta})$	Likelihood function of data given parameters
	$p(\boldsymbol{\theta} \mid \boldsymbol{x})$	Posterior distribution of parameters given data
	$p(\boldsymbol{\theta}, \boldsymbol{x})$	Joint distribution of parameters and data
	$p(\boldsymbol{x})$	Marginal likelihood
	$\mathcal{M}_j$	The j-th model under consideration
	$p(\mathcal{M}_j \mid \boldsymbol{x})$	Posterior probability of model $\mathcal{M}_j$ given data
Deep Learning	$q_{\boldsymbol{\phi}}(\cdot)$	Inference network parameterized by $\boldsymbol{\phi}$
	$\boldsymbol{\phi}$	Learnable parameters (e.g., weights) of the inference network
	$h_{\boldsymbol{\psi}}(\cdot)$	Summary network parameterized by $\boldsymbol{\psi}$
	$\boldsymbol{\psi}$	Learnable parameters (e.g., weights) of the summary network
	$\mathcal{L}$	A loss function serving as the training optimization objective

Theoretical Foundations 2

Over the past decades, progress in formalizing human cognition has been closely tied to the boundaries of computational feasibility. This chapter provides an overview of cognitive modeling, its computational challenges, and the methods seeking to address these challenges.

2.1 Modeling Cognition

Cognitive science seeks to uncover and understand the unobservable processes that give rise to observable behavior. Historically, this knowledge has solely been encapsulated in verbal theories that qualitatively describe cognitive mechanisms, such as decision making, attention, or perception. For example, a verbal theory of decision making might suppose that we continually gather information about the given decision options until we are confident enough to make a decision (Ratcliff, 1978). Verbal theories form the foundation of cognitive science, but they ultimately aim to explain observed behavior. Thus, to test the validity of a cognitive theory in the real world, a procedure for obtaining precise, quantitative predictions about observed behavior is needed.

Cognitive modeling bridges this gap by casting verbal theories into mathematical formulations (Farrell & Lewandowsky, 2018). This precisely specifies the interplay of latent processes that lead to observed behavior, enabling empirical testing of specific hypotheses. However, this endeavor is inherently challenging since we cannot directly measure cognition: An observed behavior only reflects the final outcome of multiple potentially complex and intertwined cognitive processes.

Conceptually, it is intuitive to think about a cognitive model in terms of the *forward* direction, which maps the flow of latent cognitive processes, governed by a set of parameters $\boldsymbol{\theta}$, into observed behavior $\boldsymbol{x}$ (see Figure 2.1, which will serve as a guiding example throughout this section). The core of cognitive modeling, however, lies in the *inverse* direction: using modern computational methods to infer the parameters given observed data. This inverse problem is mathematically challenging, as different combinations of latent processes can often lead to similar observed behaviors. For instance, a correct response in a memory task could stem from genuine recollection, a lucky guess, or a

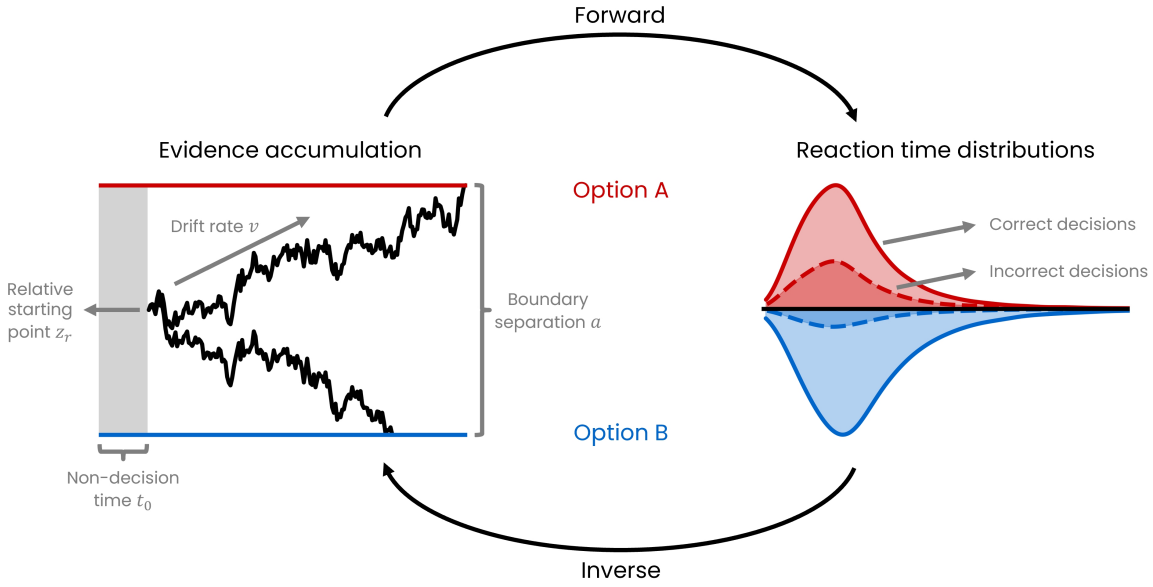
combination of both. *Statistical inference* provides the mathematical toolkit to tackle this ambiguity: It lets us estimate the most probable parameter configuration of a statistical model given the observed data and quantify the uncertainty associated with this estimate. Nevertheless, the computational hardships of this inferential process for complex models will be a recurring issue throughout this thesis. In this way, progress in cognitive modeling is directly linked to advancements in computational efficiency.

The computational methods developed in this thesis are broadly applicable within and beyond cognitive modeling. However, this thesis will focus on a class of cognitive models whose detailed resolution frequently causes computational issues—and that thus directly benefits from computational advancements: Evidence accumulation models (EAMs) are widely employed for modeling decision making in fundamental cognitive research as well as diverse applications (Boag et al., 2025), such as assessing goalkeeper anticipation in handball (Weinberg et al., 2025), performance in air-traffic control (Neal & Kwantes, 2009), entertainment media selection processes (Gong et al., 2023), or psychopathology (Sripada & Weigard, 2021). EAMs aim to explain simple decisions, such as whether a collection of letters represents a word or not, by a continuous and noisy accumulation of evidence until a decision boundary is reached (see Figure 2.1). This decomposes a set of observations consisting of decisions and their associated reaction times into a set of cognitively meaningful latent parameters. For instance, EAM analyses suggest that the age-related slowing of reaction times throughout adulthood is not primarily driven by a decrease in mental speed, but rather increases in decision caution and non-decisional processes like motor speed (Theisen et al., 2021; von Krause et al., 2022).

To render the mathematical processes with which computational models and especially EAMs explain observed behavior more tangible, let us consider a concrete example: The drift-diffusion model of decision making (Ratcliff, 1978), the most prominent EAM for modeling binary decision tasks. In its basic variant shown in Figure 2.1, the drift-diffusion model proposes four parameters: (i) drift rate v, reflecting mental speed via the average speed of evidence accumulation, (ii) boundary separation a, reflecting decision caution via the amount of evidence required to commit to a decision, (iii) relative starting point z_r, reflecting an a priori bias towards one of the decision options, and (iv) non-decision time t_0, reflecting external processes such as sensory encoding and motor execution. The core of the drift-diffusion model consists of modeling the change in evidence accumulation $\mathrm{d}x(t)$ via a diffusion process defined by the following stochastic differential equation (Smith, 2000):

$$\mathrm{d}x(t) = v\,\mathrm{d}t + \sigma\,\mathrm{d}w(t), \tag{2.1}$$

where t denotes decision time and $w(t)$ denotes a Wiener process that represents stochastic



(a) Drift-diffusion model. (b) Statistical paths. (c) Observed decision data.

Figure 2.1: Forward and inverse paths in statistical models as illustrated by evidence accumulation modeling: (a) The computational model, specifically the latent evidence accumulation process of the drift-diffusion model, for two example trials. (b) The statistical paths of forward simulation and inverse inference, linking the latent and manifest worlds. (c) The behavioral data that is modeled, specifically the manifest reaction time distributions across many trials per decision option.

Gaussian noise in evidence accumulation, scaled by the diffusion coefficient σ (typically fixed for identifiability, e.g., $\sigma = 1$). Evidence accumulation starts at $x(0) = a\, z_r$, determined by the relative starting point z_r between the two decision boundaries 0 and a, and proceeds until one of the decision boundaries is reached. Then, a decision for the respective decision option is triggered, leading to a binary decision d and its decision time t_d:

$$d = \begin{cases} 1, & \text{if } x(t) \geq a \\ 0, & \text{if } x(t) \leq 0 \end{cases} \tag{2.2}$$

$$t_d = \inf \left\{ t \geq 0 : x(t) \geq a \text{ or } x(t) \leq 0 \right\}. \tag{2.3}$$

The final reaction time t_f is obtained by including external processes via the non-decision time constant t_0, $t_f = t_d + t_0$.

Numerous EAM variants exist, ranging from extensions of the drift-diffusion model (e.g., allowing parameters to vary between experimental trials; Ratcliff & Rouder, 1998; Ratcliff & Tuerlinckx, 2002) to alternative EAM formulations (e.g., proposing multiple accumulators instead of a single one; LaBerge, 1962; Tillman et al., 2020). A particularly

interesting variant that is investigated in multiple publications of this thesis is the Lévy flight model of decision making (Voss et al., 2019; Wieschen et al., 2020). It extends the drift-diffusion model by replacing the assumption of Gaussian evidence accumulation noise with a more flexible Lévy alpha-stable distribution. The heavy tails of this noise distribution allow for abrupt changes in evidence accumulation that could reflect "jumping to conclusion" decision-making behavior (McKay et al., 2006; Voss et al., 2019).

In summary, EAMs offer a fine-grained and neurally plausible (Pereira et al., 2021; Purcell & Palmeri, 2017; Shadlen & Kiani, 2013) decomposition of observed decision-making behavior into underlying cognitive parameters. Nevertheless, as I will lay out in the following section, the very richness that makes EAMs powerful also renders them computationally intensive, which has historically constrained modelers to computationally tractable specifications (Evans & Wagenmakers, 2020).

2.2 Bayesian Inference

Bayesian inference directly operationalizes the process of scientific discovery: updating knowledge in light of new evidence. This knowledge is captured by probability distributions, which enables an intuitive encoding of prior knowledge and a principled propagation of uncertainty throughout a Bayesian analysis. Both properties are valuable for working with increasingly complex cognitive models: For example, prior knowledge about expected parameter ranges can substantially constrain the search space during model fitting, while uncertainty-related questions, such as the probability of a parameter exceeding a specific threshold, can be answered directly. As this section will illustrate, the theoretical appeal of Bayesian inference comes with a number of computational obstacles. Concurrently with the continuous reduction of these obstacles, Bayesian methods are becoming an increasingly central part of psychological science (Van De Schoot et al., 2017).

Specifically, *Bayesian parameter estimation* starts with specifying a prior distribution $p(\boldsymbol{\theta})$ that encapsulates knowledge about the model parameters $\boldsymbol{\theta}$ and a likelihood $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ that expresses the assumed stochastic relationship between model parameters $\boldsymbol{\theta}$ and observed data $\boldsymbol{x}$:

$$\boldsymbol{\theta} \sim p(\boldsymbol{\theta}), \quad \boldsymbol{x} \sim p(\boldsymbol{x} \mid \boldsymbol{\theta}). \tag{2.4}$$

This generative recipe formalizes the forward direction of the probabilistic model. As a concrete EAM example, $p(\boldsymbol{\theta})$ captures substantive knowledge such as the typical time taken for non-decisional processes t_0, whereas $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ specifies the stochastic evidence

accumulation process. The inverse direction of obtaining the posterior distribution $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ representing the updated belief about the model parameters is, in turn, defined by Bayes' theorem:

$$p(\boldsymbol{\theta} \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{p(\boldsymbol{x})} = \frac{p(\boldsymbol{x} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{\int p(\boldsymbol{x} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta}}. \tag{2.5}$$

For most models of interest, the integral defining the marginal likelihood $p(\boldsymbol{x})$ is analytically intractable and hopeless to approximate numerically with sufficient precision. This has motivated the development of a wide range of approximate methods for Bayesian inference. For instance, the current gold-standard method for Bayesian parameter estimation, Markov chain Monte Carlo (MCMC), bypasses the calculation of the normalizing constant $p(\boldsymbol{x})$. Instead, MCMC approximates $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ with samples from the unnormalized posterior, leveraging the proportionality $p(\boldsymbol{\theta} \mid \boldsymbol{x}) \propto p(\boldsymbol{x} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$.

A further advantage of the Bayesian approach is its seamless integration of known hierarchies, such as nested observations within participants. *Bayesian hierarchical models* introduce group-level parameters $\boldsymbol{\eta}$ that are shared between individuals. For a two-level hierarchical model, the generative recipe of an individual m is extended as follows:

$$\boldsymbol{\eta} \sim p(\boldsymbol{\eta}), \quad \boldsymbol{\theta}_m \sim p(\boldsymbol{\theta} \mid \boldsymbol{\eta}), \quad \boldsymbol{x}_m \sim p(\boldsymbol{x} \mid \boldsymbol{\theta}_m). \tag{2.6}$$

The shared information from estimating individual-level parameters $\boldsymbol{\theta}_m$ and group-level parameters $\boldsymbol{\eta}$ simultaneously can considerably improve estimation precision. This enables accurate inference in situations with little individual information available, a frequent challenge in applied psychological settings (e.g., Weigard et al., 2023), as well as principled modeling of individual differences (Haaf & Rouder, 2019). The flexibility of hierarchical models comes at the cost of increased model complexity (i.e., potentially high-dimensional parameter spaces), which can complicate model specification and introduce further computational challenges (e.g., complex posterior geometries; Betancourt & Girolami, 2015).

Beyond parameter estimation within a single model, Bayesian inference also offers a natural framework for comparing multiple models (e.g., of competing cognitive theories). We can delineate the different models in the set of compared models $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_J\}$ by explicitly introducing the previously implicit dependence on a model $\mathcal{M}_j$:

$$\mathcal{M}_j \sim p(\mathcal{M}), \quad \boldsymbol{\theta} \sim p(\boldsymbol{\theta} \mid \mathcal{M}_j), \quad \boldsymbol{x} \sim p(\boldsymbol{x} \mid \boldsymbol{\theta}, \mathcal{M}_j), \tag{2.7}$$

where the model prior $p(\mathcal{M})$ encodes the prior belief about the probabilities of all competing models (usually assuming equal prior probabilities). In *Bayesian model comparison*, the marginal likelihood first encountered in Equation 2.5 becomes the central quantity of

interest:

$$p(\boldsymbol{x} \mid \mathcal{M}_j) = \int p(\boldsymbol{x} \mid \boldsymbol{\theta}, \mathcal{M}_j)\, p(\boldsymbol{\theta} \mid \mathcal{M}_j)\, \mathrm{d}\boldsymbol{\theta}. \tag{2.8}$$

Integrating the likelihood over the prior domain automatically penalizes overly complex models (MacKay, 2003): Compared to a simpler model that encodes more specific prior predictions before encountering the data, a more complex model will spread its prior predictions over a wider range of possibilities. However, in Equation 2.8, only those regions of the parameter space that fit the data well and are likely under the prior will contribute meaningfully to the integral. That means that if both models predict the data well, the simpler model will be favored over the complex model with diffuse prior predictions. The reliance on a model's prior predictive behavior is not undisputed (Gelman & Yao, 2020) but is often regarded as beneficial in psychological research (Heck et al., 2023). This is based on the perspective that all parts of the model, including the prior, should encode a scientific theory (Haaf et al., 2025; Vanpaemel, 2010). Prior sensitivity analyses reveal this disputed influence on inferential results (Berger, 1990), and Chapter 4 develops an efficient method to substantially alleviate their computational burden.

The actual model comparison takes place when considering multiple models from $\mathcal{M}$, either by contrasting the relative evidence for two models $\mathcal{M}_i$ and $\mathcal{M}_j$ via the Bayes Factor

$$\mathrm{BF}_{ij} = \frac{p\left(\boldsymbol{x} \mid \mathcal{M}_i\right)}{p\left(\boldsymbol{x} \mid \mathcal{M}_j\right)}, \tag{2.9}$$

or by updating the prior belief $p(\mathcal{M})$ to arrive at posterior model probabilities

$$p\left(\mathcal{M}_j \mid \boldsymbol{x}\right) = \frac{p(\boldsymbol{x} \mid \mathcal{M}_j)\, p(\mathcal{M}_j)}{\sum_{i=1}^{J} p\left(\boldsymbol{x} \mid \mathcal{M}_i\right)\, p(\mathcal{M}_i)}. \tag{2.10}$$

While Bayesian model comparison offers an elegant and comprehensive way to compare competing models, relying on the marginal likelihood inherits all its associated computational challenges. Chapter 3 proposes a method for handling the particularly challenging combination of Bayesian model comparison and hierarchical models.

All of these computational challenges are further aggravated when a critical condition is not fulfilled, namely, that a model's likelihood function $p(\boldsymbol{x} \mid \boldsymbol{\theta})$, which has to be evaluated many times during model fitting, is explicitly available. A likelihood function is explicitly available whenever its distributional family is known analytically and its density can be evaluated for any pair $(\boldsymbol{x}, \boldsymbol{\theta})$ in a reasonable time via a closed-form analytical solution or efficient numerical methods (Diggle & Gratton, 1984; Marin et al., 2012). However, a growing class of sophisticated models in cognitive science does not permit this. In such implicit likelihood models (Diggle & Gratton, 1984), the generative model is defined via a

Monte Carlo simulation program G,

$$\boldsymbol{x} = G(\boldsymbol{\theta}, \boldsymbol{\xi}) \quad \text{with} \quad \boldsymbol{\theta} \sim p(\boldsymbol{\theta}), \quad \boldsymbol{\xi} \sim p(\boldsymbol{\xi} \mid \boldsymbol{\theta}), \tag{2.11}$$

where $\boldsymbol{\xi}$ denotes latent program states of the simulation program with density function $p(\boldsymbol{\xi} \mid \boldsymbol{\theta})$. Executing the simulation program G is equivalent to sampling from an implicitly defined likelihood (Cranmer et al., 2020):

$$p(\boldsymbol{x} \mid \boldsymbol{\theta}) = \int p(\boldsymbol{x}, \boldsymbol{\xi} \mid \boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\xi}. \tag{2.12}$$

While obtaining single simulations is feasible, evaluating such an implicit likelihood by integrating across all possible program states $\boldsymbol{\xi}$ of the simulation program is typically intractable. This renders standard algorithms for Bayesian inference, such as MCMC, inapplicable and marginal-likelihood-based model comparison even doubly intractable.

Implicit likelihood models are increasingly used for capturing the complexity of natural phenomena across scientific fields and scales, such as particle physics, cognitive (neuro)science, epidemiology, or cosmology (Cranmer et al., 2020; Zammit-Mangion et al., 2025). Considering EAMs, the likelihood of a decision and its reaction time is explicitly available for the drift-diffusion model, usually via a combination of the analytic Wiener first-passage-time density and numerical approximations (Henrich et al., 2024; Navarro & Fuss, 2009; Ratcliff & Tuerlinckx, 2002; Voss & Voss, 2008; Wiecki et al., 2013). Nevertheless, the unavailability of tractable likelihood functions for many theoretically appealing EAM variants is a major reason for the computational hardships discussed in Section 2.1. Prominent examples include the Lévy flight model introduced in Section 2.1, the integration of evidence leakage or inhibition between accumulation processes (Usher & McClelland, 2001), or modeling within-trial caution decreases with non-linear collapsing bounds (Hawkins et al., 2015; Palestro et al., 2018). Until recently, likelihood intractability represented a major obstacle on the path towards fine-grained models that adequately capture the complexity of human cognition. The following sections will introduce simulation-based approaches for resolving this obstacle.

2.3 Simulation-Based Inference

The field of *simulation-based inference* enables inference for implicit likelihood models. Historically, simulation-based inference mainly referred to *approximate Bayesian computation* (ABC; Beaumont et al., 2002; Pritchard et al., 1999) methods (Cranmer et al., 2020).

These methods replace the intractable likelihood evaluation with a comparison between simulated data $\boldsymbol{x}$ and observed data $\boldsymbol{x}_{\text{obs}}$.

In its naïve rejection-based form, the ABC algorithm approximates a parameter posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{obs}})$ by accepting proposals $\boldsymbol{\theta}^* \sim p(\boldsymbol{\theta})$ if the distance of a resulting synthetic data set $\boldsymbol{x} \sim p(\boldsymbol{x} \mid \boldsymbol{\theta}^*)$ to the observed data $\boldsymbol{x}_{\text{obs}}$ falls below a pre-defined threshold. The collection of accepted proposals then approximates the target posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{obs}})$. For model comparison, the procedure is extended by first sampling a proposal model $\mathcal{M}_j^* \sim p(\mathcal{M})$ and approximating the posterior model probability $p(\mathcal{M}_j \mid \boldsymbol{x})$ by the fraction of accepted samples generated from model $\mathcal{M}_j$.

While powerful in enabling inference for implicit likelihood models, most ABC methods come with a number of challenges that have prevented their widespread adoption in cognitive modeling: First, the *curse of dimensionality* (i.e., an exponential decrease of relative volume leading to vanishing acceptance probabilities) requires the compression of high-dimensional data with typically manually generated summary statistics (Sisson et al., 2018). This solution, in turn, introduces instead a *curse of insufficiency* through the information loss associated with the compression (Marin et al., 2018; Robert et al., 2011). Second, obtaining a high-quality approximation requires a strict acceptance threshold. However, such a low-distance threshold massively increases the computational load by rejecting most proposals (Cranmer et al., 2020). State-of-the-art ABC algorithms thus often sequentially refine the proposal distribution to focus on relevant parameter regions (Beaumont, 2019; Picchini & Tamborrino, 2024; Sisson et al., 2007). This directly leads to a third challenge: The resulting dependence of the proposal distribution on $\boldsymbol{x}_{\text{obs}}$ necessitates repeating the potentially expensive computational process when inferring parameters for a new data set. This is especially relevant in cognitive modeling, which frequently requires a large number of model fits. Typical situations include validating a computational model on thousands of simulated data sets or fitting it to observed experimental data from multiple participants, sometimes even millions (von Krause et al., 2022).

2.4 Amortized Bayesian Inference with Deep Learning

Recently, the rapid speed of development in deep learning has substantially accelerated simulation-based inference. *Amortized Bayesian inference* (ABI) seeks to eliminate the need to repeat costly approximation procedures for every new data set by amortizing an initial computational investment across multiple rapidly obtained approximations (Bürkner

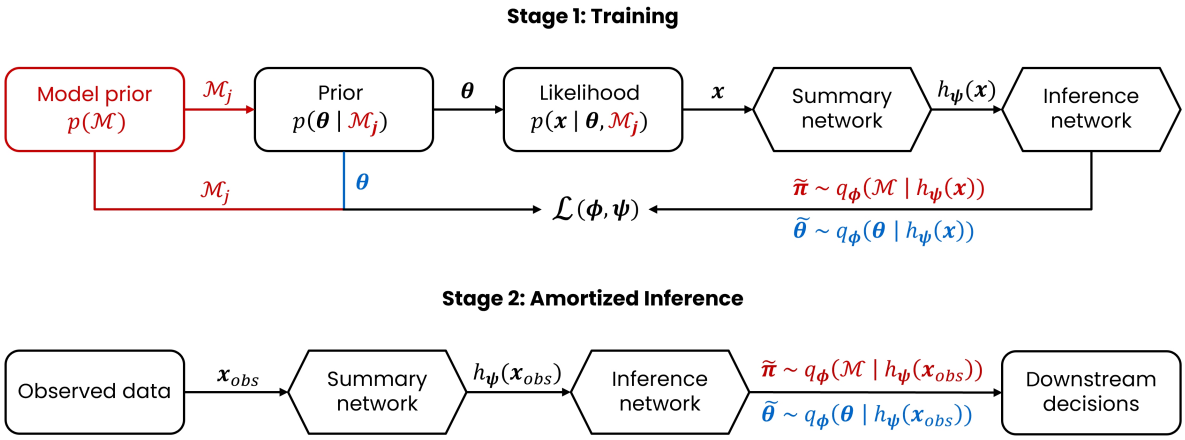


Figure 2.2: Amortized Bayesian inference with deep learning: In each step of the upfront training stage, the neural networks are updated to minimize the training loss $\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\psi})$. Afterwards, the trained networks perform amortized inference on a potentially large number of observed data sets. While most of the workflow is shared between approximation targets, unique aspects for parameter estimation and model comparison are highlighted.

et al., 2023; Radev et al., 2020; Stuhlmüller et al., 2013). This is realized by framing approximate inference as a prediction task that is learned by a function approximator based on a training set of simulated pairs $(\boldsymbol{x}, \boldsymbol{\theta})$ of data and parameters. Within this relatively young field, deep neural networks have quickly emerged as suitable function approximators for this task: As flexible and universal function approximators (Hornik et al., 1989), they particularly profit from the amount of training data only being limited by simulation cost. This leads to ABI with deep learning,[1] in which the cost of training a deep neural network for posterior inference is amortized by subsequent rapid inference performed by the trained network (see Figure 2.2 for an overview of the concepts introduced in this chapter).

Typically, a modular neural network architecture tailored for both inference task and data requirements is chosen: The inference task itself is approximated by an *inference network* q_{ϕ} parameterized by learnable neural network weights $\boldsymbol{\phi}$. For the most common task of neural posterior estimation (NPE; Papamakarios & Murray, 2016), this network directly approximates the posterior density $p(\boldsymbol{\theta} \mid \boldsymbol{x}) \approx q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid \boldsymbol{x})$ with a generative neural network architecture, such as normalizing flows (Papamakarios et al., 2021; Rezende & Mohamed, 2015), flow matching (Lipman et al., 2023; Liu et al., 2023), or diffusion

[1]Given the current inseparability of ABI and deep learning, I will follow the common convention of using "ABI" to refer to "ABI with deep learning" throughout this thesis.

models[2] (Geffner et al., 2023). Amortized Bayesian model comparison can, in turn, be realized by a standard neural network architecture, the multilayer perceptron (Rosenblatt, 1958). This is combined with a final softmax output layer that normalizes the network outputs to a categorical distribution, thus enabling the approximation of posterior model probabilities $p(\mathcal{M} \mid \boldsymbol{x}) \approx q_{\boldsymbol{\phi}}(\mathcal{M} \mid \boldsymbol{x})$ as a classification task (Radev et al., 2021). Both of these approaches to parameter estimation and model comparison amortize the training cost with rapid inference across multiple applications. This contrasts with other neural simulation-based methods that specialize inference for a specific observed data set, such as sequential NPE (Papamakarios & Murray, 2016), or require additional sampling algorithms for each observation, such as neural likelihood estimation (Papamakarios et al., 2019) or neural ratio estimation (Hermans et al., 2020).

Common data requirements for the network architecture consist of high dimensionality, unique probabilistic structures (e.g., time series), and variable input length (e.g., varying numbers of trials between participants). To address this, the inference network is usually preceded by a *summary network* $h_{\boldsymbol{\psi}}$ compressing data of variable length into fixed-size embeddings $h_{\boldsymbol{\psi}}(\boldsymbol{x})$ of learned summary statistics. This modular approach allows the summary network's architecture to be tailored to the probabilistic structure of the processed data: For example, DeepSet (Zaheer et al., 2017) or Set Transformer (Lee et al., 2019) architectures encode the permutation invariance (i.e., the invariance of the results to the ordering of data points) inherent to most psychological data, whereas time-series data can be approached with architectures specialized for sequential data, such as LSTM (Hochreiter & Schmidhuber, 1997) or transformer (Vaswani et al., 2017) architectures. Further, enabling the processing of variable-length data expands the scope of amortization from data sets with the exact same size to the realistic condition of varying sizes (e.g., due to missing data).

During training, the network weights $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$ are adjusted to minimize an optimization objective. For parameter estimation via NPE, the standard optimization objective is the negative log posterior loss, which minimizes the Kullback-Leibler (KL) divergence between the true posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ and the approximate posterior $q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}))$ (Papamakarios & Murray, 2016; Radev et al., 2020):

$$\begin{aligned} \mathcal{L}_{\text{NPE}}(\boldsymbol{\phi}, \boldsymbol{\psi}) &= \text{KL}\Big(p(\boldsymbol{\theta} \mid \boldsymbol{x}) \,||\, q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x})\Big) \\ &= \mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})}\Big[- \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}))\Big]. \end{aligned} \tag{2.13}$$

[2]This highlights two potentially confusing overlaps in terminology between the fields: In deep learning, "model" refers to the computational architecture itself and the only conceptual overlap of diffusion model architectures with cognitive drift-diffusion models is the central role of diffusion processes.

For (neural) model comparison, the categorical cross-entropy loss provides an analogous approach (Radev et al., 2021),

$$\begin{aligned}\mathcal{L}_{\text{NMC}}(\boldsymbol{\phi}, \boldsymbol{\psi}) &= \text{KL}\Big(p\left(\mathcal{M} \mid \boldsymbol{x}\right) \mid\mid q_{\boldsymbol{\phi}}(\mathcal{M} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x})\Big) \\ &= \mathbb{E}_{p(\mathcal{M}, \boldsymbol{\theta}, \boldsymbol{x})}\Big[- \sum_{j=1}^{J} \mathbb{I}_{\mathcal{M}_j} \log q_{\boldsymbol{\phi}}(\mathcal{M}_j \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}))\Big],\end{aligned} \tag{2.14}$$

where the indicator function $\mathbb{I}_{\mathcal{M}_j}$ denotes a one-hot encoding for the true model index (i.e., $\mathbb{I}_{\mathcal{M}_j} = 1$ if $\mathcal{M}_j$ is the data-generating model). For both approximation targets, we can approximate the optimization objective via its Monte Carlo estimate: Since the simulator $p(\boldsymbol{\theta}, \boldsymbol{x}) = p(\boldsymbol{x} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$ grants access to the joint distribution that the expectation $\mathbb{E}[\cdot]$ is taken over, we can generate large amounts of simulations from the joint model for minimizing the optimization objectives via stochastic gradient descent (Radev et al., 2020). In theory, Equation 2.13 and Equation 2.14 guarantee under perfect convergence that $h_{\boldsymbol{\psi}}$ outputs sufficient statistics and $q_{\boldsymbol{\phi}}$ samples from the true posterior (Radev et al., 2020, 2021). However, since perfect (asymptotic) convergence cannot be assumed in practice, extensive validation of trained networks is an essential and actively researched component of ABI workflows.

Overall, ABI addresses all three challenges of ABC outlined in Section 2.3: (i) Summary statistics are learned end-to-end to optimally retain information for posterior inference, (ii) simulations are used more efficiently (Lueckmann et al., 2021), and (iii) amortized inference enables obtaining thousands of posterior samples for a data set in a fraction of a second after training. In contrast, even gold-standard MCMC methods for explicit likelihood models are usually too slow for large data sets or real-time inference. Thus, ABI's inference speed transitions simulation-based inference from a computationally intensive last resort for implicit likelihood models to an appealing modeling method even for explicit likelihood models. Besides enabling large-scale inference, amortized inference allows for validating properties like parameter recovery on thousands of simulated data sets as a default part of the modeling workflow and real-time inference in adaptive experimental design settings (Rainforth et al., 2024) or time-critical systems (e.g., industrial applications; Heringhaus et al., 2022). Chapter 4 takes this line of research one step further and proposes a method to unlock large-scale sensitivity analyses that have previously been computationally infeasible.

While ABI emerges as a powerful method for cognitive modeling, it is situated in a rapidly developing and fast-moving field. Best practices as well as method validation workflows are still evolving, and application opportunities in cognitive science are continually being explored (see Chapter 6 for an extensive discussion of the current state). This thesis aims to realize the potential of ABI for scaling cognitive modeling. Specifically, it

develops new deep learning methods to provide missing capabilities in cognitive modeling (Chapter 3), leverages the unique advantages of amortized inference to enable large-scale sensitivity analyses (Chapter 4), and examines approaches for addressing the fundamental challenge of robustness in the presence of model misspecification (Chapter 5).

Amortized Hierarchical Model Comparison (Publication I)

3

Publication	**Link to full-text paper**

Elsemüller, L., Schnuerch, M., Bürkner, P. C., & Radev, S. T. (2024). A deep learning method for comparing Bayesian hierarchical models. *Psychological Methods*. Advance online publication. https://doi.org/10.1037/met0000645

Hierarchical models offer a principled approach to capture variability at multiple levels, and are increasingly popular in cognitive science and psychology (Rouder et al., 2017). As for their non-hierarchical counterparts, the comparison of competing models that embody different theoretical explanations is a central part of cognitive modeling workflows (Schad et al., 2023). For example, comparing different hierarchical models of individual differences can reveal whether unexpected response patterns of an individual in a cognitive task merely reflect measurement noise or fundamental qualitative differences in a psychological effect (Haaf & Rouder, 2019; Schnuerch et al., 2021). However, as discussed in Section 2.2, the increased complexity of hierarchical models aggravates computational challenges: When it comes to comparing such models, the integral defining the central quantity of Bayesian model comparison, the marginal likelihood (see Equation 2.8), can quickly grow intractably high-dimensional. Additionally, the state-of-the-art approximate method addressing this problem, *bridge sampling* (Bennett, 1976; Gronau et al., 2019; Meng & Wong, 1996), requires a large number of samples from the parameter posterior as well as an explicitly available likelihood function. Thus, even a single model with an analytically intractable likelihood function renders bridge sampling inapplicable.

In this work, we propose and systematically test a deep learning architecture that extends amortized Bayesian model comparison (Radev et al., 2021) to Bayesian hierarchical models. The central idea of our architecture lies in sequentially compressing data on each hierarchical level while respecting the respective probabilistic structures. We focus on the most common hierarchical model structure in cognitive science (Farrell & Lewandowsky, 2018; Singmann & Kellen, 2019): Two levels assuming independent and identically distributed (IID) random variables, such as trials nested in participants. IID random variables are permutation invariant, meaning that the order in which observations

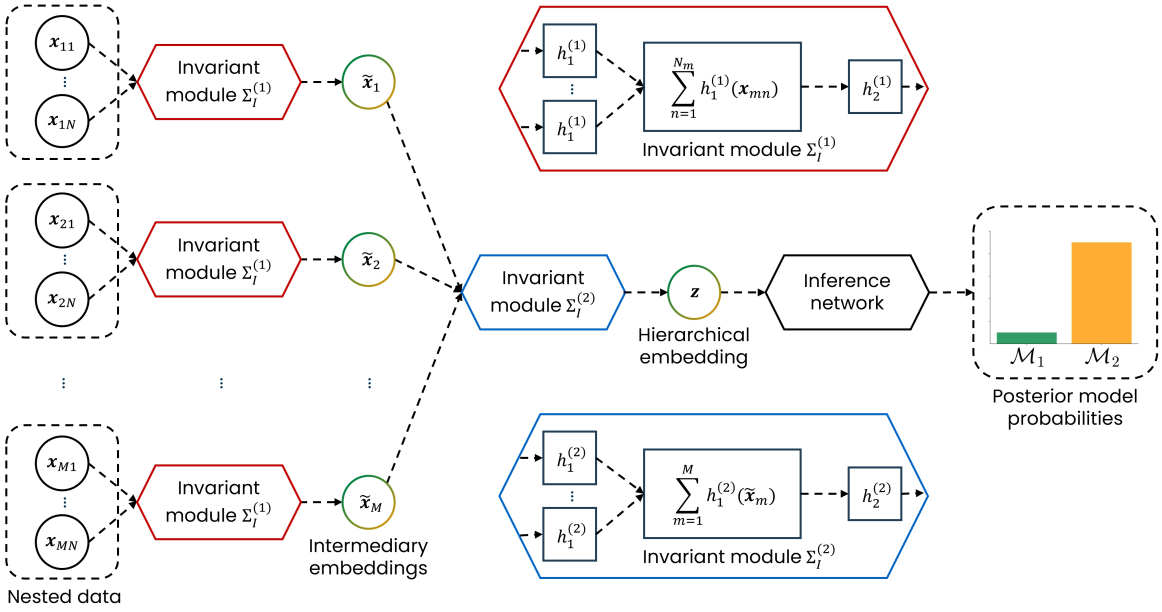


Figure 3.1: Our neural network architecture for comparing Bayesian hierarchical models: Invariant modules compress exchangeable nested data first within, then between groups, followed by an inference network approximating posterior model probabilities.

are collected does not affect the inference drawn from them (Bloem-Reddy & Teh, 2020). We eliminate the need to tediously learn this probabilistic structure within each hierarchical level by employing *permutation-invariant neural networks* (Radev et al., 2021; Zaheer et al., 2017). Starting at the lowest hierarchical level (e.g., the observations nested within each person), our architecture sequentially compresses nested data $\boldsymbol{x}_{mn}$ of M groups with N_m observations each into approximate posterior model probabilities of J compared models $q(\mathcal{M} \mid \boldsymbol{x})$ (see Figure 3.1):[1]

1. Within-group compression: An invariant module $\Sigma_I^{(1)}$ processes each group m's data matrix $\boldsymbol{x}_{mn}$ to a group-wise embedding vector $\widetilde{\boldsymbol{x}}_m$. $\Sigma_I^{(1)}$ consists of two non-linear functions $h_1^{(1)}$ and $h_2^{(1)}$, parametrized by neural networks, and an intermediate pooling operation (e.g., summation) ensuring permutation invariance:

$$\widetilde{\boldsymbol{x}}_m = \Sigma_I^{(1)}\left(\{\boldsymbol{x}_n\}_m\right) = h_2^{(1)}\left(\sum_{n=1}^{N_m} h_1^{(1)}(\boldsymbol{x}_{mn})\right), \tag{3.1}$$

   where the set notation $\{\boldsymbol{x}_n\}_m$ expresses exchangeability.

2. Between-group compression: A second invariant module $\Sigma_I^{(2)}$ processes the matrix $\widetilde{\boldsymbol{x}}$ of M group-wise embedding vectors $\widetilde{\boldsymbol{x}}_m$ to obtain a single hierarchical embedding $\boldsymbol{z}$

[1]Trainable neural network parameters are hidden in this chapter for notational clarity.

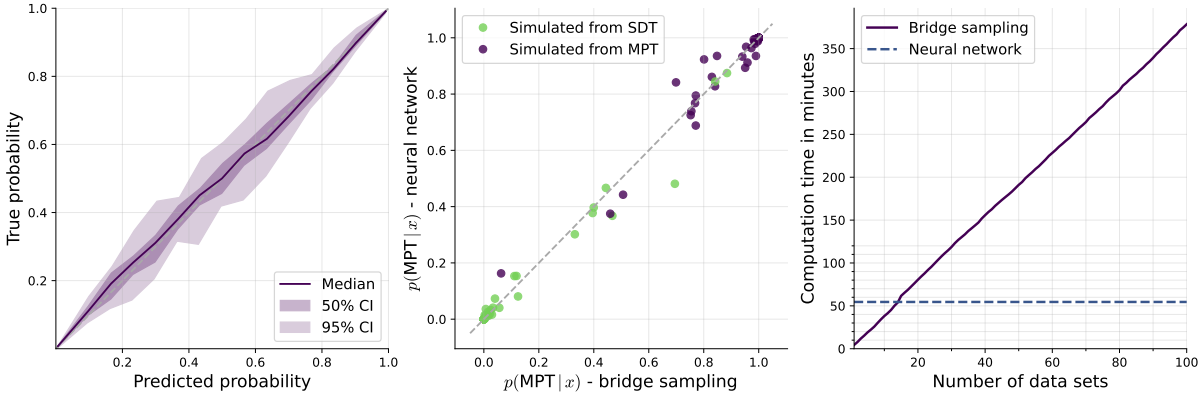
vector:

$$\boldsymbol{z} = \Sigma_I^{(2)}\left(\{\widetilde{\boldsymbol{x}}_m\}\right) = h_2^{(2)}\left(\sum_{m=1}^{M} h_1^{(2)}(\widetilde{\boldsymbol{x}}_m)\right). \tag{3.2}$$

3. Approximate inference: After the hierarchical data has been compressed according to its probabilistic structure, a final classification inference network q transforms the hierarchical embedding $\boldsymbol{z}$ to a vector of J approximate posterior model probabilities, $p(\mathcal{M} \mid \boldsymbol{x}) \approx q(\mathcal{M} \mid \boldsymbol{z})$.

In a first experiment featuring the comparison between simple hierarchical Gaussian models, we tested our architecture under increasingly difficult conditions induced by the scope of amortizing across data set sizes: from learning the model comparison problem for a fixed sample size of $M = 50$ groups and $N_m = 50$ observations, up to amortizing across $M = 1, 2, \ldots, 100$ groups and $N_m = 1, 2, \ldots, 100$ observations (i.e., 1 up to 10 000 total data points across all groups). In all conditions, we observed equivalent performance of our method to the gold-standard bridge sampling method. In a second benchmarking experiment, we chose a more realistic setting from recognition memory: the comparison of hierarchical models based on signal detection theory (SDT; Green & Swets, 1966) and multinomial processing trees (MPT; Riefer & Batchelder, 1988), respectively. These popular model classes differ in their fundamental cognitive assumptions, with SDT models assuming a continuous recognition process and MPT models a series of discrete steps. Nevertheless, the sparse information of binary old-new recognition data renders model comparison challenging. As shown in Figure 3.2, we found excellent calibration of approximate posterior model probabilities and again a convergence with bridge sampling. Figure 3.2c illustrates the initial computational investment and subsequent amortization of ABI methods. As a final validation, we tested the behavior of our method under model misspecification. When confronted with data generated from random noise rather than the SDT or the MPT model, we again found converging approximations with bridge sampling. While the following chapters will illustrate the complexity of model misspecification in ABI, we found robust approximation performance for at least this specific validation.

After these successful validations, we applied our method to a model comparison problem including intractable likelihood models, which is infeasible to approach with bridge sampling. Based on the setting and data of Wieschen et al. (2020), we compared four hierarchical EAMs arising from the combination of two factors: (i) whether main model parameters vary between trials, an actively discussed model extension (Boehm et al., 2018), and (ii) whether evidence accumulation noise follows a fixed Gaussian distribution for all participants, as in the drift-diffusion model (Ratcliff, 1978), or a flexible, participant-specific alpha-stable distribution, as in the recently proposed Lévy flight model of decision



(a) Calibration of the neural network.

(b) Convergence of posterior model probabilities.

(c) Computation times.

Figure 3.2: Results for our neural network approach and bridge sampling when comparing hierarchical signal detection theory (SDT) and multinomial processing tree (MPT) models. All posterior model probabilities refer to the MPT model.

making (see Section 2.1; Voss et al., 2019; Wieschen et al., 2020). Since the full data set consisted of $M \cdot N_M = 40 \cdot 900 = 36\,000$ data points, we used a novel training scheme consisting of pre-training with smaller simulated data sets of $N_M = 100$ trials and only subsequently using the full trial size for fine-tuning. Our hierarchical model comparison confirmed the advantages of including inter-trial variabilities found in the non-hierarchical analysis of Wieschen et al. (2020), but with a slight advantage of the full diffusion model over the full Lévy flight model for this particular data set. As hypothesized by Wieschen et al. (2020), the rather long experimental duration of 45 minutes might have caused fluctuations that are well-captured by inter-trial variability parameters.

Overall, our experiments confirmed the viability of our ABI method for hierarchical model comparison. While the presented deep learning architecture focuses on two-level hierarchical models with exchangeable data at each level, the modularity of our approach allows for straightforward extensions to further hierarchical levels. Additionally, modules at each hierarchical level can be exchanged with summary networks specialized for different probabilistic structures, such as the temporal within-person sequences assumed by time-series models (Driver & Voelkle, 2018; Schumacher et al., 2023). In a recent astrophysical work, Karchev et al. (2023)[2] used a similar sequential hierarchical network architecture to compare high-dimensional simulation-based models of type Ia supernovæ luminosity. Relatedly, Habermann et al. (2024) augmented the proposed hierarchical summary network architecture with a hierarchical inference network architecture to enable

[2]While the publication presented in this chapter was published in 2024, a preprint was already available online in 2023.

amortized parameter estimation, which makes ABI fully available for hierarchical models.

The evaluation of our proposed method for hierarchical model comparison has mostly been confined to simulated benchmarks, a single empirical application, and a single validation under model misspecification. How our ABI approach performs "in the wild"—where real data sets with potentially noisy measurements may violate various modeling assumptions—is a complex matter that remains largely unexplored due to its comprehensiveness. Addressing this gap is crucial, as the subsequent chapters will demonstrate.

Sensitivity-Aware Amortized Bayesian Inference (Publication II)

4

Publication	Link to full-text paper

Elsemüller, L., Olischläger, H., Schmitt, M., Bürkner, P. C., Köthe, U., & Radev, S. T. (2024). Sensitivity-aware amortized Bayesian inference. *Transactions on Machine Learning Research*.

Wagenmakers et al. (2022) recently argued that "any single analysis hides an iceberg of uncertainty". While Bayesian inference equips modelers with principled uncertainty quantification, this uncertainty rests on the specific set of assumptions, procedures, and data chosen for the statistical analysis. As a classic example, whenever results from Bayesian analyses are presented, the question "but what if one had used a different prior?" is already in the air. In recent years, awareness of the additional uncertainty induced by such external factors has increased: For instance, multiverse analyses (Steegen et al., 2016) repeat analyses across a range of plausible data processing schemes, and this holistic approach is becoming increasingly popular in psychological modeling. Multiverse analyses can be seen as an instantiation of the more general scheme of repetition-based *sensitivity analyses*, which quantify the change in results caused by modifying components of an analysis. While such sensitivity analyses are critical for reliable inference, they quickly become infeasible for complex models where fitting a model even once can already be computationally challenging. For example, fitting cognitive models with implicit likelihoods using ABC can require hours for each data set (Kangasrääsiö et al., 2019; Radev et al., 2020). ABI greatly speeds up inference across multiple data sets, but repeating the initial training—which can also require hours—for hundreds of plausible cognitive model specifications remains out of reach.

In this work, we develop *sensitivity-aware ABI*, a method for incorporating efficient sensitivity analyses for all major model components directly into ABI. Our approach builds upon a recently proposed decomposition of a Bayesian model into four major components (Bürkner et al., 2023): prior (P), likelihood (L), approximator (A), and data (D). We explicitly consider typically implicit sensitivity sources for all four components, such as the choice of prior or preprocessing scheme, as context variables $C = (C_P, C_L, C_A, C_D)$. This

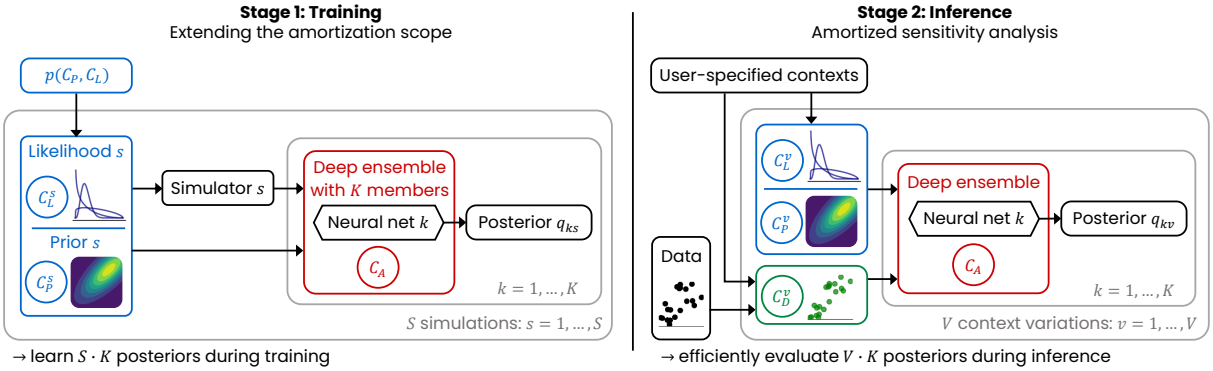


Figure 4.1: Sensitivity-aware amortized Bayesian inference: During training, a distribution $p(C_P, C_L)$ over plausible **p**rior and **l**ikelihood choices is encoded via context variables C_P and C_L in a deep ensemble of neural **a**pproximators forming C_A. During subsequent inference, near-instant neural network predictions conditioned on user-specified context C replace costly model refits for each combination in C, enabling large-scale sensitivity analyses.

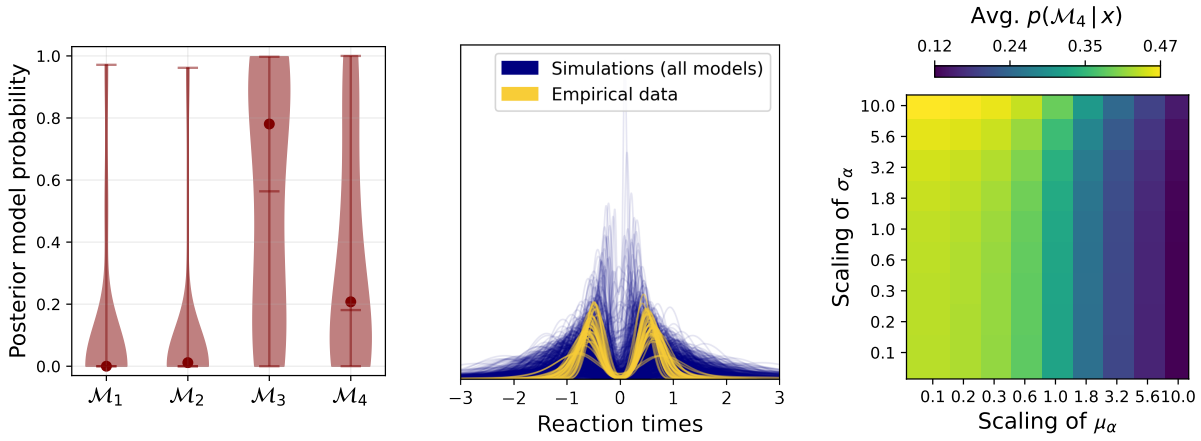
enables explicit modeling of these sensitivity sources as additional conditioning variables for posterior inference, next to $\boldsymbol{x}$, via $p(\boldsymbol{\theta} \mid \boldsymbol{x}, C)$.[1] As I will lay out in the following, our approach enables a systematic evaluation of sensitivity in all major model components without retraining a neural network for every possible configuration (see Figure 4.1).

We start with the notion that ABI is already well-suited for multiverse analyses that examine the influence of C_D, since a trained neural network can rapidly output posterior approximations for thousands of unseen data sets. In practice, this can be applied to analyze an arbitrary amount of data variations, e.g., from different preprocessing or bootstrapping schemes. The core contribution of our work lies in augmenting this strength with complementary approaches to quantify sensitivity in the remaining three model components. Assessing sensitivity to the assumptions of the probabilistic model is computationally costly for standard ABI since the training process is based on a *single* probabilistic model $p(\boldsymbol{\theta}, \boldsymbol{x}) = p(\boldsymbol{x} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$, and any modifications to prior or likelihood require costly retraining. Intuitively, if we want a neural network to generalize across different modeling assumptions, we have to include them in the training. We achieve this by sampling diverse plausible variations (C_P, C_L) of the probabilistic model from a "meta-prior" distribution $p(C_P, C_L)$ during training. We then inform the network of the specific context combination by providing conditioning variables C_P and C_L, extending the approximation target to $p(\boldsymbol{\theta} \mid \boldsymbol{x}, C_P, C_L)$. For example, C_L could be a dummy-encoded vector indicating different EAMs, while C_P could be a binary indicator encoding less or more informative

[1] In the publication, we propose sensitivity-aware ABI for both parameter estimation and model comparison, but I focus on the former here for notational brevity.

prior distributions. Targeting $p(\boldsymbol{\theta} \mid \boldsymbol{x}, C_P, C_L)$ as a training goal achieves amortization across $p(C_P, C_L)$, so that inference under all $C_P \times C_L$ possible model formulations can be obtained in near real-time by passing the respective indicator variables together with the empirical data. Due to the structural similarities of plausible model instantiations (e.g., all EAMs are relatively similar compared to the set of all possible models), jointly learning the inverse problem for all plausible model instantiations is much more efficient than separate training (up to 500 times in our experiments). Finally, the approximator itself can also cause sensitivities: Neural networks are known to be unstable in out-of-distribution situations where the encountered data deviates from their training distribution (Ovadia et al., 2019). This is particularly relevant for ABI, where model misspecification may cause a gap between simulated training data and real-world observations (Frazier et al., 2024; Kelly et al., 2025; Schmitt et al., 2024). Thus, we propose to augment ABI workflows with a proven and practical method for out-of-distribution detection, namely, deep ensembles (Lakshminarayanan et al., 2017; Ovadia et al., 2019; Yang et al., 2022). Concretely, we realize an approximator context C_A via the predictive variability of a deep ensemble of multiple equally configured and independently trained neural networks. This allows for a two-step approach to assessing approximator sensitivity: (i) When validating a trained neural network on thousands of simulated data sets, performance variability between the ensemble members informs about sensitivity due to finite training or suboptimal convergence; (ii) when applying a trained neural network on real-world data, variability between the ensemble members—despite stable performance in step (i)—signals an out-of-distribution scenario. In terms of ABI, this scenario translates to model misspecification, since the empirical data diverges from the simulated training distribution. While our ensemble approach requires repeated training, repeated simulation can be circumvented by reusing the same simulated training data for all ensemble members.

With these modifications, sensitivity in all model components can be efficiently assessed at inference time, which enables large-scale sensitivity analyses. In our experiments, we validated sensitivity-aware ABI in three real-world modeling settings. The first experiment benchmarked amortized prior sensitivity analysis in modeling the first two weeks of COVID-19 outbreak dynamics with an SIR model (Dehning et al., 2020), where our benchmarks confirmed minor performance trade-offs but massive efficiency gains for amortizing over a family of priors. In the second experiment, we used our method to approximate global warming forecasts for 18 climate models, each under a less versus a more informative prior, with a single neural network. Jointly training across the resulting 36 C_L and C_P combinations was especially beneficial in this setting: Since climate model simulations are extremely computationally intensive, each model was implicitly defined by only a limited amount of publicly available simulations. Based on current spatial temperature



(a) Approximator sensitivity. (b) Data visualization. (c) Prior sensitivity.

Figure 4.2: Sensitivities in the comparison between four hierarchical evidence accumulation models: (a) The approximator sensitivity revealed by a deep ensemble of 20 neural networks, with dots representing the original results and horizontal bars the median and extrema of the deep ensemble predictions. (b) The simulated and empirical reaction time distributions, with negative and positive reaction times distinguishing between the two decision options. (c) The effect of systematically widening and shrinking the hierarchical priors on the Lévy flight model's unique α parameter on the average evidence for $\mathcal{M}_4$. $\mathcal{M}_1$ = Drift-diffusion model. $\mathcal{M}_2$ = Lévy flight model. $\mathcal{M}_3$ = Drift-diffusion model with inter-trial variabilities. $\mathcal{M}_4$ = Lévy flight model with inter-trial variabilities.

data from 2023, our results revealed global warming forecasts of the time until reaching the 1.5 °C threshold to be robust to the choice of prior but to vary substantially between climate models. In the third experiment, we leveraged the new opportunities unlocked by sensitivity-aware ABI to conduct a large-scale sensitivity analysis on all components (prior, likelihood, approximator, and data) of the model comparison between hierarchical drift-diffusion model and Lévy flight model variants in Chapter 3. The resulting 162 000 posterior approximations revealed substantial approximator sensitivity: Despite near-perfect performance of all ensemble members on simulated validation data, we found highly inconsistent predictions on the empirical data (see Figure 4.2a). Even though the distributions of simulated reaction time data visually captured the empirical reaction time distributions (Figure 4.2b), follow-up tests revealed the empirical data to be out-of-distribution of the typical set of model simulations seen during training. A possible reason for this subtle misspecification could be data heterogeneity introduced by aggregating across two tasks. While the main result of the analysis in Chapter 3, the advantage of including inter-trial variability parameters, still held, the deep ensemble uncovered substantial uncertainty due to model misspecification. Finally, we investigated the sensitivity of the results to the specification of the hierarchical priors on the Lévy flight model's unique α

parameter, which governs the individual tendency for jumps in evidence accumulation. Figure 4.2c shows that model comparison results are insensitive to the specification of the scale parameter σ_α but sensitive to the location parameter μ_α. The increasing evidence for the Lévy flight model with smaller scaling factors (i.e., broader priors) on μ_α implies that the observed data are best explained by prior specifications allowing for large interindividual differences in the tendency for jumps in evidence accumulation. This broad exploration of a wide range of prior perturbations would have been, to the best of our knowledge, impossible to realize efficiently with any existing method, even if the likelihood of the Lévy flight model was tractable.

Overall, sensitivity-aware ABI resolves a critical tension in modern modeling: With growing complexity, the need for sensitivity analyses to shed light on the many moving model parts grows—as well as the infeasibility of traditional repetition-based sensitivity analyses, especially for implicit likelihood models. Our approach scales sensitivity analyses to complex models and large data sets by providing a modular workflow for efficiently assessing sensitivity in all model components. This enables sensitivity-aware modeling as a default choice in ABI, allowing researchers to investigate the implications of *all* plausible modeling paths instead of committing to a single one.

Since the publication of this work, sensitivity-aware ABI methods have been integrated into multiple applied studies: Gahlot et al. (2025) employed sensitivity-aware modeling over a continuous range of plausible rock physics models (i.e., likelihoods) for forecasting subsurface CO_2 flows in underground energy storage monitoring. Additionally, Schumacher et al. (2025) used a deep ensemble to assess approximator sensitivity when comparing different approaches to integrating non-stationary dynamics into evidence accumulation modeling. Lastly, in Liss et al. (2025), we leveraged sensitivity-aware ABI for a comprehensive investigation of the Lévy flight model's psychological interpretation. Specifically, we investigated whether its proposed mechanism of sudden jumps in evidence accumulation can explain fast errors under time pressure. When contrasting parameter estimates between the Lévy flight model and the drift-diffusion model, we leveraged the similarity of the two models to encode them in a single trained approximator via a likelihood context. Moreover, we safeguarded all parameter estimation and model comparison workflows with deep ensembles against approximator sensitivity. The results underscored the value of the Lévy flight model's evidence jump mechanism in capturing the distinct data patterns that arise under time pressure.

Our experimental evaluations confirmed the practicality of sensitivity-aware ABI for applied workflows and highlighted the complex issue of model misspecification in ABI. While our ensemble approach aims to *detect* this issue, the following chapter will thoroughly investigate promising methods for *reducing* the susceptibility to model misspecification.

5 Unsupervised Domain Adaptation for Robust Amortized Bayesian Inference (Publication III)

Publication	**Link to full-text paper**

Elsemüller, L., Pratz, V., von Krause, M., Voss, A., Bürkner, P. C., & Radev, S. T. (2025). Does unsupervised domain adaptation improve the robustness of amortized Bayesian inference? A systematic evaluation. Under review at *Transactions on Machine Learning Research*.

The arguably most famous quote of statistics states that "all models are wrong" (Box, 1976). This is especially pertinent for statistical models grounded in scientific theory, since parsimony underpins core principles in the philosophy of science (e.g., Occam's razor; Sober, 2015). In ABI, this leads to a natural tension between training on simulations from simplified statistical models on the one hand, and inference on noisy real-world data from unknown data-generating processes on the other. Therefore, ABI inherently operates in a shift between the simulated and the observed domain, exacerbated by the degree of model misspecification. Combined with the fragility of neural networks when confronted with data outside their training distribution (Ovadia et al., 2019), real-world robustness poses a critical challenge for ABI (Frazier et al., 2024; Kelly et al., 2025; Schmitt et al., 2024). This is also a central obstacle in scaling cognitive modeling with ABI, as we can rarely expect our computational models to exhaustively capture the full complexity of human cognition.

Motivated by the danger of approximator sensitivity highlighted in Chapter 4, this work takes a deeper dive into the fundamental challenge of inferential robustness in the real world. In the context of ABI, we can frame this issue as a *domain adaptation* problem of generalizing from the simulated domain with known ground-truths (i.e., data-generating parameter values) to the observed domain, where we cannot obtain ground-truths (such as "true" cognitive parameter values for participants). This framing allows us to turn to the field of unsupervised domain adaptation (UDA), which studies the adaptation of machine learning algorithms trained in a supervised source domain where ground-truths

are available to an unsupervised target domain without ground-truths. UDA methods rest on the central theoretical insight that the error in the target domain can be minimized by minimizing not only the error in the source domain (i.e., the optimization goal of standard training), but also reducing the domain shift itself by minimizing the divergence between the source domain data and target domain data (Ben-David et al., 2006, 2010; Redko et al., 2020). Since we cannot directly modify the data distributions, most UDA methods aim to learn a transformation $h_{\boldsymbol{\psi}}$ that aligns the domains in the embedding space of a neural network, leading to *domain-invariant* embeddings. ABI is well-suited for a combination with UDA, since simulation-based training already minimizes error in the simulated source domain and ABI workflows usually learn a transformation $h_{\boldsymbol{\psi}}$ compressing potentially high-dimensional data $\boldsymbol{x}$ to learned summary statistics $h_{\boldsymbol{\psi}}(\boldsymbol{x})$. Indeed, the match between the distributions of simulated and observed summary statistics has recently been identified as critical for inferential robustness (Frazier et al., 2024; Huang et al., 2023; Schmitt et al., 2024; Wehenkel et al., 2025). Integrating UDA can therefore be achieved by adding an incentive $\mathcal{L}_{\text{UDA}}$ to the standard loss function $\mathcal{L}_{\text{NPE}}$ from Equation 2.13 to align the distributions between simulated summary statistics $h_{\boldsymbol{\psi}}(\boldsymbol{x})$ and observed summary statistics $h_{\boldsymbol{\psi}}(\boldsymbol{x}_{\text{obs}})$:

$$\begin{aligned} \mathcal{L}_{\text{NPE-UDA}}(\boldsymbol{\phi}, \boldsymbol{\psi}) &= \mathcal{L}_{\text{NPE}} + \lambda \cdot \mathcal{L}_{\text{UDA}} \\ &= \mathbb{E}_{p(\boldsymbol{\theta},\boldsymbol{x})\, p(\boldsymbol{x}_{\text{obs}})} \big[-\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x})) + \lambda \cdot d(h_{\boldsymbol{\psi}}(\boldsymbol{x}), h_{\boldsymbol{\psi}}(\boldsymbol{x}_{\text{obs}})) \big], \end{aligned} \tag{5.1}$$

where the weight λ controls the relative importance of domain alignment regularization and $d(\cdot, \cdot)$ is a divergence measure that attains its global minimum if and only if $h_{\boldsymbol{\psi}}(\boldsymbol{x}) = h_{\boldsymbol{\psi}}(\boldsymbol{x}_{\text{obs}})$.

Focusing on neural posterior estimation (NPE), we systematically evaluate the combination with two popular UDA approaches (see Figure 5.1 for an overview of the training modifications and Figure 5.2 for an illustration of the resulting summary space adaptations): NPE-MMD directly minimizes the distance in summary space by setting the maximum mean discrepancy (MMD; Gretton et al., 2012) as the divergence measure $d(\cdot, \cdot)$. The MMD is a popular metric in generative modeling that measures the distance between two distributions based on a set of samples from each distribution (Bischoff et al., 2024). NPE-DANN, on the other hand, aligns the summary space implicitly via domain-adversarial neural networks (DANN; Ganin et al., 2016). Here, adversarial competition between the summary network $h_{\boldsymbol{\psi}}$ and an additional domain classification network incentivizes the summary network to output domain-invariant summary statistics.

Two recent studies exploring NPE-MMD variants (Huang et al., 2023; Swierc et al., 2024) found promising results for robustifying inference against model misspecification.

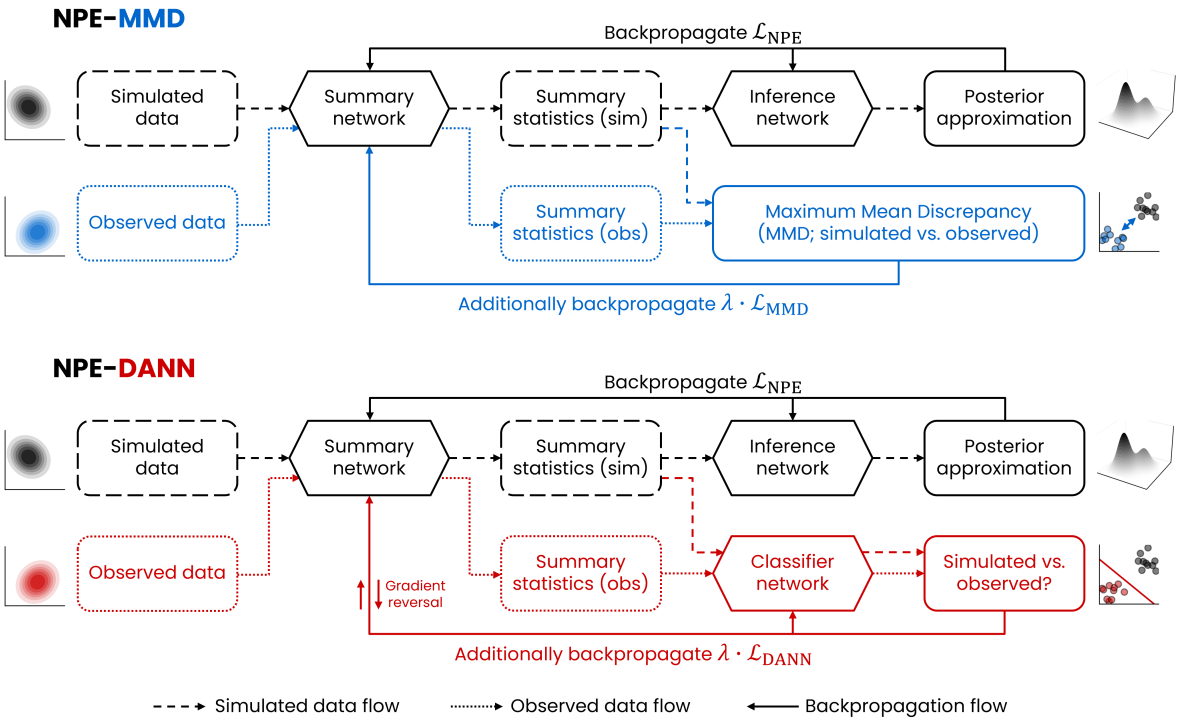
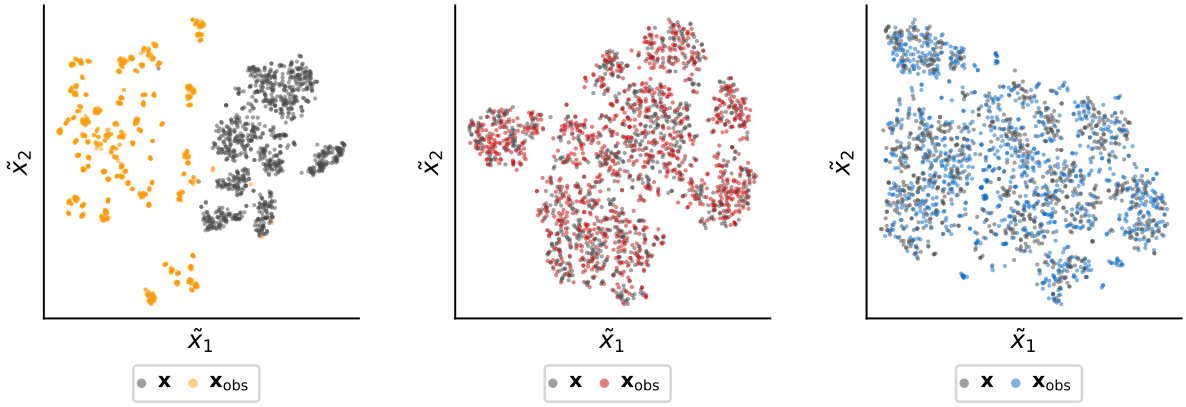


Figure 5.1: NPE-UDA methods combining neural posterior estimation (NPE) with unsupervised domain adaptation (UDA): NPE-UDA methods aim to enhance robustness in the observed domain by augmenting simulation-based training with the additional goal of aligning the simulated and the observed domain in summary space. NPE-MMD (maximum mean discrepancy) directly minimizes the distance between distributions, whereas NPE-DANN (domain-adversarial neural networks) uses adversarial competition between an auxiliary domain classifier network and the summary network.

However, both studies focused on likelihood misspecification by adding artificial noise to the observed data, where ignoring unmodeled information in the observed data is beneficial. Prior misspecification, where unmodeled information in the data is crucial for counteracting a badly chosen prior (e.g., that places little probability density on the relevant parameter region), has mostly been unexplored. Further, despite DANN being the prevailing UDA approach, its combination with NPE has not been investigated yet. Thus, a thorough investigation of the behavior of NPE-UDA methods in various misspecification scenarios is essential before such methods can be used in applied modeling projects. This is especially important in cognitive modeling, where inevitably simplified models of human cognition are employed to draw conclusions from noisy behavioral measurements.

NPE-UDA methods come with a number of additional hyperparameters that potentially influence their effects, such as the architecture of the domain classification network in NPE-DANN. Thus, in a first exploratory study, we used Bayesian hyperparameter optimization (Akiba et al., 2019) to systematically quantify the trade-offs introduced by these extra



(a) NPE (b) NPE-DANN (c) NPE-MMD

Figure 5.2: Summary space differences between standard neural posterior estimation (NPE) and NPE methods integrating unsupervised domain adaptation (UDA). Whereas the domains are clearly separated for NPE, the NPE-UDA methods lead to domain alignment in summary space. 2-dimensional t-SNE (Van der Maaten & Hinton, 2008) visualizations of the 32-dimensional summary spaces in a benchmarking experiment with the challenging task of inferring 256 parameters in probabilistic image denoising.

hyperparameters in a classic likelihood misspecification setting. The results confirmed the promising performance of NPE-UDA methods under likelihood misspecification and found a relative insensitivity to most hyperparameters, but a critical dependence on the domain alignment weight λ. Two subsequent benchmarking experiments featuring multiple prior and likelihood misspecification scenarios revealed a nuanced pattern: We consistently found that NPE-UDA methods can flexibly adapt to various likelihood misspecifications, but substantially worsen performance under prior misspecification. Since we are ultimately interested in robustness in the real world, we also tested the NPE-UDA methods for modeling a large empirical data set of a binary decision-making task with the drift-diffusion model. Specifically, we tested whether the implicit filtering mechanism of NPE-UDA methods enables the handling of unprocessed behavioral data collected in a noisy online assessment environment. Surprisingly, we found only minor effects of the NPE-UDA methods. Despite the comparably noisy data, the drift-diffusion model was misspecified (based on atypicality in summary space; Schmitt et al., 2024) for less than 1% of the participants. Thus, general adaptation to the observed domain, which would be the default approach in practical applications, did not extend to this distinct subpopulation. The uncovered issue could be tackled by a two-step approach: In a first step, standard NPE would be used to obtain parameter estimates as well as summary-space-based misspecification scores for all participants, retaining a participant's parameter estimates if the model is well-specified. In a second step, the trained NPE approximator

could be fine-tuned with NPE-UDA methods to adapt to the subpopulation for which the model is misspecified.

Nevertheless, several hurdles remain on the path towards reliable real-world applicability of NPE-UDA methods: First, the subpopulation for which the model is misspecified might not be homogeneous itself. For example, some participants in an experiment might be inattentive, leading to negatively skewed reaction time distributions, while others might employ various guessing strategies, leading to positively skewed distributions.

Second, in practice, we rarely know the exact nature of misspecification—if we did, this knowledge would be incorporated into the statistical model in the first place. In cases of likelihood misspecification, NPE-UDA methods can flexibly adapt to decrease the influence of various unknown misspecification sources. On the other hand, NPE-UDA methods might also silently fail due to prior misspecification, which is challenging to diagnose in the real world due to the lack of ground-truth parameter values for observed data.

Third, we argue that like many other robust methods, NPE-UDA methods explicitly deviate from the analytic posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{obs}})$: They incentivize the summary network h_ψ to not only compress $\boldsymbol{x}_{\text{obs}}$ for posterior inference but also reduce the domain divergence in summary space. Interpreting the resulting posterior $p(\boldsymbol{\theta} \mid h_\psi(\boldsymbol{x}_{\text{obs}}))$ is challenging, since it is not transparent which parts of the data are filtered for achieving domain invariant summary statistics in a given application. Here, machine learning interpretability methods mapping summary space adaptations back to the data space are promising tools for gaining deeper insights into NPE-UDA mechanisms.

Lastly, we confirmed the central role of the λ hyperparameter controlling the weight of the domain adaptation loss (Zellinger et al., 2021) in the ABI context. For both NPE-UDA methods, we uncovered application-specific instabilities for higher λ values, leading to a total collapse of posterior inference, whereas too low λ values might not induce domain adaptation at all. This is part of the overarching UDA problem of finding optimal λ parameters despite not having access to any guiding ground-truth parameter values in the real world (Musgrave et al., 2022; Zellinger et al., 2021). While our work provided a first approach to this problem based on measuring the distance of posterior predictions to the observed data, it remains an open problem. The observed data itself is not a clean target, since it might contain various artifacts (e.g., outliers) that a robust method should not consider during posterior inference. Additionally, both established model misspecification metrics in ABI cannot be used for setting λ: The distance of the observed data from the training distribution in summary space (Schmitt et al., 2024) is no longer an objective measure since it is actively minimized by UDA. Deep ensemble variability on the observed data, as proposed in Chapter 4 and recently used for tuning λ in concurrent research on NPE-MMD for galaxy clustering analysis (Pierre et al., 2025), is also not suitable:

It incentivizes overly strong regularization, since collapsing the posterior to the prior minimizes ensemble variability. Although including further metrics for monitoring such collapses to the prior, such as posterior contraction (i.e., the change from prior to posterior), could help, the computational overhead associated with training deep ensembles renders them impractical for hyperparameter optimization involving many training repetitions.

Overall, our systematic evaluation revealed that, despite promising results of earlier studies focusing on likelihood misspecification scenarios, the complex effects of NPE-UDA methods prevent their straightforward integration into existing modeling workflows.

6 General Discussion

A central proposition of this thesis is that the fusion of deep learning, simulation science, and Bayesian inference holds great potential for scaling cognitive modeling, but this potential can only be realized through careful and critical adaptation. The methods developed in this dissertation represent concrete steps in the adaptation process for ABI, each designed to address a specific challenge in cognitive modeling: (i) a novel method for comparing Bayesian hierarchical models, (ii) an augmentation of ABI workflows for comprehensive yet efficient sensitivity analyses, and (iii) a systematic evaluation of the potential and limitations of UDA for robust ABI.

In this chapter, I will first review the phases of adapting ABI to the needs of cognitive modelers, discussing their current state as well as contextualizing the contributions of this thesis within each phase. Then, I will turn to future directions that encompass all three phases, concerning ABI as well as the prospective advancements in cognitive science it unlocks.

6.1 Phases of Adaptation

The adoption of any new scientific methodology proceeds in phases. For ABI, the initial phase focused on developing and demonstrating its fundamental capabilities. As ABI methods transition from specialized applications to broader use, we are entering the more mature phases of (i) ensuring trustworthiness and (ii) increasing accessibility. All three phases are interdependent: The availability of capabilities is what enables the application of a method in the first place, but even the most powerful tool will remain unused if trust or accessibility are not given. In the following, I will review the state of each phase, embed the contributions of this thesis, and discuss open challenges.

Expanding Existing Capabilities ABI has successfully established its core utility for cognitive modelers, providing efficient tools for parameter estimation and model comparison. However, prior to the research presented in this thesis, capabilities for the increasingly popular class of Bayesian hierarchical models were missing. The work presented in

Chapter 3 directly addresses this gap for model comparison, while a subsequent work by Habermann et al. (2024) extended the proposed architecture for hierarchical parameter estimation.

The field continues to evolve rapidly, with new methods emerging to address common modeling challenges, such as mixture modeling (Kucharský & Bürkner, 2025) or handling missing data (Verma et al., 2025; Wang et al., 2024). For hierarchical modeling, a particular challenge is the application to large empirical data sets with many groups, where simulations quickly become orders of magnitude larger than in the non-hierarchical (group-wise) case. We approached this challenge in Chapter 3 with a pre-training approach, where we first pre-trained the neural network on smaller simulated data sets and only used the full data size during a second fine-tuning step. This approach could be generalized to various schemes that continuously scale the simulation size during training (e.g., linearly or logistically) instead of starting with the full size, increasing training efficiency for large hierarchical but also non-hierarchical data sets. An even more promising approach was recently proposed by Arruda et al. (2025), who leverage a specialized diffusion-based inference network architecture to enable group-wise data processing for hierarchical parameter estimation. This approach shrinks the training simulations from potentially thousands of groups to a single group per simulated data set and alleviates the need for hierarchical summary networks. However, the proposed inference network architecture is only suited for generative tasks and thus not applicable for the classification task of amortized model comparison.

Currently, two major limitations regarding ABI's capabilities and their advancement remain: First, the speed advantages of amortized inference come at the cost of a potentially computationally expensive training phase. Chapter 4 addresses this limitation for the common case of multiple plausible probabilistic models, proposing to encode them within a single neural network to substantially reduce training time. More generally, ABI methods directly benefit from advances in the intensively researched field of deep learning (e.g., hardware, architectures, training algorithms), which further alleviates this limitation. Second, the rapid pace of development has rendered the only available benchmark suite for simulation-based inference (Lueckmann et al., 2021) not challenging enough for testing state-of-the-art ABI methods. This currently necessitates the construction of specific benchmark scenarios for validating new ABI methods. Thus, the field would greatly benefit from standardized, large-scale, and challenging benchmarks that capture the diversity of application domains, including cognitive modeling.

Ensuring Trustworthiness Robustness to model misspecification has been recognized as the primary challenge for ABI methods and their trustworthiness in multiple fields

(Cannon et al., 2022; Dingeldein et al., 2024; Frazier et al., 2024; Rainforth et al., 2024). ABI methods have repeatedly proven their performance on simulated data, but simulation-based assessments usually test the well-specified scenario by simulating training and test data from the same probabilistic model. As Chapter 4 and Chapter 5 demonstrated, the sensitivity of neural networks towards data that substantially deviates from their training distribution is especially relevant in ABI, where the training distribution consists exclusively of simulations from a simplified probabilistic model. At the same time, ABI can profit from advances in the broader machine learning fields of out-of-distribution detection and robustness. In this thesis, promising approaches from both fields were integrated into ABI and evaluated. In both of the developed approaches, the currently strict boundaries between training and validation in the simulated world and inference in the real world were softened by integrating observed data earlier in the workflow.

First, sensitivity-aware ABI as proposed in Chapter 4 draws on deep ensembles to provide a protective layer for the modeling workflow: By training small ensembles of networks by default, extrapolation instabilities in the posterior estimates can be detected. Thereby, the comprehensive suite of simulation-based ABI validations is extended by a diagnostic based on empirical data. Unlike abstract misspecification scores, this approach directly quantifies the impact of misspecification on the reliability of inferential results, as demonstrated for the comparison between EAMs in Chapter 4. Nevertheless, while our deep ensemble approach avoids repeated simulation, it comes with the drawback of training multiple neural networks. Bayesian neural networks offer a theoretically elegant alternative for quantifying approximator sensitivity within a single neural network, but are currently challenging to implement in practice (Arbel et al., 2024; Izmailov et al., 2021). It remains to be determined whether future advancements render Bayesian neural networks practically viable enough to serve as a principled tool for assessing approximator sensitivity in ABI workflows. In the meantime, the efficiency of deep ensembles can be increased by leveraging their predictive power (Lakshminarayanan et al., 2017) as a second advantage after validating approximator sensitivity: Recently, Yao et al. (2024) proposed a stacking approach for optimally combining posterior estimates from multiple approximators, which fully realizes the predictive power of ensemble approaches for ABI.

Second, Chapter 5 systematically evaluated UDA for enhancing robustness in ABI. Although ABI can naturally be framed as a domain adaptation problem and initial studies provided promising results, our evaluation highlighted the crucial dependence on the type of misspecification. This underscores the critical need for careful evaluation of methods imported from the broader machine learning field to scientific applications in ABI: While mechanisms like the automatic filtering of observed data in UDA can be advantageous when dealing with, e.g., prediction on noisy cosmological images (Agarwal et al., 2024),

they can also aggravate the impact of a badly chosen prior when used in the context of ABI. Instead of explicitly deviating from the analytic posterior as many robust methods, including NPE-UDA, do, explicitly strengthening adherence to the analytic posterior on observed data has recently shown potential in ABI (Mishra et al., 2025).

Another promising avenue of integrating observed data earlier in the ABI workflow consists of augmenting out-of-distribution detection mechanisms with machine learning interpretability methods. By translating discrepancies in the neural network's internal representations back to the data space, the specific aspects of the data that the model is failing to capture could be identified. This could not only inform about the nature of the misspecification in the iterative model-building process, but also provide bottom-up information about missing pieces in the underlying theoretical frameworks (e.g., aspects of observed reaction time distributions that are the most atypical compared to the simulated training data). In contrast to simply comparing simulated and observed data, such a remapping takes the summary network's learned compression for posterior inference into account.

Moving forward, workflows that holistically integrate advancements in misspecification detection, robustness, and interpretation will become increasingly important in ABI. As a current example, Li et al. (2024) suggested an amortized Bayesian workflow for explicit likelihood models that leverages the scalability of ABI for the majority of data sets, but falls back on more robust MCMC methods whenever diagnostics flag ABI as untrustworthy for a single data set. Such a workflow could be complemented by interpretability methods to automatically inform about the discrepancies of the data set to the simulated training distribution that cause misspecification.

Increasing Accessibility Ultimately, the goal of adapting ABI for scaling cognitive modeling is to make these methods accessible to a broad audience of applied researchers. Throughout this dissertation, all method developments have been rooted in open-source development, with a constant focus on the needs of prospective users and the feasibility of integrating new developments into existing end-to-end software workflows. Specifically, I contributed to software development, method implementation, user support, and tutorial design for the `BayesFlow` Python library for amortized Bayesian workflows (Radev et al., 2023). The adoption of `BayesFlow` is steadily growing in cognitive science—powering many of the ABI applications discussed in this thesis—but also in other fields as diverse as astrophysics (Zhou et al., 2024), engineering (Zeng et al., 2023), particle physics (Bieringer et al., 2021), or ornithology (Pitocchelli et al., 2025).

Software libraries aim to reduce complexity by providing an abstract, high-level interface while still allowing for the adaptation of all workflow components to the desired use case.

Here, trustworthiness again comes into play, since users need to be able to trust that abstracted workflow parts do not influence their inferential results. Adaptation of workflow components, in turn, is facilitated by introductory materials as well as open-source applications providing orientation, such as the EAM applications in all core publications of this thesis.

Despite continuous progress in its open-source infrastructure, ABI's novelty and interdisciplinary nature lead to several hurdles that practitioners currently face: A first hurdle consists of deep learning requiring several potentially decisive choices on a set of hyperparameters, such as the neural network architecture and size, the learning rate governing the step size during training, or the duration of the training process. This hurdle can be approached by augmenting ABI workflows with methods explored in this thesis: Deep ensembles as proposed in Chapter 4 can be extended to hyperparameter ensembles (Wenzel et al., 2020) for quantifying hyperparameter sensitivity. Further, `BayesFlow`'s modularity enables the seamless integration of efficient hyperparameter optimization algorithms, as employed in Chapter 5. On a higher level, providing robust default settings that generalize across a variety of settings is a primary goal of the `BayesFlow` library (Radev et al., 2023). Here, future work that systematically explores the hyperparameter space for the most important cognitive modeling applications, such as evidence accumulation modeling, would provide substantial insights into best practices for practitioners. A second hurdle is the establishment of validation workflows: While ABI enables large-scale validations, it lacks the decades of dedicated research in the intricacies, guarantees, and interpretation of diagnostic tools compared to established methods like MCMC (Gelman et al., 2020). Comparison standards for interpreting diagnostics metrics in cognitive applications sometimes exist for more traditional diagnostics (e.g., the recovery of data-generating parameters for an implicit likelihood EAM variant in Miletić et al., 2017), but are often missing for diagnostics that are computationally challenging to obtain with non-amortized methods (e.g., simulation-based calibration, which requires a large number of model fits; Modrák et al., 2025; Talts et al., 2018).

6.2 Future Directions

Based on the work presented in this thesis, several promising future avenues emerge. These avenues jointly advance all three previously discussed phases—capabilities, trustworthiness, and accessibility—of adapting ABI methods for scaling cognitive modeling.

Towards Fine-Grained Models of Human Cognition A central theme of this thesis is that modern inference techniques can liberate model development from methodological constraints, enabling a "specification-first" approach to modeling (Haaf et al., 2025). For decades, the availability of a tractable likelihood function was a major limitation in translating psychological theory into computational cognitive models. Simulation-based methods like ABI replace the strong constraint of likelihood tractability with the far less restrictive constraint of simulatability: If a theoretical process can be implemented as a computational simulator, it can be fit to data. Additionally, amortized inference enables fast iterations in the model-building process: For example, large-scale simulation studies of parameter recovery have been transformed from dedicated and computationally intensive projects to a default validation step within each ABI workflow.

These advancements enable researchers to revisit existing theoretically appealing implicit likelihood models as well as unrestrictedly build novel theory-driven models. This is especially impactful for EAMs, where the fine-grained modeling of evidence accumulation as a stochastic dynamic process frequently leads to an intractable likelihood: For instance, the integration of information leakage and mutual inhibition between accumulators proposed by the leaky competing accumulator model (Usher & McClelland, 2001) has been widely acknowledged as a step towards more neurally plausible EAMs, but adoption has been hindered by the resulting implicit likelihood (Miletić et al., 2017; Wientjes & Holroyd, 2025). A similar tension between theoretical elegance and practical hardships caused by an intractable likelihood function exists for the diffusion model for conflict tasks (Evans & Servant, 2020; Ulrich et al., 2015), where ABI recently allowed for efficient Bayesian estimation (Schaefer et al., 2025). ABI further enabled extensive validation of the relatively new Lévy flight model of decision making—despite its implicit likelihood—in the publications of this thesis and beyond (Ebrahimi Mehr & Rad, 2024; Hato et al., 2025; Liss et al., 2025; Rasanan et al., 2024; von Krause & Radev, 2025; Wieschen et al., 2020, 2024). Lastly, ABI also allows for fundamentally new methodological approaches in cognitive modeling, such as the superstatistical ABI framework by Schumacher et al. (2023), which explicitly models temporal dynamics of cognitive parameters (e.g., the trajectory of an individual's decision caution throughout an experimental session).

A particularly exciting prospect of ABI consists of broadening the information basis of cognitive models. As models grow in complexity to capture finer-grained cognitive dynamics, relying solely on behavioral data, which only captures the outcome of a cognitive process, can become an information bottleneck. For example, even if the leaky competing accumulator model can now be easily estimated with ABI, its parameters are challenging to recover (Miletić et al., 2017). This suggests that behavioral data alone may not be rich enough to constrain the model's complex interplay of evidence accumulators. ABI

addresses this limitation by removing likelihood tractability constraints, which in turn enables a straightforward integration of data from multiple modalities within a single probabilistic model. By integrating high-resolution psychophysiological data streams such as eye-tracking or neuroimaging (e.g., EEG, MEG), researchers can build *multimodal* cognitive models that explain not just the outcome of a decision but also the neural and physiological dynamics that unfold throughout the process. This can render previously unidentifiable parameters identifiable, as recently demonstrated for the usually fixed diffusion coefficient representing within-trial noise in evidence accumulation (Nunez et al., 2025). Moreover, multimodal cognitive models enable explicit modeling of shared cognitive processes that manifest in different data modalities: For instance, ABI has been a cornerstone of a recent approach to neurocognitive EAM models that jointly explain trial-level behavioral and EEG data through shared cognitive processes (Ghaderi-Kangavari et al., 2023).

Looking even further ahead, I expect the AI for science paradigm to have a profound impact on cognitive science. In addition to scaling modeling, as pursued in this thesis, the frontier of AI for cognitive science is being explored in further ways: For example, Eckstein et al. (2023) systematically replaced components of cognitive models with flexible neural networks to discover overly restrictive model components. Binz et al. (2025) took this direction a step further by fine-tuning a large language model on large-scale experimental data to create a black-box emulator of human cognition. Relatedly, Holt et al. (2025) proposed to automate parts of the model-building process by pairing simulator generation via large language models with simulation-based inference to iteratively refine a probabilistic model. While such approaches can provide valuable bottom-up information for advancing our understanding of human cognition, automation in the model-building process can be a double-edged sword—carrying the risk of creating theory-poor, overly complex models that may overfit noise rather than capture meaningful cognitive processes. Here, the Bayesian framework becomes particularly important by providing principled tools for managing complexity: informative priors can encode theoretical constraints and promote sparsity, while Bayesian model comparison offers a rigorous method for determining whether added model complexity is justified. Additionally, the rapid, simulation-based diagnostics inherent to the ABI workflow allow for the immediate identification of unrecoverable model parts. By scaling Bayesian inference, this thesis contributes to facilitating the handling of fine-grained yet theoretically and empirically grounded models of human cognition.

Towards Foundation Models for Amortized Bayesian Inference Whereas the previous section discussed scaling cognitive modeling via ABI, this chapter discusses the complementary perspective of advancing cognitive modeling by scaling ABI itself. The

famous *bitter lesson* thesis in AI research (Sutton, 2019) proposes that scaling computation and data has consistently proven to be the most effective strategy for improving performance. This has been impressively demonstrated in recent years of large language model research, where scaling has repeatedly enabled major capability leaps. At the same time, the term *foundation model* has emerged to describe large deep learning architectures that have been trained on vast and diverse amounts of data and can be used for a variety of downstream applications (Bommasani et al., 2021). Such foundation models provide a wide range of capabilities, mitigate unstable out-of-distribution settings via an extremely broad training distribution, and can often easily be accessed via web-based interfaces.

While most neural networks used in ABI today are fairly small compared to language foundation models—with millions instead of billions of trainable parameters—the same benefits of scaling likely apply. Moreover, ABI foundation models can be seen as the logical consequence of a constant push towards increasing the scope of amortization, as detailed in the following. Sensitivity-aware ABI (Chapter 4) represents a first step towards ABI foundation models: It enables amortization across diverse contexts while explicit prior and likelihood context conditioning retains inferential exactness. As our experiments demonstrated, such a joint encoding of multiple probabilistic models is highly efficient, since it leverages the typically high similarity between multiple plausible models for a given phenomenon (e.g., different EAM variants). In our work, we directly pass the context information together with the output of the summary network as conditions to the inference network. In future applications, dedicated context summary networks could be used to efficiently encode a wide range of contexts. Further, our context-aware approach assumes knowledge about all prior and likelihood contexts of interest at training time, requiring fine-tuning for additional contexts. This is reasonable for sensitivity analyses, but represents a limitation that restricts the flexibility of ABI foundation models. Here, first works are enabling more flexibility at inference time, either through explicit prior adaptation (Chang, Loka, et al., 2025; Chang, Rissanen, et al., 2025; Gloeckler et al., 2024; Whittle et al., 2025) or by leveraging in-context learning with large, pre-trained approximators (Reuter et al., 2025; Vetter et al., 2025).

Overall, ABI foundation models possess the potential to drastically lower the barrier to entry for sophisticated Bayesian analyses. Similarly to language foundation models enabling accessible amortized text generation for a wide range of users without dedicated deep learning knowledge or hardware, pre-trained and thoroughly validated ABI foundation models could be easily accessed via a web-based interface, performing inference on uploaded data in near real-time. Scaling components like the network architecture, training duration, and the number of simulations is conceptually straightforward in the context of ABI foundation models. However, scaling training data diversity is less straightforward,

since this requires manual construction of a diverse set of scientifically grounded simulators. Automation in the model-building process, as discussed above, holds the potential to accelerate this process. During the development of ABI foundation models, standardized and large-scale benchmarks for typical cognitive modeling settings, as proposed in Section 6.1, would enable fast iteration based on comprehensive evaluations. In the near future, EAMs as a popular model class in cognitive modeling with many intractable likelihood variants would particularly benefit from dedicated ABI foundation models.

6.3 Concluding Remarks

The human mind is inherently complex. Cognitive modeling seeks to formally capture this complexity in testable computational models. While Bayesian inference offers a principled statistical framework for balancing complexity and parsimony in these models, the direct translation of theoretical ideas into probabilistic models has historically been limited by computational barriers. This thesis contributed to overcoming these constraints by scaling Bayesian cognitive modeling with ABI. By enabling a seamless iterative cycle between formalizing a theoretical idea as a simulator and rigorously evaluating it on large-scale data, ABI fosters a dynamic interplay between theoretical insight and empirical evidence. These principles extend beyond cognitive modeling, contributing to the overarching AI for science vision of accelerating scientific discovery across all quantitative domains.

References

Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A. J., Bambrick, J., Bodenstein, S. W., Evans, D. A., Hung, C.-C., O'Neill, M., Reiman, D., Tunyasuvunakool, K., Wu, Z., Žemgulytė, A., Arvaniti, E., ... Jumper, J. M. (2024). Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, *630*(8016), 493–500. https://doi.org/10.1038/s41586-024-07487-w

Agarwal, S., Ćiprijanović, A., & Nord, B. D. (2024). Neural network prediction of strong lensing systems with domain adaptation and uncertainty quantification. *arXiv preprint arXiv:2411.03334*.

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2623–2631.

Arbel, J., Pitas, K., Vladimirova, M., & Fortuin, V. (2024). A primer on Bayesian neural networks: Review and debates. *Statistical Science*, 1–46.

Arruda, J., Pandey, V., Sherry, C., Barroso, M., Intes, X., Hasenauer, J., & Radev, S. T. (2025). Compositional amortized inference for large-scale hierarchical Bayesian models. *arXiv preprint arXiv:2505.14429*.

Beaumont, M. A. (2019). Approximate Bayesian computation. *Annual Review of Statistics and Its Application*, *6*(1), 379–403. https://doi.org/10.1146/annurev-statistics-030718-105212

Beaumont, M. A., Zhang, W., & Balding, D. J. (2002). Approximate Bayesian computation in population genetics. *Genetics*, *162*(4), 2025–2035. https://doi.org/10.1093/genetics/162.4.2025

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., & Vaughan, J. W. (2010). A theory of learning from different domains. *Machine Learning*, *79*, 151–175. https://doi.org/10.1007/s10994-009-5152-4

Ben-David, S., Blitzer, J., Crammer, K., & Pereira, F. (2006). Analysis of representations for domain adaptation. *Advances in Neural Information Processing Systems*, *19*.

Bennett, C. H. (1976). Efficient estimation of free energy differences from Monte Carlo data. *Journal of Computational Physics*, *22*(2), 245–268. https://doi.org/10.1016/0021-9991(76)90078-4

Berens, P., Cranmer, K., Lawrence, N. D., von Luxburg, U., & Montgomery, J. (2023). AI for science: An emerging agenda. *arXiv preprint arXiv:2303.04217*.

Berger, J. O. (1990). Robust Bayesian analysis: Sensitivity to the prior. *Journal of Statistical Planning and Inference*, *25*(3), 303–328. https://doi.org/10.1016/0378-3758(90)90079-A

Betancourt, M., & Girolami, M. (2015). Hamiltonian Monte Carlo for hierarchical models. In *Current trends in Bayesian methodology with applications*. Chapman; Hall/CRC.

Bieringer, S., Butter, A., Heimel, T., Höche, S., Köthe, U., Plehn, T., & Radev, S. T. (2021). Measuring QCD splittings with invertible networks. *SciPost Physics*, *10*(6), 126. https://doi.org/10.21468/SciPostPhys.10.6.126

Binz, M., Akata, E., Bethge, M., Brändle, F., Callaway, F., Coda-Forno, J., Dayan, P., Demircan, C., Eckstein, M. K., Éltető, N., Griffiths, T. L., Haridi, S., Jagadish, A. K., Ji-An, L., Kipnis, A., Kumar, S., Ludwig, T., Mathony, M., Mattar, M., … Schulz, E. (2025). A foundation model to predict and capture human cognition. *Nature*. https://doi.org/10.1038/s41586-025-09215-4

Bischoff, S., Darcher, A., Deistler, M., Gao, R., Gerken, F., Gloeckler, M., Haxel, L., Kapoor, J., Lappalainen, J. K., Macke, J. H., Moss, G., Pals, M., Pei, F. C., Rapp, R., Sağtekin, A. E., Schröder, C., Schulz, A., Stefanidi, Z., Toyota, S., … Vetter, J. (2024). A practical guide to sample-based statistical distances for evaluating generative models in science. *Transactions on Machine Learning Research*.

Bloem-Reddy, B., & Teh, Y. W. (2020). Probabilistic symmetries and invariant neural networks. *Journal of Machine Learning Research*, *21*(90), 1–61.

Boag, R. J., Innes, R. J., Stevenson, N., Bahg, G., Busemeyer, J. R., Cox, G. E., Donkin, C., Frank, M. J., Hawkins, G. E., Heathcote, A., Hedge, C., Lerche, V., Lilburn, S. D., Logan, G. D., Matzke, D., Miletić, S., Osth, A. F., Palmeri, T. J., Sederberg, P. B., … Forstmann, B. U. (2025). An expert guide to planning experimental tasks for evidence-accumulation modeling. *Advances in Methods and Practices in Psychological Science*, *8*(2). https://doi.org/10.1177/25152459251336127

Boehm, U., Annis, J., Frank, M. J., Hawkins, G. E., Heathcote, A., Kellen, D., Krypotos, A.-M., Lerche, V., Logan, G. D., Palmeri, T. J., van Ravenzwaaij, D., Servant, M., Singmann, H., Starns, J. J., Voss, A., Wiecki, T. V., Matzke, D., & Wagenmakers, E.-J. (2018). Estimating across-trial variability parameters of the diffusion decision model: Expert advice and recommendations. *Journal of Mathematical Psychology*, *87*, 46–75. https://doi.org/https://doi.org/10.1016/j.jmp.2018.09.004

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N. S., Chen, A. S., Creel, K. A., Davis, J., Demszky, D.,

... Liang, P. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Box, G. E. (1976). Science and statistics. *Journal of the American Statistical Association*, *71*(356), 791–799. https://doi.org/10.1080/01621459.1976.10480949

Bürkner, P.-C., Scholz, M., & Radev, S. T. (2023). Some models are useful, but how do we know which ones? Towards a unified Bayesian model taxonomy. *Statistic Surveys*, *17*, 216–310. https://doi.org/10.1214/23-SS145

Cannon, P., Ward, D., & Schmon, S. M. (2022). Investigating the impact of model misspecification in neural simulation-based inference. *arXiv preprint arXiv:2209.01845*.

Chang, P. E., Loka, N. R. B. S., Huang, D., Remes, U., Kaski, S., & Acerbi, L. (2025). Amortized probabilistic conditioning for optimization, simulation and inference. *International Conference on Artificial Intelligence and Statistics*.

Chang, P. E., Rissanen, S., Loka, N. R. B. S., Huang, D., & Acerbi, L. (2025). Inference-time prior adaptation in simulation-based inference via guided diffusion models. *Frontiers in Probabilistic Inference: Learning Meets Sampling*.

Cranmer, K., Brehmer, J., & Louppe, G. (2020). The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, *117*(48), 30055–30062. https://doi.org/10.1073/pnas.1912789117

Dehning, J., Zierenberg, J., Spitzner, F. P., Wibral, M., Neto, J. P., Wilczek, M., & Priesemann, V. (2020). Inferring change points in the spread of COVID-19 reveals the effectiveness of interventions. *Science*, *369*(6500), eabb9789. https://doi.org/10.1126/science.abb9789

Diggle, P. J., & Gratton, R. J. (1984). Monte Carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *46*(2), 193–212. https://doi.org/10.1111/j.2517-6161.1984.tb01290.x

Dingeldein, L., Silva-Sánchez, D., Dimprima, E., Grigorieff, N., Covino, R., & Cossio, P. (2024). Amortized identification of biomolecular conformations in cryo-EM using simulation-based inference. *Biophysical Journal*, *123*(3), 282a. https://doi.org/10.1016/j.bpj.2023.11.1758

Driver, C. C., & Voelkle, M. C. (2018). Hierarchical Bayesian continuous time dynamic modeling. *Psychological Methods*, *23*(4), 774–799. https://doi.org/10.1037/met0000168

Ebrahimi Mehr, M., & Rad, J. A. (2024). Investigating the potential psychological significance of the alpha parameter in the Lévy flight model of decision making: A reliability analysis approach. *OSF Preprints*. https://doi.org/10.31219/osf.io/58p2v

Eckstein, M. K., Summerfield, C., Daw, N., & Miller, K. (2023). Predictive and interpretable: Combining artificial neural networks and classic cognitive models to understand

human learning and decision making. *Proceedings of the Annual Meeting of the Cognitive Science Society*, *45*(45).

Elsemüller, L., Olischläger, H., Schmitt, M., Bürkner, P.-C., Koethe, U., & Radev, S. T. (2024). Sensitivity-aware amortized Bayesian inference. *Transactions on Machine Learning Research*.

Elsemüller, L., Pratz, V., von Krause, M., Voss, A., Bürkner, P.-C., & Radev, S. T. (2025). Does unsupervised domain adaptation improve the robustness of amortized Bayesian inference? A systematic evaluation. Under review at *Transactions on Machine Learning Research*.

Elsemüller, L., Schnuerch, M., Bürkner, P.-C., & Radev, S. T. (2024). A deep learning method for comparing Bayesian hierarchical models. *Psychological Methods*. Advance online publication. https://doi.org/10.1037/met0000645

Evans, N. J., & Servant, M. (2020). A comparison of conflict diffusion models in the flanker task through pseudolikelihood Bayes factors. *Psychological Review*, *127*(1), 114–135. https://doi.org/10.1037/rev0000165

Evans, N. J., & Wagenmakers, E.-J. (2020). Evidence accumulation models: Current limitations and future directions. *The Quantitative Methods for Psychology*, *16*, 73–90. https://doi.org/10.20982/tqmp.16.2.p073

Farrell, S., & Lewandowsky, S. (2018). *Computational modeling of cognition and behavior*. Cambridge University Press. https://doi.org/10.1017/CBO9781316272503

Frazier, D. T., Kelly, R., Drovandi, C., & Warne, D. J. (2024). The statistical accuracy of neural posterior and likelihood estimation. *arXiv preprint arXiv:2411.12068*.

Gahlot, A. P., Erdinc, H. T., & Herrmann, F. J. (2025). Sensitivity-aware rock physics enhanced digital shadow for underground-energy storage monitoring. *arXiv preprint arXiv:2504.14405*.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., March, M., & Lempitsky, V. (2016). Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, *17*(59), 1–35.

Geffner, T., Papamakarios, G., & Mnih, A. (2023). Compositional score modeling for simulation-based inference. *International Conference on Machine Learning*, 11098–11116.

Gelman, A., Vehtari, A., Simpson, D., Margossian, C. C., Carpenter, B., Yao, Y., Kennedy, L., Gabry, J., Bürkner, P.-C., & Modrák, M. (2020). Bayesian workflow. *arXiv preprint arXiv:2011.01808*.

Gelman, A., & Yao, Y. (2020). Holes in Bayesian statistics. *Journal of Physics G: Nuclear and Particle Physics*, *48*(1), 014002. https://doi.org/10.1088/1361-6471/abc3a5

Ghaderi-Kangavari, A., Rad, J. A., & Nunez, M. D. (2023). A general integrative neurocognitive modeling framework to jointly describe EEG and decision-making on single trials. *Computational Brain & Behavior*, *6*, 317–376. https://doi.org/10.1007/s42113-023-00167-4

Gloeckler, M., Deistler, M., Weilbach, C. D., Wood, F., & Macke, J. H. (2024). All-in-one simulation-based inference. *International Conference on Machine Learning*, 15735–15766.

Gong, X., Huskey, R., Eden, A., & Ulusoy, E. (2023). Computationally modeling mood management theory: A drift-diffusion model of people's preferential choice for valence and arousal in media. *Journal of Communication*, *73*(5), 476–493. https://doi.org/10.1093/joc/jqad020

Green, D. M., & Swets, J. A. (1966). *Signal detection theory and psychophysics* (Vol. 1). Wiley.

Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., & Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research*, *13*, 723–773.

Gronau, Q. F., Wagenmakers, E.-J., Heck, D. W., & Matzke, D. (2019). A simple method for comparing complex models: Bayesian model comparison for hierarchical multinomial processing tree models using Warp-III bridge sampling. *Psychometrika*, *84*(1), 261–284. https://doi.org/10.1007/s11336-018-9648-3

Haaf, J. M., Klaassen, F., & Rouder, J. N. (2025). Bayes factor vs. posterior predictive model assessment: Insights from ordinal constraints. *Computational Brain & Behavior*, 1–10. https://doi.org/10.1007/s42113-025-00240-0

Haaf, J. M., & Rouder, J. N. (2019). Some do and some don't? Accounting for variability of individual difference structures. *Psychonomic Bulletin & Review*, *26*(3), 772–789. https://doi.org/10.3758/s13423-018-1522-x

Habermann, D., Schmitt, M., Kühmichel, L., Bulling, A., Radev, S. T., & Bürkner, P.-C. (2024). Amortized Bayesian multilevel models. *arXiv preprint arXiv:2408.13230*.

Hato, T., Schumacher, L., Radev, S. T., & Voss, A. (2025). Lévy versus Wiener: Assessing the effects of model misspecification on diffusion model parameters. *Computational Brain & Behavior*, 1–19. https://doi.org/10.1007/s42113-025-00248-6

Hawkins, G. E., Forstmann, B. U., Wagenmakers, E.-J., Ratcliff, R., & Brown, S. D. (2015). Revisiting the evidence for collapsing boundaries and urgency signals in perceptual decision-making. *Journal of Neuroscience*, *35*(6), 2476–2484. https://doi.org/10.1523/JNEUROSCI.2410-14.2015

Heck, D. W., Boehm, U., Böing-Messing, F., Bürkner, P.-C., Derks, K., Dienes, Z., Fu, Q., Gu, X., Karimova, D., Kiers, H. A. L., Klugkist, I., Kuiper, R. M., Lee, M. D., Leenders, R., Leplaa, H. J., Linde, M., Ly, A., Meijerink-Bosman, M.,

Moerbeek, M., ... Hoijtink, H. (2023). A review of applications of the Bayes factor in psychological research. *Psychological Methods*, *28*(3), 558–579. https://doi.org/https://doi.org/10.1037/met0000454

Henrich, F., Hartmann, R., Pratz, V., Voss, A., & Klauer, K. C. (2024). The seven-parameter diffusion model: An implementation in Stan for Bayesian analyses. *Behavior Research Methods*, *56*(4), 3102–3116. https://doi.org/10.3758/s13428-023-02179-1

Heringhaus, M. E., Zhang, Y., Zimmermann, A., & Mikelsons, L. (2022). Towards reliable parameter extraction in MEMS final module testing using Bayesian inference. *Sensors*, *22*(14), 5408. https://doi.org/10.3390/s22145408

Hermans, J., Begy, V., & Louppe, G. (2020). Likelihood-free MCMC with amortized approximate ratio estimators. *International Conference on Machine Learning*, 4239–4248.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Holt, S., Luyten, M. R., Berthon, A., & van der Schaar, M. (2025). G-sim: Generative simulations with large language models and gradient-free calibration. *International Conference on Machine Learning*.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366. https://doi.org/10.1016/0893-6080(89)90020-8

Huang, D., Bharti, A., Souza, A., Acerbi, L., & Kaski, S. (2023). Learning robust statistics for simulation-based inference under model misspecification. *Advances in Neural Information Processing Systems*, *36*, 7289–7310.

Izmailov, P., Vikram, S., Hoffman, M. D., & Wilson, A. G. G. (2021). What are Bayesian neural network posteriors really like? *International Conference on Machine Learning*, 4629–4640.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., ... Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature*, *596*(7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2

Kangasrääsiö, A., Jokinen, J. P., Oulasvirta, A., Howes, A., & Kaski, S. (2019). Parameter inference for computational cognitive models with approximate Bayesian computation. *Cognitive Science*, *43*(6), e12738. https://doi.org/10.1111/cogs.12738

Karchev, K., Trotta, R., & Weniger, C. (2023). SimSIMS: Simulation-based supernova Ia model selection with thousands of latent variables. *arXiv preprint arXiv:2311.15650*.

Kelly, R. P., Warne, D. J., Frazier, D. T., Nott, D. J., Gutmann, M. U., & Drovandi, C. (2025). Simulation-based Bayesian inference under model misspecification. *arXiv preprint arXiv:2503.12315*.

Kucharskỳ, Š., & Bürkner, P. C. (2025). Amortized Bayesian mixture models. *arXiv preprint arXiv:2501.10229*.

LaBerge, D. (1962). A recruitment theory of simple behavior. *Psychometrika*, *27*(4), 375–396. https://doi.org/10.1007/BF02289645

Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, *30*.

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Alet, F., Ravuri, S., Ewalds, T., Eaton-Rosen, Z., Hu, W., Merose, A., Hoyer, S., Holland, G., Vinyals, O., Stott, J., Pritzel, A., Mohamed, S., & Battaglia, P. (2023). Learning skillful medium-range global weather forecasting. *Science*, *382*(6677), 1416–1421. https://doi.org/10.1126/science.adi2336

Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., & Teh, Y. W. (2019). Set Transformer: A framework for attention-based permutation-invariant neural networks. *International Conference on Machine Learning*, 3744–3753.

Li, C., Vehtari, A., Bürkner, P.-C., Radev, S. T., Acerbi, L., & Schmitt, M. (2024). Amortized Bayesian workflow. *arXiv preprint arXiv:2409.04332*.

Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., & Le, M. (2023). Flow matching for generative modeling. *International Conference on Learning Representations*.

Liss, J. V., von Krause, M., Elsemüller, L., Hunsmann, E. M., & Lerche, V. (2025). Time to jump: Exploring the distribution of noise in evidence accumulation as a function of time pressure. Under review at *Cognitive Psychology*.

Liu, X., Gong, C., & Liu, Q. (2023). Flow straight and fast: Learning to generate and transfer data with rectified flow. *International Conference on Learning Representations*.

Lueckmann, J.-M., Boelts, J., Greenberg, D., Goncalves, P., & Macke, J. (2021). Benchmarking simulation-based inference. *International Conference on Artificial Intelligence and Statistics*, 343–351.

MacKay, D. (2003). *Information theory, inference and learning algorithms*. Cambridge University Press.

Marin, J.-M., Pudlo, P., Estoup, A., & Robert, C. (2018). Likelihood-free model choice. In *Handbook of approximate Bayesian computation* (pp. 153–178). Chapman; Hall/CRC.

Marin, J.-M., Pudlo, P., Robert, C. P., & Ryder, R. J. (2012). Approximate Bayesian computational methods. *Statistics and Computing*, *22*(6), 1167–1180. https://doi.org/10.1007/s11222-011-9288-2

McKay, R., Langdon, R., & Coltheart, M. (2006). Need for closure, jumping to conclusions, and decisiveness in delusion-prone individuals. *The Journal of Nervous and Mental Disease*, *194*(6), 422–426. https://doi.org/10.1097/01.nmd.0000221353.44132.25

Meng, X.-L., & Wong, W. H. (1996). Simulating ratios of normalizing constants via a simple identity: A theoretical exploration. *Statistica Sinica*, 831–860.

Miletić, S., Turner, B. M., Forstmann, B. U., & van Maanen, L. (2017). Parameter recovery for the leaky competing accumulator model. *Journal of Mathematical Psychology*, *76*, 25–50. https://doi.org/10.1016/j.jmp.2016.12.001

Mishra, A., Habermann, D., Schmitt, M., Radev, S. T., & Bürkner, P.-C. (2025). Robust amortized Bayesian inference with self-consistency losses on unlabeled data. *arXiv preprint arXiv:2501.13483*.

Modrák, M., Moon, A. H., Kim, S., Bürkner, P., Huurre, N., Faltejsková, K., Gelman, A., & Vehtari, A. (2025). Simulation-based calibration checking for Bayesian computation: The choice of test quantities shapes sensitivity. *Bayesian Analysis*, *20*(2), 461–488. https://doi.org/10.1214/23-BA1404

Musgrave, K., Belongie, S., & Lim, S.-N. (2022). Three new validators and a large-scale benchmark ranking for unsupervised domain adaptation. *arXiv preprint arXiv:2208.07360*.

Navarro, D. J., & Fuss, I. G. (2009). Fast and accurate calculations for first-passage times in Wiener diffusion models. *Journal of Mathematical Psychology*, *53*(4), 222–230. https://doi.org/10.1016/j.jmp.2009.02.003

Neal, A., & Kwantes, P. J. (2009). An evidence accumulation model for conflict detection performance in a simulated air traffic control task. *Human Factors*, *51*(2), 164–180. https://doi.org/10.1177/0018720809335071

Nunez, M. D., Schubert, A.-L., Frischkorn, G. T., & Oberauer, K. (2025). Cognitive models of decision-making with identifiable parameters: Diffusion decision models with within-trial noise. *Journal of Mathematical Psychology*, *125*, 102917. https://doi.org/10.1016/j.jmp.2025.102917

Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J., Lakshminarayanan, B., & Snoek, J. (2019). Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*, *32*.

Palestro, J. J., Weichart, E., Sederberg, P. B., & Turner, B. M. (2018). Some task demands induce collapsing bounds: Evidence from a behavioral analysis. *Psychonomic Bulletin & Review*, *25*(4), 1225–1248. https://doi.org/10.3758/s13423-018-1479-9

Papamakarios, G., & Murray, I. (2016). Fast ε-free inference of simulation models with Bayesian conditional density estimation. *Advances in Neural Information Processing Systems*, *29*.

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., & Lakshminarayanan, B. (2021). Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, *22*(1), 2617–2680.

Papamakarios, G., Sterratt, D., & Murray, I. (2019). Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. *International Conference on Artificial Intelligence and Statistics*, 837–848.

Pereira, M., Megevand, P., Tan, M. X., Chang, W., Wang, S., Rezai, A., Seeck, M., Corniola, M., Momjian, S., Bernasconi, F., Blanke, O., & Faivre, N. (2021). Evidence accumulation relates to perceptual consciousness and monitoring. *Nature Communications*, *12*(1), 3261. https://doi.org/10.1038/s41467-021-23540-y

Picchini, U., & Tamborrino, M. (2024). Guided Sequential ABC Schemes for Intractable Bayesian Models. *Bayesian Analysis*, 1–32. https://doi.org/10.1214/24-BA1451

Pierre, S., Blancard, B. R.-S., Hahn, C., & Eickenberg, M. (2025). Mitigating model misspecification in simulation-based inference for galaxy clustering. *arXiv preprint arXiv:2507.03086*.

Pitocchelli, J., Albina, A., Bentley, R. A., Guerra, D., & Youngblood, M. (2025). Temporal stability in songs across the breeding range of Geothlypis philadelphia (Mourning Warbler) may be due to learning fidelity and transmission biases. *Ornithology*, *142*(1). https://doi.org/10.1093/ornithology/ukae046

Price, I., Sanchez-Gonzalez, A., Alet, F., Andersson, T. R., El-Kadi, A., Masters, D., Ewalds, T., Stott, J., Mohamed, S., Battaglia, P., Lam, R., & Willson, M. (2025). Probabilistic weather forecasting with machine learning. *Nature*, *637*(8044), 84–90. https://doi.org/10.1038/s41586-024-08252-9

Pritchard, J. K., Seielstad, M. T., Perez-Lezaun, A., & Feldman, M. W. (1999). Population growth of human Y chromosomes: A study of Y chromosome microsatellites. *Molecular Biology and Evolution*, *16*(12), 1791–1798. https://doi.org/10.1093/oxfordjournals.molbev.a026091

Purcell, B. A., & Palmeri, T. J. (2017). Relating accumulator model parameters and neural dynamics. *Journal of Mathematical Psychology*, *76*, 156–171. https://doi.org/10.1016/j.jmp.2016.07.001

Radev, S. T., D'Alessandro, M., Mertens, U. K., Voss, A., Köthe, U., & Bürkner, P.-C. (2021). Amortized Bayesian model comparison with evidential deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, *34*(8), 4903–4917. https://doi.org/10.1109/TNNLS.2021.3124052

Radev, S. T., Mertens, U. K., Voss, A., Ardizzone, L., & Köthe, U. (2020). Bayesflow: Learning complex stochastic models with invertible neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, *33*(4), 1452–1466. https://doi.org/10.1109/TNNLS.2020.3042395

Radev, S. T., Schmitt, M., Schumacher, L., Elsemüller, L., Pratz, V., Schälte, Y., Köthe, U., & Bürkner, P.-C. (2023). Bayesflow: Amortized Bayesian workflows with neural networks. *Journal of Open Source Software*, *8*(89), 5702. https://doi.org/10.21105/joss.05702

Rainforth, T., Foster, A., Ivanova, D. R., & Bickford Smith, F. (2024). Modern Bayesian experimental design. *Statistical Science*, *39*(1), 100–114. https://doi.org/10.1214/23-STS915

Rasanan, A. H. H., Rad, J. A., & Sewell, D. K. (2024). Are there jumps in evidence accumulation, and what, if anything, do they reflect psychologically? An analysis of Lévy flights models of decision-making. *Psychonomic Bulletin & Review*, *31*(1), 32–48. https://doi.org/10.3758/s13423-023-02284-4

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*(2), 59–108. https://doi.org/10.1037/0033-295X.85.2.59

Ratcliff, R., & Rouder, J. N. (1998). Modeling response times for two-choice decisions. *Psychological Science*, *9*(5), 347–356. https://doi.org/10.1111/1467-9280.00067

Ratcliff, R., & Tuerlinckx, F. (2002). Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, *9*(3), 438–481. https://doi.org/10.3758/BF03196302

Redko, I., Morvant, E., Habrard, A., Sebban, M., & Bennani, Y. (2020). A survey on domain adaptation theory: Learning bounds and theoretical guarantees. *arXiv preprint arXiv:2004.11829*.

Reuter, A., Rudner, T. G. J., Fortuin, V., & Rügamer, D. (2025). Can transformers learn full Bayesian inference in context? *International Conference on Machine Learning*.

Rezende, D., & Mohamed, S. (2015). Variational inference with normalizing flows. *International Conference on Machine Learning*, 1530–1538.

Riefer, D. M., & Batchelder, W. H. (1988). Multinomial modeling and the measurement of cognitive processes. *Psychological Review*, *95*(3), 318–339. https://doi.org/10.1037/0033-295X.95.3.318

Robert, C. P., Cornuet, J.-M., Marin, J.-M., & Pillai, N. S. (2011). Lack of confidence in approximate Bayesian computation model choice. *Proceedings of the National Academy of Sciences*, *108*(37), 15112–15117. https://doi.org/10.1073/pnas.1102900108

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408. https://doi.org/10.1037/h0042519

Rouder, J. N., Morey, R. D., & Pratte, M. S. (2017). Bayesian hierarchical models of cognition. In W. H. Batchelder, H. Colonius, E. N. Dzhafarov, & J. Myung (Eds.), *New handbook of mathematical psychology: Foundations and methodology* (pp. 504–551). Cambridge University Press.

Schad, D. J., Nicenboim, B., Bürkner, P.-C., Betancourt, M., & Vasishth, S. (2023). Workflow techniques for the robust use of Bayes factors. *Psychological Methods*, *28*(6), 1404–1426. https://doi.org/10.1037/met0000472

Schaefer, S. B., Radev, S. T., Göttmann, J., & Schubert, A.-L. (2025). Amortized Bayesian workflow for modeling congruency effects using the diffusion model for conflict tasks. *PsyArXiv Preprints*. https://doi.org/10.31234/osf.io/dypcw_v1

Schmitt, M., Bürkner, P.-C., Köthe, U., & Radev, S. T. (2024). Detecting model misspecification in amortized Bayesian inference with neural networks. In U. Köthe & C. Rother (Eds.), *Pattern recognition* (pp. 541–557). Springer Nature Switzerland.

Schnuerch, M., Nadarevic, L., & Rouder, J. N. (2021). The truth revisited: Bayesian analysis of individual differences in the truth effect. *Psychonomic Bulletin & Review*, *28*(3), 750–765. https://doi.org/10.3758/s13423-020-01814-8

Schumacher, L., Bürkner, P.-C., Voss, A., Köthe, U., & Radev, S. T. (2023). Neural superstatistics for Bayesian estimation of dynamic cognitive models. *Scientific Reports*, *13*(1), 13778. https://doi.org/10.1038/s41598-023-40278-3

Schumacher, L., Schnuerch, M., Voss, A., & Radev, S. T. (2025). Validation and comparison of non-stationary cognitive models: A diffusion model application. *Computational Brain & Behavior*, *8*(2), 191–210. https://doi.org/10.1007/s42113-024-00218-4

Shadlen, M. N., & Kiani, R. (2013). Decision making as a window on cognition. *Neuron*, *80*(3), 791–806. https://doi.org/10.1016/j.neuron.2013.10.047

Shallue, C. J., & Vanderburg, A. (2018). Identifying exoplanets with deep learning: A five-planet resonant chain around Kepler-80 and an eighth planet around Kepler-90. *The Astronomical Journal*, *155*(2), 94. https://doi.org/10.3847/1538-3881/aa9e09

Singmann, H., & Kellen, D. (2019). An introduction to mixed models for experimental psychology. In *New methods in cognitive psychology* (pp. 4–31). Routledge.

Sisson, S. A., Fan, Y., & Beaumont, M. (2018). *Handbook of approximate Bayesian computation*. CRC Press. https://doi.org/10.1201/9781315117195

Sisson, S. A., Fan, Y., & Tanaka, M. M. (2007). Sequential Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, *104*(6), 1760–1765. https://doi.org/10.1073/pnas.0607208104

Smith, P. L. (2000). Stochastic dynamic models of response time and accuracy: A foundational primer. *Journal of Mathematical Psychology*, *44*(3), 408–463. https://doi.org/10.1006/jmps.1999.1260

Sober, E. (2015). *Ockham's razors: A user's manual*. Cambridge University Press. https://doi.org/10.1017/CBO9781107705937

Sripada, C., & Weigard, A. (2021). Impaired evidence accumulation as a transdiagnostic vulnerability factor in psychopathology. *Frontiers in Psychiatry*, *12*, 627179. https://doi.org/10.3389/fpsyt.2021.627179

Steegen, S., Tuerlinckx, F., Gelman, A., & Vanpaemel, W. (2016). Increasing transparency through a multiverse analysis. *Perspectives on Psychological Science*, *11*(5), 702–712. https://doi.org/10.1177/1745691616658637

Stuhlmüller, A., Taylor, J., & Goodman, N. (2013). Learning stochastic inverses. *Advances in Neural Information Processing Systems*, *26*.

Sutton, R. (2019). The bitter lesson. *Incomplete Ideas*. Retrieved July 15, 2025, from http://www.incompleteideas.net/IncIdeas/BitterLesson.html

Swierc, P., Tamargo-Arizmendi, M., Ćiprijanović, A., & Nord, B. D. (2024). Domain-adaptive neural posterior estimation for strong gravitational lens analysis. *arXiv preprint arXiv:2410.16347*.

Talts, S., Betancourt, M., Simpson, D., Vehtari, A., & Gelman, A. (2018). Validating Bayesian inference algorithms with simulation-based calibration. *arXiv preprint arXiv:1804.06788*.

Theisen, M., Lerche, V., von Krause, M., & Voss, A. (2021). Age differences in diffusion model parameters: A meta-analysis. *Psychological Research*, *85*, 2012–2021. https://doi.org/10.1007/s00426-020-01371-8

Tillman, G., Van Zandt, T., & Logan, G. D. (2020). Sequential sampling models without random between-trial variability: The racing diffusion model of speeded decision making. *Psychonomic Bulletin & Review*, *27*(5), 911–936. https://doi.org/10.3758/s13423-020-01719-6

Ulrich, R., Schröter, H., Leuthold, H., & Birngruber, T. (2015). Automatic and controlled stimulus processing in conflict tasks: Superimposed diffusion processes and delta functions. *Cognitive Psychology*, *78*, 148–174. https://doi.org/10.1016/j.cogpsych.2015.02.005

Usher, M., & McClelland, J. L. (2001). The time course of perceptual choice: The leaky, competing accumulator model. *Psychological Review*, *108*(3), 550–592. https://doi.org/10.1037/0033-295x.108.3.550

Van De Schoot, R., Winter, S. D., Ryan, O., Zondervan-Zwijnenburg, M., & Depaoli, S. (2017). A systematic review of Bayesian articles in psychology: The last 25 years. *Psychological Methods*, *22*(2), 217–239. https://doi.org/10.1037/met0000100

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(11), 2579–2605.

Vanpaemel, W. (2010). Prior sensitivity in theory testing: An apologia for the Bayes factor. *Journal of Mathematical Psychology*, *54*(6), 491–498. https://doi.org/10.1016/j.jmp.2010.07.003

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*.

Verma, Y., Bharti, A., & Garg, V. (2025). Robust simulation-based inference under missing data via neural processes. *International Conference on Learning Representations*.

Vetter, J., Gloeckler, M., Gedon, D., & Macke, J. H. (2025). Effortless, simulation-efficient Bayesian inference using tabular foundation models. *arXiv preprint arXiv:2504.17660*.

von Krause, M., & Radev, S. T. (2025). A big data analysis of the associations between cognitive parameters and socioeconomic outcomes. *OSF Preprints*. https://doi.org/10.31219/osf.io/ge83u

von Krause, M., Radev, S. T., & Voss, A. (2022). Mental speed is high until age 60 as revealed by analysis of over a million participants. *Nature Human Behaviour*, *6*(5), 700–708. https://doi.org/10.1038/s41562-021-01282-7

Voss, A., Lerche, V., Mertens, U., & Voss, J. (2019). Sequential sampling models with variable boundaries and non-normal noise: A comparison of six models. *Psychonomic Bulletin & Review*, *26*(3), 813–832. https://doi.org/10.3758/s13423-018-1560-4

Voss, A., & Voss, J. (2008). A fast numerical algorithm for the estimation of diffusion model parameters. *Journal of Mathematical Psychology*, *52*(1), 1–9. https://doi.org/10.1016/j.jmp.2007.09.005

Wagenmakers, E.-J., Sarafoglou, A., & Aczel, B. (2022). One statistical analysis must not rule them all. *Nature*, *605*(7910), 423–425. https://doi.org/10.1038/d41586-022-01332-8

Wang, Z., Hasenauer, J., & Schälte, Y. (2024). Missing data in amortized simulation-based neural posterior estimation. *PLOS Computational Biology*, *20*(6), e1012184. https://doi.org/10.1371/journal.pcbi.1012184

Wehenkel, A., Gamella, J. L., Sener, O., Behrmann, J., Sapiro, G., Jacobsen, J.-H., & Cuturi, M. (2025). Addressing misspecification in simulation-based inference through data-driven calibration. *International Conference on Machine Learning*.

Weigard, A., Matzke, D., Tanis, C., & Heathcote, A. (2023). A cognitive process modeling framework for the ABCD study stop-signal task. *Developmental Cognitive Neuroscience*, *59*, 101191. https://doi.org/10.1016/j.dcn.2022.101191

Weinberg, H., Müller, F., & Cañal-Bruland, R. (2025). Context modulates evidence accumulation in split-second handball penalty decisions. *Cognitive Research: Principles and Implications*, *10*(1), 1–15. https://doi.org/10.1186/s41235-025-00615-8

Wenzel, F., Snoek, J., Tran, D., & Jenatton, R. (2020). Hyperparameter ensembles for robustness and uncertainty quantification. *Advances in Neural Information Processing Systems*, *33*, 6514–6527.

Whittle, G., Ziomek, J., Rawling, J., & Osborne, M. A. (2025). Distribution transformers: Fast approximate Bayesian inference with on-the-fly prior adaptation. *arXiv preprint arXiv:2502.02463*.

Wiecki, T. V., Sofer, I., & Frank, M. J. (2013). HDDM: Hierarchical Bayesian estimation of the drift-diffusion model in Python. *Frontiers in Neuroinformatics*, *7*. https://doi.org/10.3389/fninf.2013.00014

Wientjes, S., & Holroyd, C. B. (2025). Episodic memory and the temporal dynamics of cognitive control. *PsyArXiv Preprints*. https://doi.org/10.31234/osf.io/9jr52

Wieschen, E. M., Makani, A., Radev, S. T., Voss, A., & Spaniol, J. (2024). Age-related differences in decision-making: Evidence accumulation is more gradual in older age. *Experimental Aging Research*, *50*(5), 537–549. https://doi.org/10.1080/0361073X.2023.2241333

Wieschen, E. M., Voss, A., & Radev, S. T. (2020). Jumping to conclusion? A Lévy flight model of decision making. *The Quantitative Methods for Psychology*, *16*(2), 120–132. https://doi.org/10.20982/tqmp.16.2.p120

Yang, J., Wang, P., Zou, D., Zhou, Z., Ding, K., Peng, W., Wang, H., Chen, G., Li, B., Sun, Y., Du, X., Zhou, K., Zhang, W., Hendrycks, D., Li, Y., & Liu, Z. (2022). OpenOOD: Benchmarking generalized out-of-distribution detection. *Advances in Neural Information Processing Systems*, *35*, 32598–32611.

Yao, Y., Régaldo-Saint Blancard, B., & Domke, J. (2024). Simulation-based stacking. *International Conference on Artificial Intelligence and Statistics*, 4267–4275.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., & Smola, A. J. (2017). Deep sets. *Advances in Neural Information Processing Systems*, *30*.

Zammit-Mangion, A., Sainsbury-Dale, M., & Huser, R. (2025). Neural methods for amortized inference. *Annual Review of Statistics and Its Application*, *12*. https://doi.org/10.1146/annurev-statistics-112723-034123

Zellinger, W., Shepeleva, N., Dinu, M.-C., Eghbal-zadeh, H., Nguyen, H. D., Nessler, B., Pereverzyev, S., & Moser, B. A. (2021). The balancing principle for parameter choice in distance-regularized domain adaptation. *Advances in Neural Information Processing Systems*, *34*, 20798–20811.

Zeng, J., Todd, M. D., & Hu, Z. (2023). Probabilistic damage detection using a new likelihood-free Bayesian inference method. *Journal of Civil Structural Health Monitoring*, *13*(2), 319–341. https://doi.org/10.1007/s13349-022-00638-5

Zhou, L., Radev, S. T., Oliver, W. H., Obreja, A., Jin, Z., & Buck, T. (2024). Evaluating sparse galaxy simulations via out-of-distribution detection and amortized Bayesian model comparison. *arXiv preprint arXiv:2410.10606*.

Publication I

Publication	**Link to online full-text**

Elsemüller, L., Schnuerch, M., Bürkner, P. C., & Radev, S. T. (2024). A deep learning method for comparing Bayesian hierarchical models. *Psychological Methods*. Advance online publication. https://doi.org/10.1037/met0000645

The reprint of the publication can be found in the following.

AMERICAN PSYCHOLOGICAL ASSOCIATION

Psychological Methods

ISSN: 1082-989X

https://doi.org/10.1037/met0000645

A Deep Learning Method for Comparing Bayesian Hierarchical Models

Lasse Elsemüller[1], Martin Schnuerch[2], Paul-Christian Bürkner[3], and Stefan T. Radev[4]

[1] Institute of Psychology, Heidelberg University

[2] Department of Psychology, University of Mannheim

[3] Department of Statistics, TU Dortmund University

[4] Cluster of Excellence STRUCTURES, Heidelberg University

Abstract

Bayesian model comparison (BMC) offers a principled approach to assessing the relative merits of competing computational models and propagating uncertainty into model selection decisions. However, BMC is often intractable for the popular class of hierarchical models due to their high-dimensional nested parameter structure. To address this intractability, we propose a deep learning method for performing BMC on any set of hierarchical models which can be instantiated as probabilistic programs. Since our method enables amortized inference, it allows efficient re-estimation of posterior model probabilities and fast performance validation prior to any real-data application. In a series of extensive validation studies, we benchmark the performance of our method against the state-of-the-art bridge sampling method and demonstrate excellent amortized inference across all BMC settings. We then showcase our method by comparing four hierarchical evidence accumulation models that have previously been deemed intractable for BMC due to partly implicit likelihoods. Additionally, we demonstrate how transfer learning can be leveraged to enhance training efficiency. We provide reproducible code for all analyses and an open-source implementation of our method.

Keywords: Bayesian statistics, model comparison, hierarchical modeling, deep learning, cognitive modeling

Supplemental materials: https://doi.org/10.1037/met0000645.supp

Hierarchical models (HMs) or multilevel models play an increasingly important methodological role in the social and cognitive sciences (Farrell & Lewandowsky, 2018; Rouder et al., 2017). HMs embody probabilistic and structural information about nested data occurring frequently in various settings, such as educational research (Ulitzsch et al., 2020), experimental psychology (Vandekerckhove et al., 2011), epidemiology (Jalilian & Mateu, 2021) or astrophysics (Hinton et al., 2019), to name just a few. Crucially, HMs can often extract more information from rich data structures than their nonhierarchical counterparts (e.g., aggregate analyses), while retaining a relatively high intrinsic interpretability of their structural components (i.e., parameters). Moreover, viewed as formal instantiations of scientific hypotheses, HMs can be employed to systematically assign preferences to these hypotheses by means of formal model comparison. For example, Haaf and Rouder (2017) proposed a powerful framework based on Bayesian HMs for formulating and testing competing theoretical positions on quantitative versus qualitative individual differences.

We consider Bayesian model comparison (BMC) as a principled framework for comparing and ranking competing HMs via Occam's razor (Kass & Raftery, 1995; Lotfi et al., 2022; MacKay, 2003). However, standard BMC is analytically intractable for nontrivial HMs, as it requires marginalization over high-dimensional parameter

Lasse Elsemüller https://orcid.org/0000-0003-0368-720X

Martin Schnuerch https://orcid.org/0000-0001-6531-2265

Paul-Christian Bürkner https://orcid.org/0000-0001-5765-8995

Stefan T. Radev https://orcid.org/0000-0002-6702-9559

Parts of this work were presented at the Conference of Experimental Psychologists (2022) in Cologne, Germany, the meeting of the European Mathematical Psychology Group (2022) in Rovereto, Italy, and the Conference of Experimental Psychologists (2023) in Trier, Germany. Lasse Elsemüller was supported by the Google Cloud Research Credits program with the award GCP19980904. Lasse Elsemüller and Martin Schnuerch were supported by a grant from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation; GRK 2277) to the research training group Statistical Modeling in Psychology (SMiP). Paul-Christian Bürkner was supported by the Deutsche Forschungsgemeinschaft under Germany's Excellence Strategy—EXC-2075-390740016 (the Stuttgart Cluster of Excellence SimTech). Stefan T. Radev was supported by the Deutsche Forschungsgemeinschaft under Germany's Excellence Strategy—EXC-2181-390900948 (the Heidelberg Cluster of Excellence STRUCTURES). Lasse Elsemüller was previously affiliated with the Department of Psychology, University of Mannheim, and Paul-Christian Bürkner with the Cluster of Excellence SimTech, University of Stuttgart. Lasse Elsemüller, Martin Schnuerch, Paul-Christian Bürkner, and Stefan T. Radev thank Lukas Schumacher for helpful comments on this article. A preprint of this article has been posted online on arXiv (https://arxiv.org/abs/2301.11873).

The experimental materials are available at https://github.com/bayesflow-org/Hierarchical-Model-Comparison.

Correspondence concerning this article should be addressed to Lasse Elsemüller, Institute of Psychology, Heidelberg University, Hauptstraße 47, 69117 Heidelberg, Germany. Email: lasse.elsemueller@gmail.com

spaces. Moreover, BMC for complex HMs without explicit likelihoods (i.e., HMs available only as randomized simulators) becomes increasingly hopeless and precludes many interesting applications in the rapidly expanding field of simulation-based inference (Cranmer et al., 2020).

In this work, we propose to tackle the problem of BMC for arbitrarily complex HMs from a simulation-based perspective using deep learning. In particular, we build on the BayesFlow framework (Radev, D'Alessandro, et al., 2021; Radev et al., 2020) for simulation-based Bayesian inference and propose a novel hierarchical neural network architecture for approximating Bayes factors (BFs) and posterior model probabilities (PMPs) for any collection of HMs.

Our neural approach circumvents the steps of explicitly fitting all models and marginalizing over the parameter space of each model. Thus, it is applicable to both HMs with explicit likelihood functions and HMs accessible only through Monte Carlo simulations (i.e., with implicit likelihood functions). Moreover, our neural networks come with an efficient way to compute their calibration error (Guo et al., 2017), which provides an important diagnostic for self-consistency. Lastly, trained networks can be adapted to related tasks, substantially reducing the computational burden when dealing with demanding simulators.

The remainder of this article is organized as follows. In the Theoretical Background section, we introduce the theoretical background and related work on (hierarchical) BMC. We then present the rationale and details of our deep learning method in the Method section. In the Experiments section, we first present two validation studies of the proposed method: one that includes toy models for illustrative purposes and one that includes two popular classes of models from the field of cognitive psychology. We then apply our method to compare hierarchical evidence accumulation models with partly intractable likelihoods on a real data set. Finally, the Discussion section summarizes our contributions and discusses future perspectives.

Theoretical Background

Bayesian Hierarchical Modeling

In order to streamline statistical analyses, researchers rely on assumptions about the probabilistic structure or symmetry of the assumed data-generating process. For instance, the canonical assumption of independent and identically distributed (IID) data in psychological modeling states that (multivariate) observations are independent of each other and sampled from the same latent probability distribution (Nicenboim et al., 2022; Singmann & Kellen, 2019).

However, more complex dependencies may arise in a variety of contexts. For instance, if there are repeated measurements per participant or participants belong to different natural groups (e.g., school classes and working groups), the respective observations exhibit higher correlations within those clusters than across them. Ignoring this nested structure in statistical analyses may result in biased conclusions (Singmann & Kellen, 2019). Bayesian HMs formalize this structural knowledge by assuming that observations are sampled from a multilevel generative process (Gelman, 2006).

For instance, the generative recipe for a two-level Bayesian HM can be written as:

$$\boldsymbol{\eta} \sim p(\boldsymbol{\eta}), \tag{1}$$

$$\boldsymbol{\theta}_m \sim p(\boldsymbol{\theta}|\boldsymbol{\eta}) \quad \text{for } m = 1, \ldots, M, \tag{2}$$

$$\boldsymbol{x}_{mn} \sim p(\boldsymbol{x}|\boldsymbol{\theta}_m) \quad \text{for } n = 1, \ldots, N_m, \tag{3}$$

where $\boldsymbol{\eta}$ denotes the group-level parameters, $\boldsymbol{\theta}_m$ denotes the individual parameters in group m, and $\boldsymbol{x}_{mn}$ represents the nth observation in group m. Such a model suggests the following (nonunique) factorization of the joint distribution:

$$p(\boldsymbol{\eta}, \{\boldsymbol{\theta}_m\}, \{\boldsymbol{x}_{mn}\}) = p(\boldsymbol{\eta}) \prod_{m=1}^{M} p(\boldsymbol{\theta}_m|\boldsymbol{\eta}) \prod_{n=1}^{N_m} p(\boldsymbol{x}_{mn}|\boldsymbol{\theta}). \tag{4}$$

The set notation $\{\boldsymbol{\theta}_m\}$ and $\{\boldsymbol{x}_{mn}\}$ implies that the number of groups and observations in each group can vary across simulations, data sets and experiments and that these quantities are exchangeable.

HMs can be considered as a compromise between a separate analysis of each group (no-pooling) that neglects the information contained in the rest of the data and an aggregate analysis of the data (complete pooling) that loses the distinction between intragroup and intergroup variability (Hox et al., 2017). The partial pooling of information induced by HMs leads to more stable and accurate individual estimates through the shrinkage properties of multilevel priors, whereby single estimates inform each other (Bürkner, 2017; Gelman, 2006).

Despite having desirable properties, hierarchical modeling comes at the cost of increased complexity and computational demands. These increased demands make it hard or even impossible to compare competing HMs within the probabilistic framework of BMC. Before we highlight these challenges, we first describe the basics of BMC for nonhierarchical models.

Bayesian Model Comparison

The starting point of BMC is a collection of J competing generative models $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_J\}$. Each $\mathcal{M}_j$ is associated with a prior $p(\boldsymbol{\theta}_j|\mathcal{M}_j)$ on the parameters $\boldsymbol{\theta}_j$ and a generative mechanism, which is either defined analytically through a (tractable) likelihood density function $p(\boldsymbol{x}|\boldsymbol{\theta}_j, \mathcal{M}_j)$ or realized as a Monte Carlo simulation program $g_j(\boldsymbol{\theta}, \boldsymbol{z})$ with random states $\boldsymbol{z}$. Together, the prior and the likelihood define the Bayesian joint model

$$p(\boldsymbol{\theta}_j, \boldsymbol{x}|\mathcal{M}_j) = p(\boldsymbol{\theta}_j|\mathcal{M}_j)p(\boldsymbol{x}|\boldsymbol{\theta}_j, \mathcal{M}_j), \tag{5}$$

which is also tacitly defined for simulator-based models by marginalizing the joint distribution $p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}_j, \mathcal{M}_j)$ over all possible execution paths (i.e., random states) of the simulation program to obtain the implicit likelihood

$$p(\boldsymbol{x}|\boldsymbol{\theta}_j, \mathcal{M}_j) = \int p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}_j, \mathcal{M}_j)\mathrm{d}\boldsymbol{z}. \tag{6}$$

This integral is typically intractable for complex simulators (Cranmer et al., 2020), which makes it impossible to evaluate the likelihood and use standard Bayesian methods for parameter inference or model comparison.

The likelihood function, be it explicit or implicit, is a key object in Bayesian inference. When the parameters $\boldsymbol{\theta}$ are systematically varied and the data $\boldsymbol{x}$ held constant, the likelihood quantifies the relative fit of each model instantiation (defined by a fixed configuration $\boldsymbol{\theta}$) to the observed data.

When we marginalize the Bayesian joint model (Equation 5) over its parameter space, we obtain the marginal likelihood or Bayesian

evidence (see MacKay, 2003, Chapter 28):

$$p(\boldsymbol{x}|\mathcal{M}_j) = \int p(\boldsymbol{x}|\boldsymbol{\theta}_j, \mathcal{M}_j) p(\boldsymbol{\theta}_j|\mathcal{M}_j) \mathrm{d}\boldsymbol{\theta}_j. \tag{7}$$

The marginal likelihood can be interpreted as the probability that we would generate data $\boldsymbol{x}$ from model $\mathcal{M}_j$ when we randomly sample from the model's parameter prior $p(\boldsymbol{\theta}_j|\mathcal{M}_j)$. Moreover, the marginal likelihood is a central quantity for prior predictive hypothesis testing or model selection (Kass & Raftery, 1995; O'Hagan, 1995; Rouder & Morey, 2012). It is well-known that the marginal likelihood encodes a notion of Occam's razor arising from the basic principles of probability (Kass & Raftery, 1995, see also Figure 1). Thus, the marginal likelihood provides a foundation for the widespread use of Bayes factors (Heck et al., 2022) or PMPs (Congdon, 2006) for BMC.

The relative evidence for a pair of models can be computed through the ratio of marginal likelihoods for the two competing models $\mathcal{M}_j$ and $\mathcal{M}_k$,

$$\mathrm{BF}_{jk} = \frac{p(\boldsymbol{x}|\mathcal{M}_j)}{p(\boldsymbol{x}|\mathcal{M}_k)}. \tag{8}$$

This ratio is called Bayes factor (BF) and is widely used for quantifying pairwise model preference in Bayesian settings (Heck et al., 2022; Kass & Raftery, 1995). Accordingly, a $\mathrm{BF}_{jk} > 1$ indicates the preference for model j over model k given available data $\boldsymbol{x}$. Alternatively, one can directly focus on the (marginal) posterior probability of a model $\mathcal{M}_j$,

$$p(\mathcal{M}_j|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\mathcal{M}_j)p(\mathcal{M}_j)}{\sum_{j=1}^{J} p(\boldsymbol{x}|\mathcal{M}_j)p(\mathcal{M}_j)}, \tag{9}$$

where $p(\mathcal{M}_j)$ is a categorical (typically uniform) prior distribution encoding a researcher's prior beliefs regarding the plausibility of each considered model. This prior distribution is then updated with the information contained in the marginal likelihood $p(\boldsymbol{x}|\mathcal{M}_j)$ to obtain the corresponding PMP, $p(\mathcal{M}_j|\boldsymbol{x})$. Occasionally in the text, we will refer to the vector of PMPs for all J models as $\boldsymbol{\pi}$ and to the individual PMPs as π_j. The ratio of two PMPs, known as posterior odds, is in turn connected to the BF via the corresponding model priors:

$$\frac{p(\mathcal{M}_j|\boldsymbol{x})}{p(\mathcal{M}_k|\boldsymbol{x})} = \frac{p(\boldsymbol{x}|\mathcal{M}_j)}{p(\boldsymbol{x}|\mathcal{M}_k)} \times \frac{p(\mathcal{M}_j)}{p(\mathcal{M}_k)}. \tag{10}$$

Despite its intuitive appeal, the marginal likelihood (and thus BFs and PMPs) represents a well-known and widely appreciated source of intractability in Bayesian workflows, since it typically involves a multidimensional integral (Equation 7) over potentially unbounded parameter spaces (Gronau, Sarafoglou, et al., 2017; Lotfi et al., 2022). Furthermore, the marginal likelihood becomes doubly intractable when the likelihood function is itself not available (e.g., in simulation-based settings), thereby making the comparison of such models a challenging and sometimes, up to this point, hopeless endeavor.

Unsurprisingly, estimating the marginal likelihood (Equation 7) in the context of hierarchical models becomes even more challenging, since the number of parameters over which we need to perform marginalization grows dramatically (i.e., parameters at all hierarchical levels enter the computation). These computational demands render the probabilistic comparison of HMs based on BFs or PMPs analytically intractable even for relatively simple models with explicit (analytical) likelihoods. Therefore, researchers need to resort to costly, approximate methods which typically only work for models with explicit likelihoods (Gelman & Meng, 1998; Gronau, Sarafoglou, et al., 2017; Meng & Schilling, 2002).

Approximate BMC

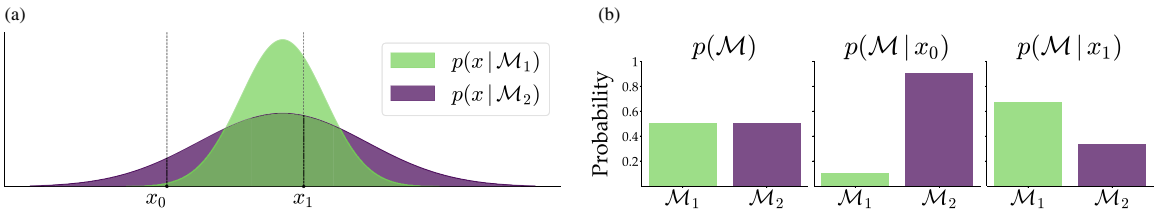
Explicit Likelihoods

The most efficient approximate methods to date require all candidate models to possess explicitly available likelihood functions. For the most simple scenario in which two HMs are nested (e.g., through an equality constraint on a parameter), the Savage–Dickey density ratio (Dickey & Lientz, 1970) provides a convenient approximation of the BF (Wagenmakers et al., 2010). Typically, however, the candidate models are not nested but exhibit notable structural differences. Thus, a general-purpose method is needed to encompass the entire plethora of model comparison scenarios arising in practical applications.

A more general method, and the current state-of-the-art for comparing HMs in psychological and cognitive modeling (Gronau et al.,

Figure 1

Hypothetical Bayesian Model Comparison Setting With a Simple Model $\mathcal{M}_1$ and a More Complex Model $\mathcal{M}_2$



Note. (a) Marginal likelihoods: The complex model which accounts for a broader range of observations needs to spread its marginal likelihood to cover its larger generative scope. It does so at the cost of diminished sharpness. Thus, even though observation x_1 is well within its generative scope, the simpler model $\mathcal{M}_1$ yields a higher marginal likelihood and is therefore preferred. In contrast, observation x_0 has a higher marginal likelihood under model $\mathcal{M}_2$, as it is very unlikely to be generated by the simpler model $\mathcal{M}_1$. (b) The corresponding posterior model probabilities given a uniform model prior. See the online article for the color version of this figure.

2020, 2019; Schad et al., 2023), is given by bridge sampling (Bennett, 1976; Meng & Wong, 1996). Bridge sampling has enabled comparisons within families of complex process models, such as multinomial processing trees (MPTs; Gronau et al., 2019) or evidence accumulation models (EAMs; Gronau et al., 2020), and serves as a simple add-on for Markov chain Monte Carlo (MCMC)-based Bayesian workflows.

Crucially, bridge sampling relies on the posterior draws generated by an MCMC sampler (e.g., Stan; Carpenter et al., 2017) to efficiently approximate the marginal likelihood of each respective model (Gronau, Sarafoglou, et al., 2017). Note, however, that bridge sampling requires considerably more random draws for stable results than standard parameter estimation (usually about an order of magnitude more; Gronau, Singmann, et al., 2017). Moreover, the approximation quality of bridge sampling is dependent on the convergence of the MCMC chains (Gronau et al., 2020). Finally, there are no strong theoretical guarantees that the approximations are unbiased and accurately reflect the true marginal likelihoods (Schad et al., 2023).

Implicit Likelihoods

With the rise of complex, high-resolution models, intractable likelihood functions (i.e., functions that do not admit a closed form or are too costly to evaluate) become more and more common in statistical modeling. Such models are not limited to psychology and cognitive science (Nicenboim et al., 2022; Van Rooij et al., 2019), but are also common in fields such as neuroscience (Gonçalves et al., 2020), epidemiology (Radev, Graw, et al., 2021), population genetics (Pudlo et al., 2016), or astrophysics (Hermans et al., 2021). Despite the common term likelihood-free, simulator-based models still possess an implicitly defined likelihood (see Bayesian Model Comparison section) from which we can obtain random draws through Monte Carlo simulations. This enables model comparison through simulation-based methods, usually by means of approximate Bayesian computation (ABC; Marin et al., 2018; Mertens et al., 2018; Pudlo et al., 2016).

Traditional (rejection-based) ABC methods for BMC repeatedly simulate data sets from the specified generative models, retaining only those simulations that are sufficiently similar to the empirical data. To enable the calculation of this (dis-)similarity even in high-dimensional cases, the information contained in the simulated data sets is reduced by computing hand-crafted summary statistics, such as the mean and variance (Csilléry et al., 2010; Sunnåker et al., 2013). The resulting acceptance rates of the candidate models represent the approximations of their PMPs (Marin et al., 2018; Mertens et al., 2018).

Even for nonhierarchical models, ABC methods are known to be notoriously inefficient and highly dependent on the concrete choice of summary statistics (Cranmer et al., 2020; Marin et al., 2018). This choice is even more challenging for HMs, as modelers now have to retain an optimal amount of information on multiple levels. Moreover, the rapidly growing number of summary statistics reduces the probability that a simulated data set is similar enough to the empirical data, which vastly increases the number of required simulations (Beaumont, 2010; Marin et al., 2018).

Regardless of the number of summary statistics, their manual computation carries the danger of insufficiently summarizing the simulations and thereby producing biased approximations (a phenomenon known as the curse of insufficiency; Marin et al., 2018). While many improvements of rejection-based ABC have been proposed, most notably ABC-MCMC (Marjoram et al., 2003; Turner & Sederberg, 2014), ABC-SMC (Sisson et al., 2007), as well as Gibbs ABC (Turner & Van Zandt, 2014) for Bayesian hierarchical modeling in particular (see also G. Clarté et al., 2021; Fengler et al., 2021), these advancements are still limited by their dependence on hand-crafted summary statistics or kernel density estimation methods.

Recent developments, such as ABC-RF (Pudlo et al., 2016), combine ABC with machine learning methods to build more expressive approximators for BMC problems. Accordingly, model comparison is treated as a supervised learning problem—the simulated data encompasses a training set for a machine learning algorithm that learns to recognize the true generative model from which the data set was simulated. The machine learning approach reduces the inefficiency problem that haunts rejection-based ABC methods, but does not alleviate the curse of insufficiency (Marin et al., 2018).

BMC With Neural Networks

Recently, Radev, D'Alessandro, et al. (2021) explored a method for simulation-based BMC using specialized neural networks. The authors proposed to jointly train two specialized neural networks using Monte Carlo simulations from each candidate model in $\mathcal{M}$: a summary network and an evidential network. The goal of the summary network is to extract maximally informative (in the optimal case, sufficient) summary statistics from complex data sets. The goal of the evidential network is to approximate PMPs as accurately as possible and, optionally, to quantify their epistemic uncertainty.

Importantly, simulation-based training of neural networks enables amortized inference for both implicit and explicit likelihood models. Amortization is a property that ensures rapid inference for an arbitrary amount of data sets after a potentially high computational investment in simulation and training (Mestdagh et al., 2019; Radev, D'Alessandro, et al., 2021; Radev et al., 2020). As a consequence, the calibration (Guo et al., 2017; Talts et al., 2018) or the inferential adequacy (Schad et al., 2021, 2023) of an amortized Bayesian method are embarrassingly easy to validate in practice.

In contrast, nonamortized methods, such as ABC-MCMC (Turner & Sederberg, 2014) or ABC-SMC (Sisson et al., 2007) need to repeat all computations from scratch for each observed data set. Thereby, it is often infeasible to assess their calibration or inferential adequacy in the predata phase of a Bayesian workflow (Gelman et al., 2020).

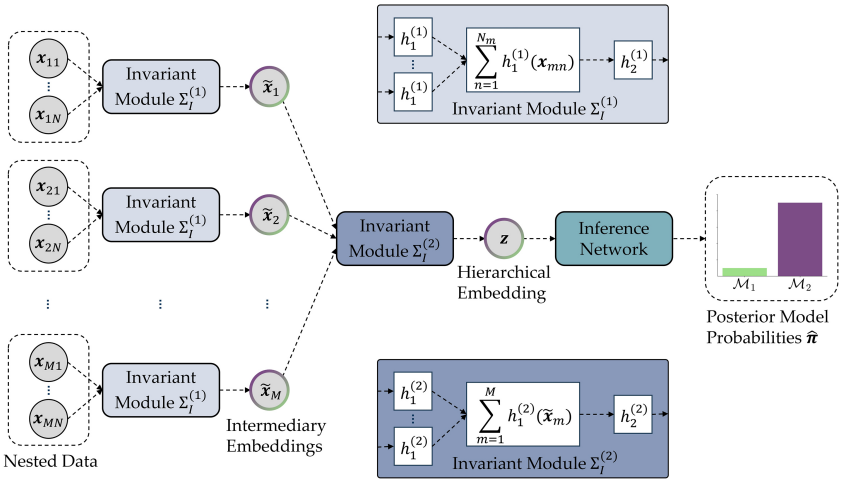
Unfortunately, the evidential method proposed by Radev, D'Alessandro, et al. (2021) is not applicable to HMs due to their nested probabilistic structure which cannot be tackled via previous summary networks. This severely limits the applicability of the method in quantitative research, where HMs have been advocated as a default choice (Lee, 2011; McElreath, 2020; Rouder et al., 2017). In the following, we describe how to extend the original method to enable amortized BMC for HMs.

Method

At its core, our method involves a multilevel permutation invariant neural network which is aligned to the probabilistic symmetry of the underlying HMs (see Figure 2 for a visualization). We hold that any method which does not rely on ad hoc summary statistics should

Figure 2

Our Proposed Hierarchical Neural Network Architecture for Encoding Permutation Invariance in the Transformation of Nested, Two-Level Data Into Posterior Model Probabilities



Note. A first invariant module $\Sigma_I^{(1)}$ reduces all N_m observations within each of the M groups to a single intermediary embedding vector $\tilde{\boldsymbol{x}}_m$. For readability of the figure, we display $N_m = N$ as constant across each group. A second invariant module $\Sigma_I^{(2)}$ reduces all intermediary embedding vectors to a hierarchical embedding vector $\boldsymbol{z}$, which gets passed through an inference network to arrive at the final vector $\hat{\boldsymbol{\pi}}$ of approximated posterior model probabilities. See the online article for the color version of this figure.

take this probabilistic symmetry (e.g., exchangeability) into account in order to ensure the structural faithfulness of its approximations. Moreover, respecting the probabilistic symmetry implied by a generative model can not only make simulation-based training easier but also suggests a particular architecture for building neural Bayesian approximators.

Permutation Invariance

Permutation invariance is the functional equivalent of the probabilistic notion of exchangeability (Bloem-Reddy & Teh, 2020; Gelman, 2006), which roughly states that the order of random variables should not influence their joint probability.

To illustrate this point, consider the model in Equation 4, which has two exchangeable levels by design, indexed by $m \in \{1, \ldots, M\}$ and $n \in \{1, \ldots, N_m\}$. In a setting familiar to social scientists, we might have M individuals, each of whom provides N_m (multivariate) responses on some scale or in repeated trials of an experiment. Now, suppose that we want to compare a set of HMs $\mathcal{M} = \{\mathcal{M}_1, \ldots, \mathcal{M}_J\}$ of the form given by Equation 4 that might differ in various ways (e.g., different prior/hyperprior assumptions or disparate likelihoods). Due to the structure of the models, the PMPs $p(\mathcal{M}|\{\boldsymbol{x}_{mn}\})$ depend on neither the ordering of the individuals nor the ordering of their responses (which also holds true for the corresponding BFs).

More precisely, if $\mathbb{S}(\cdot)$ is an arbitrary permutation of an index set, then

$$p(\mathcal{M}|\{\boldsymbol{x}_{mn}\}) = p(\mathcal{M}|\mathbb{S}(\{\boldsymbol{x}_{mn}\})), \tag{11}$$

for any $\mathbb{S}(\cdot)$ acting on $\{1, \ldots, M\} \times \{1, \ldots, N_1\} \times \cdots \times \{1, \ldots, N_M\}$, where $\times$ denotes the Cartesian product of two (index) sets. Note that this notation implies that only permuting each m and permuting each n within, but not across each group m is allowed. The property of permutation invariance is immediately obvious from the right-hand side of Equation 4 that involves two nested products (products being permutation invariant transformations when seen as functions operating on sets). Naturally, learning permutation invariance directly from data or simulations is hardly feasible with standard neural networks, even for nonnested data. Indeed, for nonhierarchical generative models, Radev, D'Alessandro, et al. (2021) proposed to use composite permutation invariant networks as employed by Zaheer et al. (2017). In the following section, we generalize this architectural concept to the hierarchical setting.

Hierarchical Invariant Neural Network Architecture

Permutation invariant networks differ from standard feedforward networks in that they can process inputs of different sizes and encode the probabilistic symmetry of the data directly (i.e., remove the need to learn the symmetry implicitly during training by supervised learning alone).

For the purpose of BMC with HMs, we realize a hierarchical permutation invariant function via a stack of invariant modules $\Sigma_I^{(l)}$ for each hierarchical level $l = 1, \ldots, L$ of the Bayesian model (see Figure 2). Each invariant module performs an equivariant nonlinear transformation $h_1^{(l)}$ acting on the individual data points, followed by a pooling operator (e.g., sum or max) and a further nonlinear transformation $h_2^{(l)}$ acting on the pooled data.

In order to preserve hierarchical symmetry, we apply each $\Sigma_I^{(l)}$ independently to each nested sequence of data points. To make this point concrete, consider the two-level model given by Equation 4 and let data point $\boldsymbol{x}_{mn}$ denote the multivariate response

of person m in trial n of some data collection experiment. Accordingly, the first invariant module $\Sigma_I^{(1)}$ operates by reducing the trial data $\{\boldsymbol{x}_n\}_m$ of each person m to a single person-vector $\widetilde{\boldsymbol{x}}_m$ of fixed size:

$$\widetilde{\boldsymbol{x}}_m = \Sigma_I^{(1)}\left(\{\boldsymbol{x}_n\}_m\right) = h_2^{(1)}\left(\sum_{n=1}^{N_m} h_1^{(1)}(\boldsymbol{x}_{mn})\right), \tag{12}$$

where h_1 and h_2 are implemented as simple feedforward neural networks with trainable parameters suppressed for clarity. The second invariant module $\Sigma_I^{(2)}$ then compresses all person vectors to a final vector $\boldsymbol{z}$ of fixed size:

$$\boldsymbol{z} = \Sigma_I^{(2)}(\{\boldsymbol{x}_m\}) = h_2^{(2)}\left(\sum_{m=1}^{M} h_1^{(2)}(\widetilde{\boldsymbol{x}}_m)\right). \tag{13}$$

In this way, the architecture becomes completely independent of the number of persons M or number of trials per person N_m, which could vary arbitrarily across persons. The vector $\boldsymbol{z}$, whose dimensionality represents a tunable hyperparameter, can be interpreted as encoding learned summary statistics for the BMC task at hand (to be discussed shortly). Moreover, it is easy to see that $\boldsymbol{z}$ is independent of the ordering of persons or the ordering of trials within persons, as necessitated by the model formulation in Equation 4. Thus, the composition $\Sigma_I^{(2)} \circ \Sigma_I^{(1)}(\{\boldsymbol{x}_{mn}\})$ reduces a hierarchical data set with two levels to a single vector $\boldsymbol{z}$ which respects the probabilistic symmetry implied by the particular HM formulation.

Increasing the Capacity of Invariant Networks

Encoding an entire hierarchical data set $\{\boldsymbol{x}_{mn}\}$ into a single vector $\boldsymbol{z}$ forces the composite neural network to perform massive data compression, creating a potential information bottleneck. For complex generative models, this task can become rather challenging and will depend highly on the representational capacity of the neural network (i.e., its ability to extract informative data set embeddings). Fortunately, we can enhance the simple architecture described in the preceding paragraph by using insights from Zaheer et al. (2017) and Bloem-Reddy and Teh (2020).

In order to increase the capacity of the previously introduced invariant transformation, we can stack together multiple equivariant modules $\Sigma_E^{(l)}$. Each equivariant module implements a combination of equivariant and invariant transformations. For instance, focusing on our two-level model example (Equation 4), the transformations at level 1 for each person m are now given by:

$$\widetilde{\boldsymbol{x}}_m = h_2^{(1)}\left(\sum_{n=1}^{N_m} h_1^{(1)}(\boldsymbol{x}_{mn})\right), \tag{14}$$

$$\widetilde{\boldsymbol{x}}_{mn} = h_3^{(1)}\left(\left[\boldsymbol{x}_{mn}, \widetilde{\boldsymbol{x}}_m\right]\right) \quad \text{for } n = 1, \ldots, N_m, \tag{15}$$

where h_3 is also implemented as a simple feedforward neural network. In this way, each intermediary output $\widetilde{\boldsymbol{x}}_{mn}$ of the equivariant module now contains information from all data points, so the network can learn considerably more flexible transformations. Moreover, we can stack K equivariant modules followed by an invariant module, in order to obtain a deep invariant module, which for the first hierarchical level ($l = 1$) takes the following form:

$$\widetilde{\boldsymbol{x}}_m = (\Sigma_I^{(1)} \circ \Sigma_E^{(K,1)} \circ \cdots \circ \Sigma_E^{(1,1)})(\{\boldsymbol{x}_n\}_m). \tag{16}$$

Compared to the simple invariant module from Equation 12, the deep invariant module involves a larger number of computations but allows the network to learn more expressive representations. Accordingly, the transformation for the second hierarchical level ($l = 2$), which yields the final summary representation $\boldsymbol{z}$, is given by:

$$\boldsymbol{z} = (\Sigma_I^{(2)} \circ \Sigma_E^{(K',2)} \circ \cdots \circ \Sigma_E^{(1,2)})(\{\widetilde{\boldsymbol{x}}_m\}), \tag{17}$$

where the number of equivariant modules K' for level 2 can differ from the number of equivariant modules K for level 1. In our experiments, reported in the Experiments section, we observe a clear advantage of using deep invariant networks over their simple counterparts. Furthermore, for two-level models, we find that the performance of the networks is largely insensitive to the choice of K or K'.

Learning the Model Comparison Problem

In order to get from the learned summary representation $\boldsymbol{z}$ to an approximation of the analytic PMPs $\widehat{\boldsymbol{\pi}}$, we apply a final neural classifier (i.e., the inference network) $\mathcal{I}(\boldsymbol{z}) = \widehat{\boldsymbol{\pi}}$, as visualized in Figure 2. We deviate from the Dirichlet-based setting by Radev, D'Alessandro, et al. (2021), since we found that implementing the inference network as a standard softmax classifier (Grathwohl et al., 2019) provides slightly better calibration and leads to more stable training in the specific context of HMs.

Denoting the entire hierarchical neural network as $f_{\boldsymbol{\phi}}(\{\boldsymbol{x}\}) = \widehat{\boldsymbol{\pi}}$ and an arbitrary hierarchical data set as $\{\boldsymbol{x}\}$, we aim to minimize the expected logarithmic loss

$$\min_{\boldsymbol{\phi}} \mathbb{E}_{p(\mathcal{M},\{\boldsymbol{x}\})}\left[-\sum_{j=1}^{J} \mathbb{I}_{\mathcal{M}_j} \cdot \log f_{\boldsymbol{\phi}}(\{\boldsymbol{x}\})_j\right], \tag{18}$$

where $\boldsymbol{\phi}$ represents the vector of trainable neural network parameters (e.g., weights and biases), $\mathbb{I}_{\mathcal{M}_j}$ is the indicator function for the "true" model. The expectation runs over the joint generative (mixture) distribution of all models $p(\mathcal{M}, \{\boldsymbol{x}\})$, which we access through Monte Carlo simulations. Since the logarithmic loss is a strictly proper loss (Gneiting & Raftery, 2007), it drives the outputs of $f_{\boldsymbol{\phi}}(\{\boldsymbol{x}\})$ to estimate the actual PMPs $p(\mathcal{M}|\{\boldsymbol{x}\})$ as best as possible. Thus, perfect convergence in theory guarantees that the network outputs the analytically correct PMPs which asymptotically select the "true" model in the closed world or the model that minimizes the Kullback-Leibler divergence to the "true" data generating process in the open world (Barron et al., 1999).

In practice, we approximate Equation 18 over a training set of B simulations from the competing HMs. Each entry b for $b = 1, \ldots, B$ in this training set represents a hierarchical data set $\{\boldsymbol{x}^{(b)}\}$ itself along with a corresponding one-hot encoded vector for the "true" model index $\mathcal{M}_j^{(b)}$. The latter denotes the model from which the data set was generated and serves as the "ground truth" for supervised learning.

Similarly to Radev, D'Alessandro, et al. (2021), our neural method encodes an implicit preference for simpler HMs (i.e., Occam's razor) inherent in all marginal likelihood-based methods

(see MacKay, 2003, Chapter 28). Since our simulation-based training approximates an expectation over the marginal likelihoods of all HMs $p(\mathcal{M})p(\boldsymbol{x}|\mathcal{M})$, data sets generated by a simpler HM will tend to be more similar compared to those generated by a more complex one (cf. Figure 1). Thus, data sets that are plausible under both HMs will be generated more often by the simpler model than by the more complex model. A sufficiently expressive neural network will capture this behavior by assigning a higher PMP for the simpler model,[1] thereby capturing complexity differences arising directly from the generative behavior of the HMs.

Finally, to increase training efficiency when working under a limited simulation budget, we also explore a novel pretraining method inspired by transfer learning (Bengio et al., 2009; Torrey & Shavlik, 2010). First, we train the networks on data sets with a reduced number of exchangeable units (e.g., reducing the number of observations at level $l = 1$). This procedure accelerates training since it uses fewer simulator calls and the forward pass through the networks becomes cheaper. In the second step, we generate data with a realistic number of exchangeable units. Crucially, since we can use the pretrained network from step one as a better-than-random initialization, we need considerably fewer simulations than if we trained the network from scratch. Indeed, our Application: Hierarchical Evidence Accumulation Models section demonstrates the utility of this training method.

Experiments

In this section, we first conduct two simulation studies in which we extensively test the approximation performance of our hierarchical neural method. We start with a comparison of two nested toy HMs in the Validation Study 1: Hierarchical Normal Models section, followed by a comparison of two complex nonnested HMs of cognition in the Validation Study 2: Hierarchical SDT Versus MPT Models section. For both validation studies, we test our method internally by examining the calibration of the approximated PMPs. Additionally, we validate our method externally by benchmarking its performance against the current state-of-the-art for comparing HMs, namely, bridge sampling (Gelman & Meng, 1998; Gronau, Singmann, et al., 2017). To enable this challenging benchmark, we limit our validation studies to the comparison of models with explicit likelihoods to which bridge sampling is applicable.

Finally, in the Application: Hierarchical Evidence Accumulation Models section, we use our deep learning method to compare four hierarchical EAMs of response times data. Two of these models have no analytic likelihood, which makes the entire BMC setup intractable with current state-of-the-art methods (e.g., bridge sampling). Moreover, with this example, we also address the utility of a novel EAM, the Lévy flight model (Voss et al., 2019), that has previously been impossible to investigate directly using Bayesian HMs.

For all experiments, we assume uniform model priors $p(\mathcal{M}_j) = 1/J$. All computations are performed on a single-graphics processing unit machine with an NVIDIA RTX 3,070 graphics card and an AMD Ryzen 5 5600X processor. The reported computation times are measured as wall-clock times. Details on the implementation of our neural networks and the employed training procedures are provided in Appendix A. Code for reproducing all results from this article is freely available at https://github.com/bayesflow-org/Hierarchical-Model-Comparison. Additionally, our proposed method is implemented in the BayesFlow Python library for amortized Bayesian workflows (Radev et al., 2023).

Validation Study 1: Hierarchical Normal Models

In this first experiment, we examine a simple and controllable model comparison setup to examine the behavior of our method under various conditions, before moving on to more complex scenarios. Inspired by Gronau (2021), we compare two hierarchical normal models $\mathcal{M}_1$ and $\mathcal{M}_2$ that share the same hierarchical structure

$$\tau^2 \sim \text{Normal}_+(0, 1), \tag{19}$$

$$\sigma^2 \sim \text{Normal}_+(0, 1), \tag{20}$$

$$\theta_m \sim \text{Normal}(\mu, \sqrt{\tau^2}) \quad \text{for } m = 1, \ldots, M, \tag{21}$$

$$x_{mn} \sim \text{Normal}(\theta_m, \sqrt{\sigma^2}) \quad \text{for } n = 1, \ldots, N_m, \tag{22}$$

with $\text{Normal}_+(\cdot)$ denoting a zero-truncated normal distribution. The models differ with respect to the parameter μ that describes the location of the individual-level parameters θ_m: whereas $\mathcal{M}_1$ assumes the location of θ_m to be fixed at 0, the more flexible $\mathcal{M}_2$ allows for μ to vary

$$\mathcal{M}_1 : \mu = 0, \tag{23}$$

$$\mathcal{M}_2 : \mu \sim \text{Normal}(0, 1). \tag{24}$$

Calibration

The most important properties of an approximate inference method are the trustworthiness of its results and, more pragmatically, whether we can diagnose the lack of trustworthiness in a given application. A useful proxy for trustworthiness is the calibration of a probabilistic classifier, which measures how closely the predicted probabilities of outcomes match their true underlying probabilities (Guo et al., 2017; Schad et al., 2023).

However, computing the calibration of a BMC procedure is hardly feasible in a nonamortized setting, since it involves applying the method to a large number of simulated data sets. For bridge sampling, for example, that would imply re-fitting the models via MCMC and running bridge sampling on at least hundreds, if not thousands of simulated data sets. The calibration of our networks, on the other hand, can be determined almost immediately after training due to their amortization property (Radev, D'Alessandro, et al., 2021).

In the following experiments, we assess the calibration of our networks visually (via calibration curves) and numerically (via a measure of calibration error). For generating a calibration curve (DeGroot & Fienberg, 1983; Niculescu-Mizil & Caruana, 2005), we first sort the predicted PMPs $\widehat{\pi}_j^{(s)}$ on S simulated data sets $s = 1, \ldots, S$, which we then partition into I equally spaced probability bins $i = 1, \ldots, I$ (we use $I = 15$ bins for all validation experiments). For each model j and each bin i containing a set $\mathcal{B}_{ij}$ of predicted model indices, we compute the mean prediction for the model (predicted probability [PP]) and the actual fraction of this model being

[1] Assuming equal prior model probabilities.

true (true probability[TP]) as follows:

$$\mathrm{PP}(\mathcal{B}_{ij}) := \frac{1}{|\mathcal{B}_{ij}|} \sum_{b \in \mathcal{B}_{ij}} \widehat{\pi}_j^{(b)}, \tag{25}$$

$$\mathrm{TP}(\mathcal{B}_{ij}) := \frac{1}{|\mathcal{B}_{ij}|} \sum_{b \in \mathcal{B}_{ij}} \mathbb{I}_{\mathcal{M}_j^{(b)}}, \tag{26}$$

where $\mathbb{I}$ again denotes the indicator function for the "true model." These two quantities varying over the bins form the *X*- and *Y*-axis of a calibration curve. A well-calibrated model comparison method with an agreement in each bin (as indicated by a diagonal line) thus yields approximations that reflect the true probabilities of the compared models (Guo et al., 2017). We further summarize this information via the expected calibration error (ECE; Naeini et al., 2015) as a single number bounded between 0 and 1, which we estimate by averaging the individual deviations between PP and TP in each bin:

$$\widehat{\mathrm{ECE}}_j := \sum_{i=1}^{I} \frac{|\mathcal{B}_{ij}|}{S} \left| \mathrm{PP}(\mathcal{B}_{ij}) - \mathrm{TP}(\mathcal{B}_{ij}) \right|. \tag{27}$$

If follows from Equation 27 that a perfect ECE can be achieved by always predicting indifferent probabilities (e.g., $\widehat{\pi}_1 = \widehat{\pi}_2 = .5$ when comparing two models). We therefore complement our calibration assessment by measuring the accuracy of recovery, for which we dichotomize the predicted PMPs $\widehat{\pi}_j^{(s)}$ on *S* simulated data sets into one-versus-rest model predictions $\widehat{\mathcal{M}}_j^{(s)}$:

$$\mathrm{Acc}_j := \frac{1}{S} \mathbb{I}_{\widehat{\mathcal{M}}_j^{(s)} = \mathcal{M}_j^{(s)}}. \tag{28}$$

Thus, in our BMC context, accuracy roughly is to ECE what sharpness is to posterior calibration in Bayesian parameter estimation (Bürkner et al., 2022; L. Clarté et al., 2022).[2]

Fixed Data Set Sizes. In the first calibration experiment, we examine the performance of our method for the most simple application case of learning a model comparison problem on a specific (fixed) data set size. Here, all data sets simulated for training and validating, the network consist of $M = 50$ groups and $N_m = 50$ observations for each group $m = 1, \ldots, M$.

We train the network for 10,000 backpropagation steps, taking 6 min. Subsequently, we calculate its calibration on 5,000 held-out validation data sets and repeat this process 25 times to obtain stable results with uncertainty quantification. Figure 3a depicts the resulting median calibration curve. Its close alignment to the dashed diagonal line (representing perfect calibration) indicates that the approximate PMPs are well-calibrated (median ECE over all repetitions of $\widehat{\mathrm{ECE}} = 0.014$). The curve's coverage of the full range of PPs and the median accuracy of $\widehat{\mathrm{Acc}} = 0.89$ confirm that the excellent calibration does not stem from indifferent predictions. The subsequent comparison of our method to bridge sampling suggests that this accuracy is indeed close to the upper bound imposed by the aleatoric uncertainty in the model-implied data.

Data Sets With Varying Numbers of Observations. We now train our hierarchical network to approximate BMC over a range of hierarchical data sets with varying number of observations within groups N_m. This amortization over observation sizes would provide a substantial efficiency gain if a researcher desires to compare HMs on multiple data sets with differing N_m, as only a single network would have to be trained for all data sets.[3] In our validation setup, each simulated data set still consists of $M = 50$ groups, but now the number of observations within those groups varies in $N_m = 1, \ldots, 100$.

We train the network for 20,000 training steps, taking 13 min. At each training step, we draw the number of observations for the current batch of simulations from a discrete uniform distribution $N_m \sim \mathrm{Uniform}_D(1{,}100)$. For each N_m used during training, we evaluate the calibration 25 times on 5,000 held-out simulated validation data sets. This repetition procedure allows us to quantify the uncertainty of our ECE estimates.

Figure 3b plots the median ECE values for each observation size. The neural network achieves high calibration with a median ECE over all observation sizes (and repetitions) of $\widehat{\mathrm{ECE}} = 0.012$. Moreover, the unsystematic pattern of the median curve and the homoscedastic variation between the observation sizes indicate that the network has learned the model comparison task equally well for all settings (with the ECE only rising slightly for the poorly identifiable $N_m = 1$ setting). Together, the low calibration error and the accurate model predictions (median accuracy $\widehat{\mathrm{Acc}} = 0.88$) indicate that our method incurs no trade-off between calibration and accuracy. We additionally observe no bias towards a model in all but the smallest observation sizes (see Figure B1 for accuracy and bias examinations in all settings).

Data Sets with Varying Numbers of Groups and Observations. In the third calibration experiment, we test the ability of the network to learn a model comparison problem over a range of data sets with varying number of groups *M* and varying observations per group N_m. This training scheme allows for amortized model comparison on multiple data sets with different sizes, which can be especially useful for a priori sample size determination on simulated data. Additionally, the trained network can be stored and reused on future data sets with yet-unknown sample sizes. For this experiment, training and validation data sets are simulated with $M = 1, \ldots, 100$ groups and $N_m = 1, \ldots, 100$ observations, resulting in a vast variability of data set sizes between 1 up to 10,000 data points.

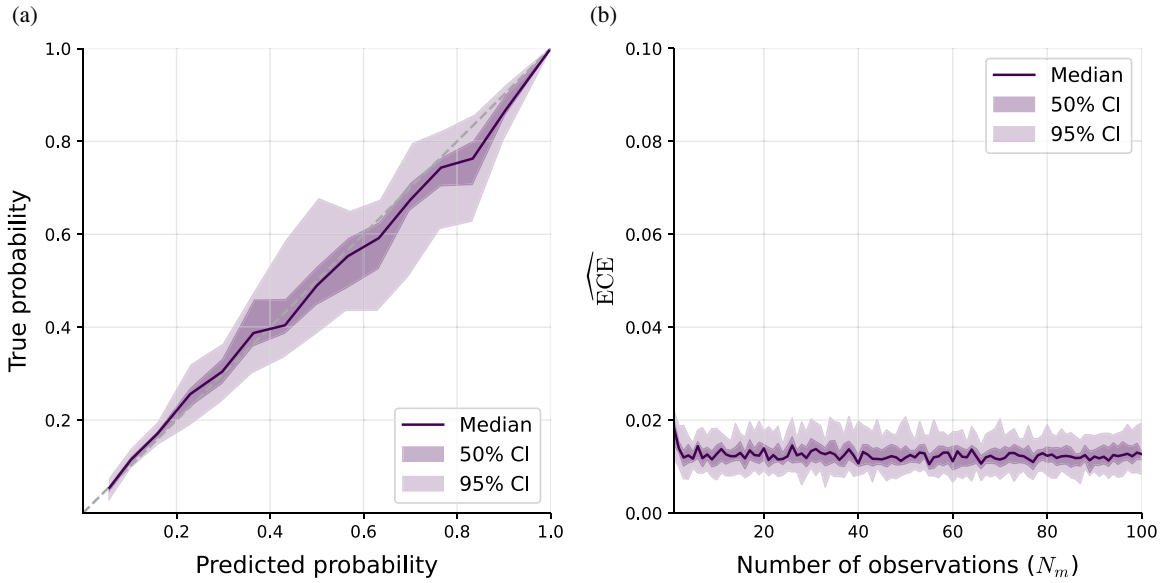
Given the complexity of the learning task, we now train the network for 40,000 training steps, taking 36 min. At each training step, we draw the number of groups and observations from discrete uniform distributions $M \sim \mathrm{Uniform}_D(1{,}100)$ and $N_m \sim \mathrm{Uniform}_D(1{,}100)$. We estimate calibration on 5,000 held-out simulations for each combination of *M* and N_m. As this implies simulating 50,000,000 data sets, we forego the repetition procedure employed in the previous experiments.

Figure 4 depicts the calibration and accuracy results for all combinations of *M* and N_m. We observe low ECEs for the vast majority of settings in Figure 4a (median ECE over all settings of $\widehat{\mathrm{ECE}} = 0.013$). In other words, the trained network is capable of generating highly calibrated PMPs over a broad range of data set sizes. Moreover, the BMC results are sensitive to the number of nested observations N_m, but not to the number of groups *M*, in our experimental setups. The only systematic drop in calibration occurs for data sets containing just a few nested observations

[2] We focus on the accuracy since we use a uniform model prior $p(\mathcal{M})$, but other metrics of predictive performance, such as the logarithmic scoring rule, would have been expedient as well.

[3] Note that we refer to variability between data sets. We describe an approach for handling within data set variability of nested trials in the Application: Hierarchical Evidence Accumulation Models section.

Figure 3
Validation Study 1: Calibration Results for the First Two Experiments



Note. (a) Results for the neural network trained on fixed data set sizes: Median calibration curve and CIs for data sets of $M = 50$ groups with $N_m = 50$ observations within each group. (b) Results for the neural network trained on data sets with the varying number of observations: Median ECEs and CIs for data sets of $M = 50$ groups with differing numbers of observations N_m within each group. Medians and CIs for all results are computed over 25 repetitions. CI = confidence interval; ECE = expected calibration error. See the online article for the color version of this figure.

($N_m \leq 5$). Considering that we observed better calibration for this low number of observations in a network trained on data sets with varying N_m (see Figure 3b), we surmise that the drop in the edge areas in Figure 4a arises from the challenging learning task over vastly different data set sizes (a phenomenon known as amortization gap; Cremer et al., 2018). The overall low (i.e., good) ECEs for all cases but the poorly identifiable $N_m = 1$ setting suggest that the networks' approximations are generally trustworthy. This is further confirmed in Figure 4b, where the observable accuracy pattern assures that this high calibration does not arise from a trade-off with predictive performance. Despite the demanding amortization setting, the network achieves an excellent median accuracy of $\widehat{\text{Acc}} = 0.88$, similar to the earlier experiments. We also find no indication of bias in any of the test settings except the $N_m = 1$ setting (see Figure B2). Marginal diagnostic plots for all metrics are provided in Figure B3.

Bridge Sampling Comparison

After validating the general trustworthiness of our method, we now benchmark it against the current gold standard for comparing HMs, namely, bridge sampling, as implemented by Gronau, Singmann, et al. (2017). As the nonamortized nature of bridge sampling restricts the feasible number of test sets, we conduct the benchmarking on 100 test sets which are simulated equally from $\mathcal{M}_1$ and $\mathcal{M}_2$. All simulated data sets consist of $M = 50$ groups and $N_m = 50$ observations per group. The fixed sample sizes of the test sets allow us to compare the two most distinct networks from the Calibration section to bridge sampling: the fixed network that is trained for this specific sample size and the more complex variable network that is trained for amortized model comparison over variable sample sizes between $M = 1, \ldots, 100$ groups and $N_m = 1, \ldots, 100$ observations per group.

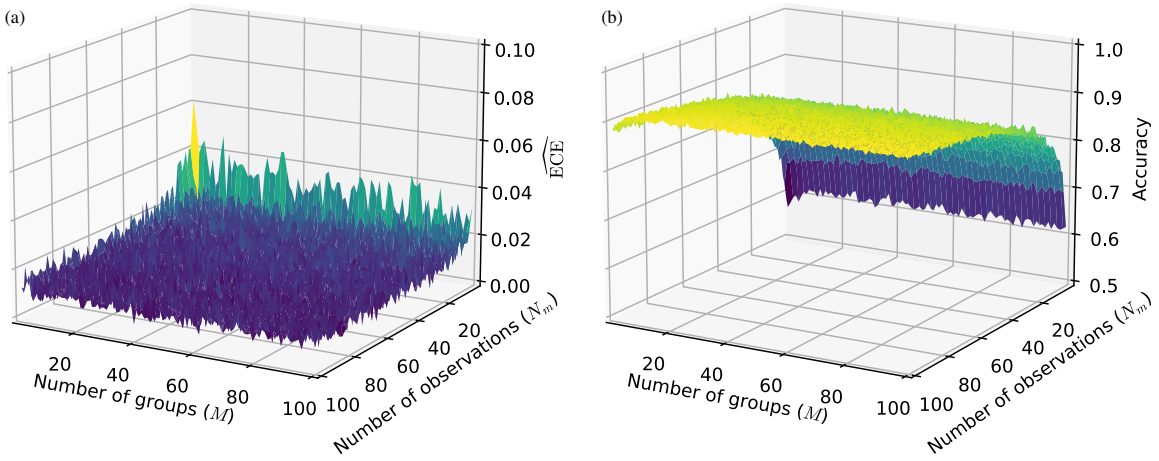
For bridge sampling, we first run four parallel MCMC chains with a warm-up period of 1,000 draws and 49,000 postwarm-up posterior draws per chain in Stan (Carpenter et al., 2017; Stan Development Team, 2019). We assess convergence through a visual inspection of the MCMC chains and an assessment of the $\widehat{R}$, bulk ESS, and tail ESS metrics (Vehtari et al., 2021). Afterward, we use the posterior draws to approximate PMPs and BFs with the bridgesampling R package (Gronau, Singmann, et al., 2017). We confirm the sufficiency of the total of 196,000 posterior draws by assessing the variability between multiple runs as by Schad et al. (2023), which yields highly similar results. Further insights via our calibration diagnostics are precluded by bridge sampling being a nonamortized method.

Approximation Performance. As we compare approximate PMPs, we can use a number of complementary metrics commonly employed to evaluate the quality of probabilistic predictions. First, we quantify the fraction of times the correct model $\mathcal{M}_j^{(s)}$ underlying a simulated data set s was detected, that is, the accuracy of recovery (see Equation 28). Second, we assess the mean absolute error (MAE) to investigate the average deviation of the approximated model probabilities $\widehat{\pi}_j^{(s)}$ from a perfect classification:

$$\text{MAE}_\text{j} := \frac{1}{\text{S}} \sum_{\text{s}=1}^{\text{S}} \left| \widehat{\pi}_\text{j}^{(\text{s})} - \mathbb{I}_{\mathcal{M}_\text{j}}^{(\text{s})} \right|. \tag{29}$$

Figure 4

Validation Study 1: Results for the Neural Network Trained and Tested Over Variable Data Set Sizes



Note. (a) ECE and (b) accuracy of recovery. ECE = expected calibration error. See the online article for the color version of this figure.

Third, we measure the root-mean-square error (RMSE), which places particular emphasis on large prediction errors, to detect whether one method produces highly incorrect approximations more frequently than the other:

$$\text{RMSE}_j := \sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(\widehat{\pi}_j^{(s)} - \mathbb{I}_{\mathcal{M}_j}^{(s)}\right)^2}. \tag{30}$$

Fourth, we calculate the Log-Score following the logarithmic scoring rule:

$$\text{LogScore}_j := -\frac{1}{S}\sum_{s=1}^{S}\left[\mathbb{I}_{\mathcal{M}_j}^{(s)} \cdot \log \widehat{\pi}_j^{(s)}\right]. \tag{31}$$

Its property as a strictly proper scoring rule implies that it is asymptotically minimized if and only if the approximate probabilities equal the true probabilities (Gneiting & Raftery, 2007). Lastly, we measure simulation-based calibration (SBC; Talts et al., 2018) as adapted by Schad et al. (2023) for model inference by the difference between the prior probability for a model and its average posterior probability in the test sets:

$$\text{SBC}_j := p(\mathcal{M}_j) - \frac{1}{S}\sum_{s=1}^{S}\widehat{\pi}_j^{(s)}. \tag{32}$$

We evaluate all metrics for $\mathcal{M}_2$ so that a bias towards $\mathcal{M}_1$ is indicated by positive SBC values and a bias towards $\mathcal{M}_2$ by negative SBC values.

Table 1 depicts the comparison results for our experimental setting. All metrics show equal performances for bridge sampling and the two neural network variants, with any differences being well within the range of the standard errors.

Approximation Convergence. In the following, we analyze the degree of convergence between the two methods at the level of individual data sets. We explore this visually by contrasting the PMP and (natural logarithmic) BF approximations of bridge sampling with the two neural network variants in Figure 5. We observe that the two methods' PMP approximations agree for the easy cases where the true underlying model is clearly classifiable. Thus, discrepancies between the two methods arise mainly for data sets with predicted PMPs close to $\widehat{\pi} = 0.5$. Even for the data sets with the largest discrepancies, the two methods do not map to qualitatively different decisions: $\widehat{\pi}_2^{(\text{bridge})} = 0.67$ and $\widehat{\pi}_2^{(\text{neural})} = 0.79$ for the fixed network, $\widehat{\pi}_2^{(\text{bridge})} = 0.32$ and $\widehat{\pi}_2^{(\text{neural})} = 0.25$ for the variable network. Most importantly, we detect no systematic pattern in these deviations.

As BFs represent the ratio of marginal likelihoods, they allow for a closer inspection of the degree of agreement between the methods in those edge cases with PMPs close to 0 or 1. We observe a close convergence for data sets classified as stemming from $\mathcal{M}_1$. Considering the predictions favoring $\mathcal{M}_2$, there are discrepancies for data sets with log BFs > 9.49. Since this corresponds to BFs $>$13,000 and PMPs > 0.9999, it is not visible in the PMP approximation plots. We obtain such extreme results only for $\mathcal{M}_2$, as this model allows for deviations of the group level parameters' location from 0 and enables the occurrence of extreme evidence in its favor. The divergence in this area of extreme evidence emerges most likely from the loss function employed for training the neural networks: the logarithmic loss obtained from a minuscule deviation of the PMP from 1 is near 0, which results in a negligible incentive for further optimization of the network's weights. We could reject a competing explanation based on limited floating-point precision, since training with an increased floating-point precision from 32-bit to 64-bit resulted in identical patterns. For visibility purposes, we exclude the 27 data sets for which bridge sampling approximated a BF $>$ 1,000,000 for the BF plots in Figure 5, all continuing the observed plateau pattern. Plots with all 100 data sets are provided in Appendix B.

Table 1
Validation Study 1: Performance Metrics for the Comparison Between Hierarchical Normal Models

	Accuracy	MAE	RMSE	Log-Score	SBC
Bridge sampling	0.86 (0.03)	0.19 (0.03)	0.32 (0.03)	0.32 (0.06)	−0.02 (0.04)
Fixed network	0.84 (0.04)	0.19 (0.03)	0.32 (0.03)	0.32 (0.06)	−0.01 (0.04)
Variable network	0.86 (0.03)	0.19 (0.03)	0.32 (0.03)	0.31 (0.06)	−0.01 (0.04)

Note. Bootstrapped mean values and standard errors (in parentheses) are presented. We use 1,000 bootstrap versions of the test data sets and estimate the standard errors from the bootstrap standard deviations of the metrics. MAE = mean absolute error; RMSE = root-mean-square error; SBC = simulation-based calibration.

The divergence we encountered provides insights into the technical nature of our method but only arises in cases of extreme evidence. Thus, it is far from altering the substantive conclusions derived from the simulated BMC setting. Considering the convergence between the two methods in the realm of practical relevance, we can conclude that our method produces highly similar approximations to bridge sampling in this scenario.

Approximation Time. Both bridge sampling and our deep learning method can be divided into two computational phases. For bridge sampling, the first phase consists of drawing from the posterior parameter distributions (taking 52 s per data set on average). Bridge sampling itself takes place in the second phase (taking 38 s on average). Notably, in contrast to amortized inference with neural networks, both phases need to be repeated for each (simulated or observed) data set. Taking the initial compilation time of 42 s into account, bridge sampling consequently took 152 min for BMC on our 100 test data sets.

For the neural networks, the first phase (training) is resource-intensive (taking 6 min for the fixed network and 36 min for the variable network). The second phase (inference) is then performed in near real-time (inference on all 100 test data sets took 0.0004 s for the fixed network and 0.007 s for the variable network) and thus amortizes the training cost over multiple applications. For the simple HMs compared here, the amortization gains of our networks over bridge sampling come into effect after performing BMC on four (fixed network) or 24 (variable network) data sets.

We acknowledge our likely suboptimal choices of computational steps for the bridge sampling workflow or the neural networks and hence wish to stress the general patterns of nonamortized versus amortized methods demonstrated here. In general, we expect an advantage of bridge sampling in terms of efficiency in situations where only one or a few data sets are available and obtaining a large number of posterior draws is feasible. The demonstrated amortization property of our method might not be so relevant for inference on a single hierarchical data set, but it becomes crucial for performing calibration or recovery studies, which necessitate multiple refits of the same model (Schad et al., 2023).

Validation Study 2: Hierarchical SDT Versus MPT Models

We now extend our validation experiments from the simple setup with nested HMs to the comparison of nonnested HMs of cognition. In this simulation study, we examine the ability of our method to distinguish between data sets generated either from an HM based on a signal detection theory model (SDT model; Green & Swets, 1966) or a hierarchical multinomial processing tree model (MPT model; Riefer & Batchelder, 1988). For illustrative purposes, we embed our simulation study within an old–new recognition scenario, where participants indicate whether or not a stimulus was previously presented to them.

We ensure a challenging model comparison setting via three design aspects: first, we specify both models to possess a similar generative behavior, that is, hardly distinguishable prior predictive distributions of hit rates and false alarm rates (prior predictive plots are provided in Appendix C). Second, data sets of old–new recognition typically contain low information as they only consist of binary variables indicating the stimulus type and response, respectively. Third, we further amplify the information sparsity of the data sets by choosing a particularly small size for all data sets of $M = 25$ simulated participants and $N_m = 50$ observations per participant.

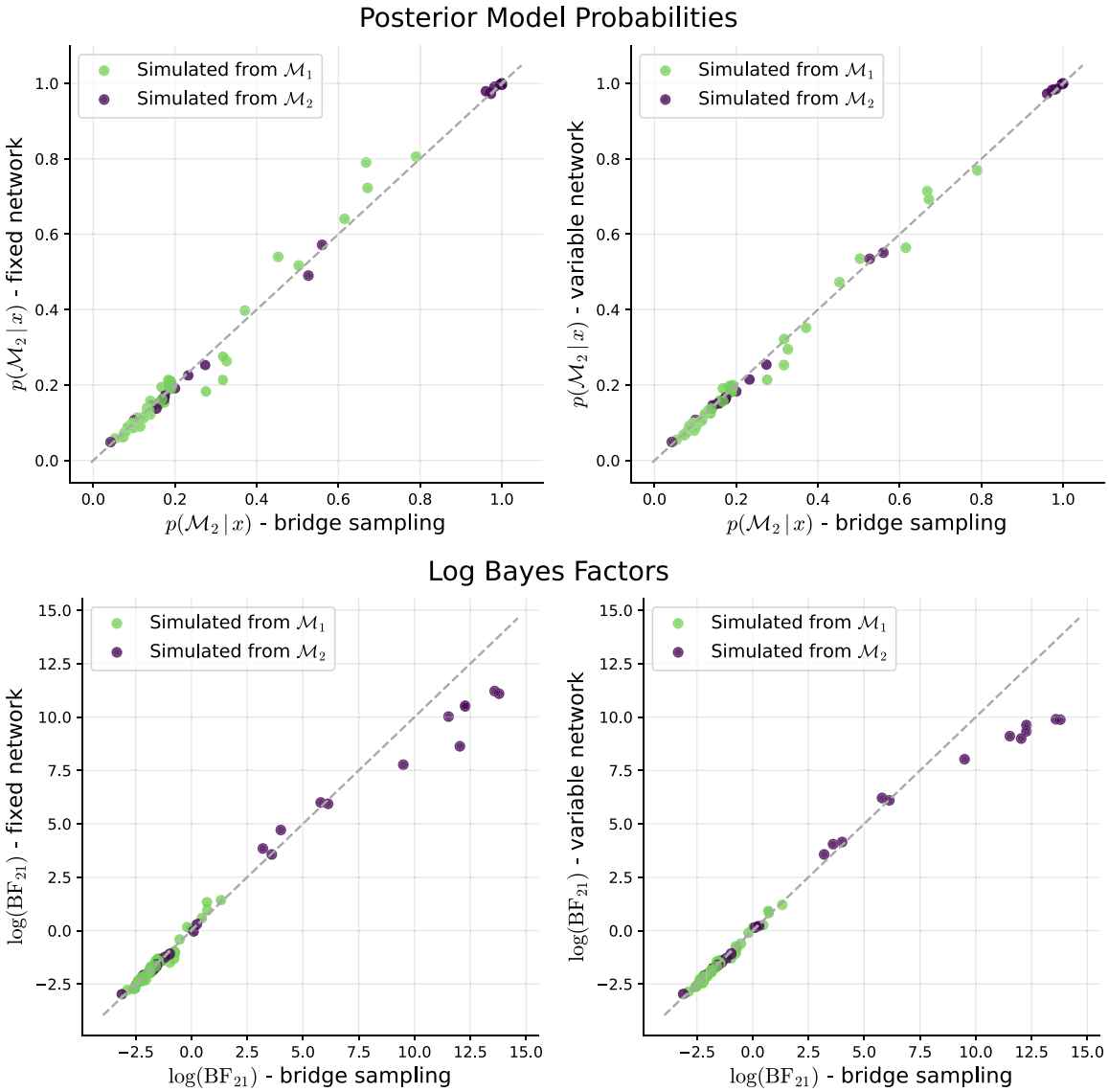
A major difference between the compared cognitive model classes lies in the assumption of a continuous latent process by the SDT model and discrete processes (or states) by the MPT model. Our specification of the SDT model follows the hierarchical formulation of the standard equal-variance model by Rouder and Lu (2005). As the competing MPT model, we specify a hierarchical latent-trait two-high-threshold model (Klauer, 2010), which, in contrast to the SDT model, explicitly models correlations between its parameters. We follow the convention of restricting the parameters that describe the probability of recognizing a previously presented stimulus as old and a distractor stimulus as new to be equal, $D_O = D_N$, to render the MPT model identifiable (Erdfelder et al., 2009; Singmann & Kellen, 2013). Our prior choices for the parameters of both models are described in Appendix C.

We train the neural network for 50,000 training steps. As in the Validation Study 1: Hierarchical Normal Models section, we first leverage the amortization property of our method to inspect its calibration for the current model comparison task. Figure 6a shows that the trained neural network generates well-calibrated PMP approximations (median ECE over 25 repetitions of $\widehat{\mathrm{ECE}} = 0.009$).

Next, we assess whether the observed calibration of the network translates into a competitive performance relative to bridge sampling. The benchmarking setup (50 simulated data sets from each model) and the implementation of the bridge sampling workflow follow the procedure described in the Bridge Sampling Comparison section.

The classification metrics depicted in Table 2 reveal the excellent performance of both methods, despite the challenging BMC scenario. We further observe a high degree of convergence between approximate PMPs derived by the two methods (cf. Figure 6b).

Figure 5
Validation Study 1: Comparison of Approximation Results



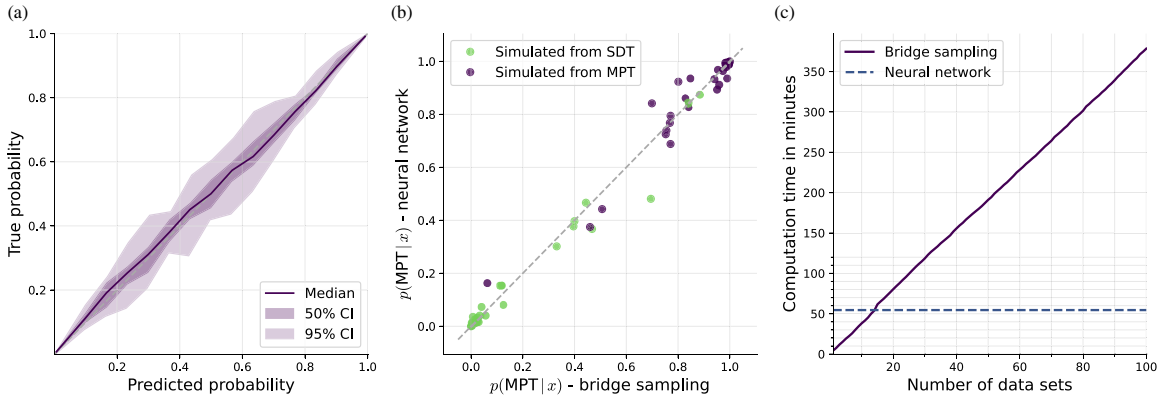
Note. Comparing bridge sampling versus the neural network trained on fixed data set sizes (left) and the neural network trained on variable data set sizes (right). For visibility purposes, the BF plots include only those 73 data sets for which bridge sampling approximated a $\mathrm{BF}_{21} < 1{,}000{,}000$ (plots with all data sets are provided in Appendix B). BF = Bayes factor. See the online article for the color version of this figure.

Again, we find discrepancies between bridge sampling and our method in areas of extreme evidence (see Figure C2 for log BFs). As depicted in Figure 6c, obtaining PMP approximations for the 100 test data sets took more than 6 hr for bridge sampling and 55 min for the neural network. For this comparison of more complex cognitive models, the amortization advantage of our method emerges when analyzing 15 or more data sets. Note that this advantage would quickly show up in validation studies involving multiple model refits (e.g., bootstrap, sensitivity analysis, or cross-validation).

All validation experiments so far have been set up in an $\mathcal{M}$-closed setting, with validation data simulated from the set $\mathcal{M}$ of models under consideration (Bernardo & Smith, 1994). Therefore, as a final validation, we test whether our method also behaves sensibly

Figure 6

Validation Study 2: Results for the Comparison Between Hierarchical SDT and MPT Models



Note. (a) Calibration of the neural network over 25 repetitions with 5,000 data sets each, (b) convergence of approximate PMPs, and (c) computation times. PMP = posterior model probability; SDT = signal detection theory; MPT = multinomial processing tree; CIs = confidence intervals. See the online article for the color version of this figure.

in an $\mathcal{M}$-open setting, where none of the models generated the test data. For this, we simulate 100 noise data sets with the same hierarchical structure as before but generate the binary values for stimulus types and responses from a Bernoulli distribution with $p = 0.5$. Our neural method agrees with bridge sampling by assigning very high PMPs to the SDT model for all noise data sets ($\bar{\pi}_{SDT}^{(\text{bridge})} = 0.999965$; $\bar{\pi}_{SDT}^{(\text{neural})} = 0.999958$). Correspondingly, the deviations between both methods are minimal. We thus observe a close alignment between bridge sampling and our neural method in both a well-specified and a misspecified scenario. This tentative result suggests that our amortized estimates are faithful approximations not only in an $\mathcal{M}$-closed but also an $\mathcal{M}$-open setting, at least for this BMC scenario.

The converging results from the two validation studies demonstrate that our neural method generates well-calibrated and accurate PMP approximations. Despite our method only accessing the likelihood function indirectly via simulations, it can successfully compete with bridge sampling, which has direct access to the likelihood function.

Application: Hierarchical Evidence Accumulation Models

In the following, we showcase the utility of our method by comparing complex hierarchical EAMs in a real-data situation where likelihood-based methods such as bridge sampling would not be applicable. More precisely, we seek to test the explanatory power of different stochastic diffusion model formulations proposed by Voss et al. (2019) for experimental response time data.

The so-called Lévy flight model increases the flexibility of the standard Wiener diffusion model (Ratcliff et al., 2016) but renders its likelihood function intractable with standard numerical approximations (Voss & Voss, 2007). The complete incorporation of all information through hierarchical modeling and the realization of BMC has consequently been infeasible so far. Thus, in a recent study, Wieschen et al. (2020) had to resort to a separate computation of the Bayesian information criterion (BIC) for each participant with subsequent aggregation. We aim to extend the study of Wieschen et al. (2020) by comparing fully hierarchical EAMs through PMPs and BFs. Moreover, we intend to answer the question formulated by Wieschen et al. (2020) as to whether the superior performance of the more complex models in their study stems from an insufficient punishment of model flexibility by the BIC. In addition to addressing a substantive research question in this application, we also demonstrate multiple advantages of our deep learning method on empirical data:

- Compare HMs with intractable likelihoods: As our method is simulation-based, including models with intractable likelihood functions in the comparison set does not alter its feasibility.

Table 2

Validation Study 2: Performance Metrics for the Comparison Between Hierarchical SDT and MPT Models

	Accuracy	MAE	RMSE	Log-Score	SBC
Bridge sampling	0.95 (0.02)	0.1 (0.02)	0.21 (0.04)	0.15 (0.04)	0.0 (0.04)
Neural network	0.95 (0.02)	0.09 (0.02)	0.21 (0.03)	0.14 (0.04)	0.0 (0.04)

Note. Bootstrapped mean values and standard errors (in parentheses) are presented. We use 1,000 bootstrap versions of the test data sets and estimate the standard errors from the bootstrap standard deviations of the metrics. MAE = mean absolute error; RMSE = root-mean-square error; SBC = simulation-based calibration.

- Adequately model nested data: Our method alleviates computational challenges that prevent modelers from adequately capturing the information contained in nested data structures through HMs.
- Reuse trained networks via fine-tuning: We accelerate the training of our neural network by pretraining it on less complex simulated data and subsequently fine-tuning it on simulated data resembling the actual experimental setting.
- Handle missing data: We train a neural network that can handle varying amounts of missing data by randomly masking simulated data during the training process.
- Validate a trained network on simulated data: The amortized nature of our method allows for extensive validation of a trained network prior to its application to empirical data.

Model Specification

For this application, we consider a Lévy flight model with nonGaussian noise (Voss et al., 2019). The Lévy flight process is driven by the following stochastic ordinary differential equation:

$$\mathrm{d}x = v\,\mathrm{d}t + \sigma\,\mathrm{d}\xi \tag{33}$$

$$\xi \sim \text{AlphaStable}\left(\alpha,\ \mu = 0,\ \sigma = \frac{1}{\sqrt{2}},\ \beta = 0\right), \tag{34}$$

which represents a Lévy walk characterized by a fat-tailed stable noise distribution.[4] In the above equation, x denotes the accumulated (perceptual) evidence, v denotes the rate of accumulation, and α controls the tail exponent of the noise variate ξ. Voss et al. (2019) and Wieschen et al. (2020) argued that the more abrupt changes in the information accumulation process that this model allows for could provide a better description of human decision-making than a Gaussian noise. The addition of Lévy noise renders the standard numerical approximation of the diffusion model likelihood intractable (Voss & Voss, 2007). Consequently, neither standard MCMC nor bridge sampling are applicable for Bayesian parameter estimation and BMC, respectively.

There is an ongoing debate about the inclusion of additional parameters that account for intertrial variability in the diffusion model parameters: while they can provide a better model fit, the estimation of intertrial variability parameters is often difficult and can result in unstable results (Boehm et al., 2018; Lerche & Voss, 2016). Thus, Wieschen et al. (2020) also compared basic (without intertrial variability parameters) to full (with intertrial variability parameters) versions of the drift-diffusion and Lévy flight models.

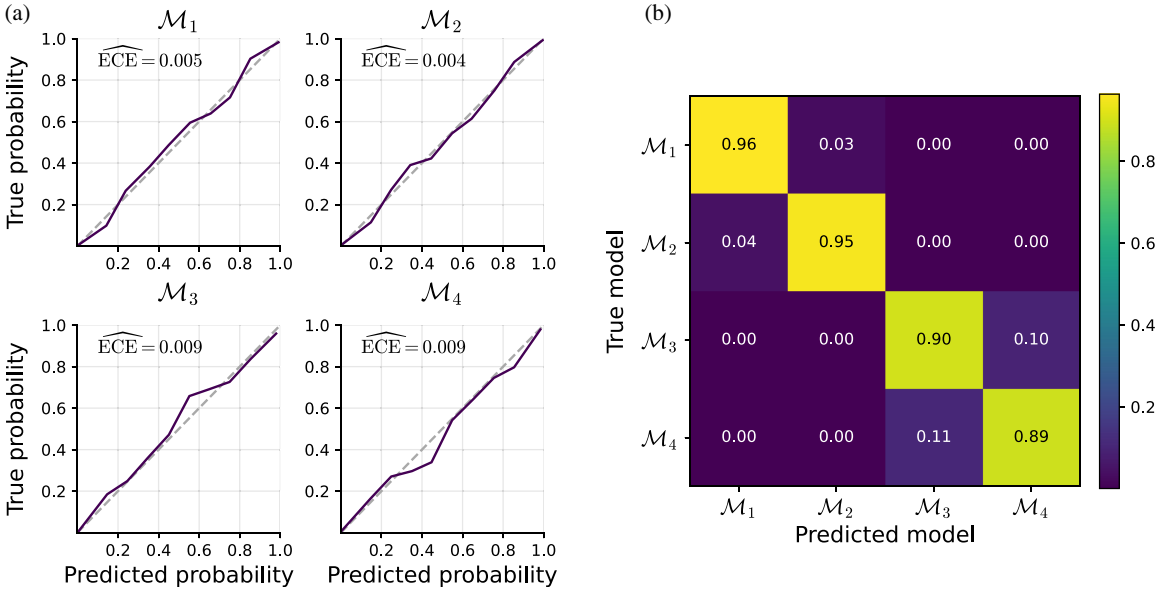
Consequently, the set of candidate models considered here consists of four EAMs with increasing flexibility (i.e., the scope of possible data patterns that they can generate):

- $\mathcal{M}_1$, the most parsimonious basic diffusion model with the parameter v describing the mean rate of information uptake, the parameter a describing the threshold at which a decision is made, the parameter z_r describing a bias of the starting point towards one decision alternative and the parameter t_0 describing the nondecision time, that is, the time spent encoding the stimulus and executing the decision.

[4] An earlier version of this work used the original formulation by Voss et al. (2019), which sets $\sigma = 1$. For the special case of $\alpha = 2.0$, which is equivalent to the Wiener diffusion model, $\sigma = 1$ leads to an unusual diffusion constant (standard deviation of Gaussian noise) of $\sqrt{2}$, whereas $\sigma = \frac{1}{\sqrt{2}}$ ensures the conventional diffusion constant of 1. Notably, model comparison results are highly sensitive to the choice of σ.

Figure 7

Real-Data Application: Validation Results for the Evidence Accumulation Models on 2,000 Simulated Data Sets per Model



Note. (a) Calibration curves and (b) confusion matrix. ECE = expected calibration error. See the online article for the color version of this figure.

Table 3
Real-Data Application: BFs and PMPs Estimated From Data by Wieschen et al. (2020)

	$\mathcal{M}_1$	$\mathcal{M}_2$	$\mathcal{M}_3$	$\mathcal{M}_4$
BF_{j3}	9.63×10^{-5}	0.01	[a]	0.27
BF_{3j}	1.04×10^{4}	78	[a]	3.71
PMP	7.51×10^{-5}	1.00×10^{-2}	0.78	0.21

Note. BFs = Bayes factors; PMPs = posterior model probabilities.
[a] The preferred model.

- $\mathcal{M}_2$, the basic Lévy flight model, in which the assumption of a Wiener diffusion process with Gaussian noise is replaced by the above introduced Lévy flight process. The additional free parameter α governs the tail behavior of the noise distribution. The setting $\alpha = 2$ is equivalent to a Gaussian distribution, whereas $\alpha = 1$ reduces to a Cauchy distribution.
- $\mathcal{M}_3$, the full diffusion model, which extends $\mathcal{M}_1$ with the parameters s_{v_m}, s_{z_m}, and s_{t_m} that denote the variability (i.e., standard deviations) of drift rate, starting point bias, and nondecision time, respectively, between trials.
- $\mathcal{M}_4$, the full Lévy flight model, that possesses the largest flexibility by including intertrial variability parameters as well as the flexible Lévy noise distribution controlled by α.

Parameter priors and prior predictive checks are provided in Appendix D.

Data

The reanalyzed data set by Wieschen et al. (2020) contains 40 participants who completed a total of 900 trials of binary decision tasks (color discrimination and lexical decision) each. On average, 3.17% of trials per participant were excluded due to extremely short or long reaction times.

Simulation-Based Training

Since simulating data from EAMs can be challenging, especially when they include nonGaussian noise, we leverage the advantage that neural networks are capable of transfer learning as described in the Method section. Transfer learning describes the utilization of representations that had been previously learned by a neural network in a particular task for a new, related task (e.g., Ng et al., 2015). In this way, neural networks can be applied in small data settings (e.g., a limited simulation budget) by reusing the training knowledge encoded from structurally similar (possibly big data) settings.

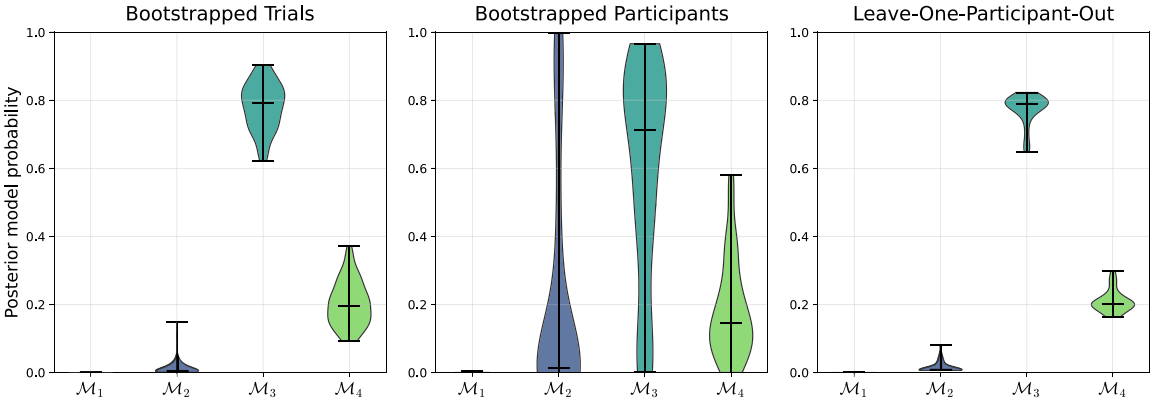
For the purpose of model comparison, we first pretrain the network for 20 epochs (i.e., passes over the whole training data) on 10,000 simulated data sets per model. These data sets resemble the empirical data in that they consist of 40 simulated participants, but differ in that the number of trials is reduced by a factor of 9 (100 instead of 900 trials per participant). Afterwards, we fine-tune the network for additional 30 epochs on 2,000 simulated data sets per model that match the empirical data set with 40 simulated participants and 900 trials per participant. Thereby, we considerably reduce the computational demand of the training process. We further speed up the training phase by simulating all data prior to the training of the network in the high-performance programming language Julia (Bezanson et al., 2017). Pretraining took 10 min for the simulations and 11 min for training the networks. Fine-tuning took 18 min for the simulations and 16 min for training the networks, resulting in a total of 55 min for the training phase.

To fully adapt the network to the characteristics of the empirical data, we also simulate missing data during fine-tuning. In each training epoch, we generate a random binary mask $\boldsymbol{f}$ coding the simulated missing values. We sample the number of masked trials from a (discretized) normal distribution truncated between one and the number of trials, 900. The distributions' mean and standard deviation match the amount and variability of missing trials in the empirical data. We then perform an element-wise multiplication $\tilde{\boldsymbol{x}} = \boldsymbol{x} \otimes \boldsymbol{f}$ and feed the "contaminated" data $\tilde{\boldsymbol{x}}$ to the network. This procedure results in a robust network that can process various proportions of missing data. We find rank stability of our results in the presence of up to 25% missing data in Appendix D.

Results

Before applying our trained network to the empirical data, we validate it on 2,000 simulated data sets per model. First, the individual

Figure 8
Real-Data Application: Model Posteriors on the Empirical Data Set With Uncertainty Under Different Data Perturbations



Note. We use 100 bootstrap samples for the bootstrapped results. See the online article for the color version of this figure.

calibration curves in Figure 7a show an excellent calibration for all models with $\widehat{\text{ECE}}$s close to 0. The calibration curves now consist of 10 instead of 15 intervals to obtain stable results despite the smaller amount of validation data sets per model. Second, we evaluate the accuracy of recovery and patterns of misclassification through the confusion matrix depicted in Figure 7b. The confusion matrix confirms that the excellent calibration of the network does not stem from chance performance. It also reveals that the selection of the "true" model becomes more difficult with increasing model complexity, which is a direct consequence of the Occam's razor property inherent in BMC (cf. Figure 1).

Table 3 presents the model comparison results on the empirical data set. Additionally, Figure 8 displays the model posteriors under different data perturbations. Consistent with the results of the nonhierarchical BIC approach by Wieschen et al. (2020), we find little evidence for both the basic diffusion model $\mathcal{M}_1$ and the basic Lévy flight model $\mathcal{M}_2$. This implies that the additional complexity of allowing parameters to vary between trials in $\mathcal{M}_3$ and $\mathcal{M}_4$ is, even under the strict penalization of prior-predictive flexibility in BMC, outweighed by better model fit. Also in agreement with Wieschen et al. (2020), we observe evidence for both $\mathcal{M}_3$ and $\mathcal{M}_4$, but, in contrast to Wieschen et al. (2020), our results slightly favor the full diffusion model $\mathcal{M}_3$ over the full Lévy flight model $\mathcal{M}_4$. Figure 8 confirms both the slight advantage of $\mathcal{M}_3$ over $\mathcal{M}_4$ and the substantial uncertainty associated with these results.

Discussion

Nested data are ubiquitous in the quantitative sciences, including psychological and cognitive research (Farrell & Lewandowsky, 2018). Yet, to avoid dealing with the complex dependencies resulting from these data, researchers often resort to simpler analyses, ignoring potentially important structural information. HMs provide a flexible way to represent the multilevel structure of nested data, but this flexibility can make BMC a daunting undertaking.

In this work, we proposed a powerful remedy to this problem: building on the BayesFlow framework (Radev et al., 2020), we developed a neural network architecture that enables approximate BMC for arbitrarily complex HMs. In two simulation studies, we showed that our deep learning method is well-calibrated and performs as accurately as bridge sampling, which is the current state-of-the-art for comparing HMs with simple likelihoods. Moreover, in a subsequent real-data application, we compared the relatively new Lévy flight model with existing evidence accumulation models. Thus, we argue that our method is well-suited to enhance the applicability of (complex) HMs in psychological research. Below, we summarize the key properties and limitations of our method while also outlining future research directions.

Amortized Inference

Our method offloads the computational demands for comparing HMs onto the training phase of a custom neural network, allowing for near real-time model comparison using the trained network. The resulting amortization offers several advantages over nonamortized methods.

First, it enables thorough validation of a trained network on thousands of simulated data sets, allowing large-scale simulation-based diagnostics to become an integral part of the BMC workflow (Gelman et al., 2020; Schad et al., 2023). Second, the trained and validated networks can be used not only for point estimates of BFs or PMPs on empirical data but also for exploring the robustness of the results against multiple data perturbations, as showcased in our real-data application.

Third, we demonstrated the feasibility of amortizing over variable data set sizes in our first validation study. This is particularly advantageous in the context of HMs since nested data sets often contain multiple exchangeable levels with variable sizes (e.g., different number of clusters, participants, and observations). Analyzing multiple hierarchical data sets with variable sizes only requires a single network that has seen different data set sizes during training. The same network could also be used for various simulation studies, such as the challenging task of designing maximally informative experiments in a hierarchical BMC setting (Heck & Erdfelder, 2019; Myung & Pitt, 2009).

Lastly, we showed that researchers do not even need to consider all possible shapes of future data sets when training such a network, as they can use transfer learning to efficiently adapt a trained network to a related setting. Beyond allowing more flexibility in reusing networks across experiments, researchers or even fields, transfer learning can also considerably reduce the computational demands associated with comparing complex HMs. As demonstrated in our real-data application, a network can be pretrained on simulated data sets with reduced size and fine-tuned afterwards on sizes matching the empirical data.

Independence From Explicit Likelihoods

Unlike other popular methods for performing BMC on HMs, such as the Savage–Dickey density ratio or bridge sampling, our method is not constrained by the availability of an explicit likelihood function for all competing models. As long as the models in question can be implemented as simulators, the neural network can be trained to perform BMC on these models. The value of such a method is evident, as it decouples the substantive task of model specification from concerns about the feasibility of estimation methods.

Statistical models are instantiations of substantive knowledge or hypotheses. As such, we argue that model specification should not be unduly restricted by considerations of computational tractability—a sentiment that is closely related to what Haaf et al. (2021) call the "specification-first-principle." Our proposed deep learning method satisfies this principle, as model specification may be guided exclusively by substantive arguments with few concerns about tractability. Thus, we believe that our method makes a contribution to the recent upsurge of innovative psychological models (Collins & Shenhav, 2022; Ghaderi-Kangavari et al., 2023; Heathcote & Matzke, 2022) by allowing for an efficient assessment of their incremental value in a hierarchical setting.

Limitations and Outlook

One of the main challenges of approximate methods and, more broadly, statistical inference, is ensuring the faithfulness of the obtained results. The outlined possibilities for validating amortized model comparison and examining the robustness of the results are important contributions of our method, but they come with open

questions. Concerning the validation of the network, framing model comparison as a supervised learning problem allows us to draw on the rich literature on classification performance and calibration metrics. Nevertheless, determining a "good enough" score for an approximate BMC method remains challenging, as the optimally possible performance is application-specific and usually unknown.

Concerning the application of the network to empirical data, we showed in Validation Study 2 that our method produces, at least in this scenario, reasonable results when confronted with data not stemming from the models under consideration. Moreover, our robustness checks are a practical proxy for measuring the reliability of BMC results in a closed-world setting. However, these checks cannot possibly capture the (lack of) absolute evidence for an HM: as a relative method, BMC may indicate that one model fits the data better than a set of competing models, but it does not provide any measure of how well (or poorly) the model itself approximates the underlying data-generating process. Additionally, it has been shown that severe model misspecification, with fundamental gaps between simulated and empirical data (e.g., parameter priors that exclude essential regions), can lead to unreliable simulation-based inference (Cranmer et al., 2020; Frazier & Drovandi, 2021). A promising direction to address this limitation could be the combination of our method with the recently proposed meta-uncertainty framework for BMC (Schmitt et al., 2022), which can be greatly accelerated with amortized deep learning methods. This combination could provide a principled delineation of different uncertainty sources, enabling the detection of model misspecification cases where none of the competing HMs can explain the observed data. Still, further research is needed to determine whether meta-uncertainty can provide reliable evidence for the open versus closed world assumption in the context of HMs.

Since BMC is a marginal likelihood (i.e., prior predictive) approach, priors over model parameters should be informed by scientific theory and will thus have a decisive influence on the results (Vanpaemel, 2010). We do not intend to re-iterate the ongoing discussion about this property of BMC (Gronau & Wagenmakers, 2019a,2019b; Haaf et al., 2021; Vehtari et al., 2019), but want to highlight a specific difficulty that arises for HMs: Parameter priors of an HM are connected via multilevel dependencies, increasing the risk that poor prior choices lead to nonintended model behavior (for a recent discussion of this problem in cognitive modeling, see Sarafoglou et al., 2022). Therefore, prior predictive checks and prior sensitivity analyses become especially important when conducting BMC on competing HMs. While transfer learning reduces the computational demands of retraining a neural network for sensitivity analyses, another avenue for future research would be the amortization over different prior choices, enabling immediate prior sensitivity assessment.

Finally, it should be noted that the version of our method explored here can only compare HMs assuming exchangeable data at each hierarchical level. Although the majority of HMs in social science research follow this probabilistic symmetry, some researchers may want to compare nonexchangeable HMs, for example, to study within-person dynamics (Driver & Voelkle, 2018; Lodewyckx et al., 2011; Schumacher et al., 2022). Fortunately, the modularity of our method allows easy adaptation of the neural network architecture to handle nonexchangeable HMs. To compare hierarchical time-series models with temporal dependencies at the lowest level, for instance, the first invariant module could be exchanged for a recurrent network, as proposed by Radev, D'Alessandro et al. (2021) for nonhierarchical models. Thus, future research could extend and validate our method in BMC settings involving nonexchangeable HMs.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., & Devin, M. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. arXiv preprint arXiv:1603.04467.

Barron, A., Schervish, M. J., & Wasserman, L. (1999). The consistency of posterior distributions in nonparametric problems. *The Annals of Statistics*, *27*(2), 536–561. https://doi.org/10.1214/aos/1018031206

Beaumont, M. A. (2010). Approximate Bayesian computation in evolution and ecology. *Annual Review of Ecology, Evolution, and Systematics*, *41*(1), 379–406. https://doi.org/10.1146/ecolsys.2010.41.issue-1

Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). *Curriculum learning* [Conference session]. In Proceedings of the 26th Annual International Conference on Machine Learning.

Bennett, C. H. (1976). Efficient estimation of free energy differences from Monte Carlo data. *Journal of Computational Physics*, *22*(2), 245–268. https://doi.org/10.1016/0021-9991(76)90078-4

Bernardo, J. M., & Smith, A. F. (1994). *Bayesian theory*. John Wiley & Sons.

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, *59*(1), 65–98. https://doi.org/10.1137/141000671

Bloem-Reddy, B., & Teh, Y. W. (2020). Probabilistic symmetries and invariant neural networks. *Journal of Machine Learning Research*, *21*(1), 1–61.

Boehm, U., Annis, J., Frank, M. J., Hawkins, G. E., Heathcote, A., Kellen, D., Krypotos, A.-M., Lerche, V., Logan, G. D., Palmeri, T. J., van Ravenzwaaij, D., Servant, M., Singmann, H., Starns, J. J., Voss, A., Wiecki, T. V., Matzke, D., & Wagenmakers, E.-J. (2018). Estimating across-trial variability parameters of the diffusion decision model: Expert advice and recommendations. *Journal of Mathematical Psychology*, *87*, 46–75. https://doi.org/10.1016/j.jmp.2018.09.004

Bürkner, P. C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, *80*(1), 1–28. https://doi.org/10.18637/jss.v080.i01

Bürkner, P. C., Scholz, M., & Radev, S. T. (2022). *Some models are useful, but how do we know which ones? Towards a unified Bayesian model taxonomy*. arXiv preprint arXiv:2209.02439.

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., & Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, *76*(1), 1–32. https://doi.org/10.18637/jss.v076.i01

Clarté, G., Robert, C. P., Ryder, R. J., & Stoehr, J. (2021). Componentwise approximate Bayesian computation via Gibbs-like steps. *Biometrika*, *108*(3), 591–607. https://doi.org/10.1093/biomet/asaa090

Clarté, L., Loureiro, B., Krzakala, F., & Zdeborová, L. (2022). *A study of uncertainty quantification in overparametrized high-dimensional models*. arXiv preprint arXiv:2210.12760.

Collins, A. G., & Shenhav, A. (2022). Advances in modeling learning and decision-making in neuroscience. *Neuropsychopharmacology*, *47*(1), 104–118. https://doi.org/10.1038/s41386-021-01126-y

Congdon, P. (2006). Bayesian model choice based on Monte Carlo estimates of posterior model probabilities. *Computational Statistics & Data Analysis*, *50*(2), 346–357. https://doi.org/10.1016/j.csda.2004.08.001

Cranmer, K., Brehmer, J., & Louppe, G. (2020). The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, *117*(48), 30055–30062. https://doi.org/10.1073/pnas.1912789117

Cremer, C., Li, X., & Duvenaud, D. (2018). *Inference suboptimality in variational autoencoders* [Conference session]. In Proceedings of the 35th International Conference on Machine Learning.

Csilléry, K., Blum, M. G., Gaggiotti, O. E., & François, O. (2010). Approximate Bayesian computation (ABC) in practice. *Trends in Ecology & Evolution*, *25*(7), 410–418. https://doi.org/10.1016/j.tree.2010.04.001

DeGroot, M. H., & Fienberg, S. E. (1983). The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, *32*(1–2), 12–22. https://doi.org/10.2307/2987588

Dickey, J. M., & Lientz, B. (1970). The weighted likelihood ratio, sharp hypotheses about chances, the order of a Markov chain. *The Annals of Mathematical Statistics*, *41*(1), 214–226. https://doi.org/10.1214/aoms/1177697203

Driver, C. C., & Voelkle, M. C. (2018). Hierarchical Bayesian continuous time dynamic modeling. *Psychological Methods*, *23*(4), 774–799. https://doi.org/10.1037/met0000168

Erdfelder, E., Auer, T. S., Hilbig, B. E., Aßfalg, A., Moshagen, M., & Nadarevic, L. (2009). Multinomial processing tree models: A review of the literature. *Zeitschrift für Psychologie/Journal of Psychology*, *217*(3), 108–124. https://doi.org/10.1027/0044-3409.217.3.108

Farrell, S., & Lewandowsky, S. (2018). *Computational modeling of cognition and behavior*. Cambridge University Press.

Fengler, A., Govindarajan, L. N., Chen, T., & Frank, M. J. (2021). Likelihood approximation networks (LANs) for fast inference of simulation models in cognitive neuroscience. *eLife*, *10*, Article e65074. https://doi.org/10.7554/eLife.65074

Frazier, D. T., & Drovandi, C. (2021). Robust approximate Bayesian inference with synthetic likelihood. *Journal of Computational and Graphical Statistics*, *30*(4), 958–976. https://doi.org/10.1080/10618600.2021.1875839

Gelman, A. (2006). Multilevel (hierarchical) modeling: What it can and cannot do. *Technometrics*, *48*(3), 432–435. https://doi.org/10.1198/004017005000000661

Gelman, A., & Meng, X. L. (1998). Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical Science*, 163–185. https://www.jstor.org/stable/2676756

Gelman, A., Vehtari, A., Simpson, D., Margossian, C. C., Carpenter, B., Yao, Y., Kennedy, L., Gabry, J., Bürkner, P.-C., & Modrák, M. (2020). *Bayesian workflow*. arXiv preprint arXiv:2011.01808.

Ghaderi-Kangavari, A., Rad, J. A., & Nunez, M. D. (2023). A general integrative neurocognitive modeling framework to jointly describe EEG and decision-making on single trials. *Computational Brain & Behavior*, 6, 317–376. https://doi.org/10.1007/s42113-023-00167-4

Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, *102*(477), 359–378. https://doi.org/10.1198/016214506000001437

Gonçalves, P. J., Lueckmann, J. M., Deistler, M., Nonnenmacher, M., Öcal, K., Bassetto, G., Chintaluri, C., Podlaski, W. F., Haddad, S. A., Vogels, T. P., & Greenberg, D. S. (2020). Training deep neural density estimators to identify mechanistic models of neural dynamics. *eLife*, *9*, Article e56261. https://doi.org/10.7554/eLife.56261

Grathwohl, W., Wang, K. C., Jacobsen, J. H., Duvenaud, D., Norouzi, M., & Swersky, K. (2019). *Your classifier is secretly an energy based model and you should treat it like one*. arXiv preprint arXiv:1912.03263.

Green, D. M., & Swets, J. A. (1966). *Signal detection theory and psychophysics* (Vol. 1). Wiley.

Gronau, Q. F. (2021). *Hierarchical normal example (Stan)*. https://cran.csiro.au/web/packages/bridgesampling/vignettes/bridgesampling_example_stan.html

Gronau, Q. F., Heathcote, A., & Matzke, D. (2020). Computing Bayes factors for evidence-accumulation models using Warp-III bridge sampling. *Behavior Research Methods*, *52*(2), 918–937. https://doi.org/10.3758/s13428-019-01290-6

Gronau, Q. F., Sarafoglou, A., Matzke, D., Ly, A., Boehm, U., Marsman, M., Leslie, D. S., Forster, J. J., Wagenmakers, E.-J., & Steingroever, H. (2017). A tutorial on bridge sampling. *Journal of Mathematical Psychology*, *81*, 80–97. https://doi.org/10.1016/j.jmp.2017.09.005

Gronau, Q. F., Singmann, H., & Wagenmakers, E. J. (2017). *Bridgesampling: An R package for estimating normalizing constants*. arXiv preprint arXiv:1710.08162.

Gronau, Q. F., & Wagenmakers, E. J. (2019a). Limitations of Bayesian leave-one-out cross-validation for model selection. *Computational Brain & Behavior*, *2*(1), 1–11. https://doi.org/10.1007/s42113-018-0011-7

Gronau, Q. F., & Wagenmakers, E. J. (2019b). Rejoinder: More limitations of Bayesian leave-one-out cross-validation. *Computational Brain & Behavior*, *2*(1), 35–47. https://doi.org/10.1007/s42113-018-0022-4

Gronau, Q. F., Wagenmakers, E. J., Heck, D. W., & Matzke, D. (2019). A simple method for comparing complex models: Bayesian model comparison for hierarchical multinomial processing tree models using Warp-III bridge sampling. *Psychometrika*, *84*(1), 261–284. https://doi.org/10.1007/s11336-018-9648-3

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). *On calibration of modern neural networks* [Conference session]. In Proceedings of the 34th International Conference on Machine Learning.

Haaf, J. M., Klaassen, F., & Rouder, J. N. (2021). *Bayes factor vs. posterior-predictive model assessment: Insights from ordinal constraints*. PsyArXiv preprint.

Haaf, J. M., & Rouder, J. N. (2017). Developing constraint in Bayesian mixed models. *Psychological Methods*, *22*(4), 779–798. https://doi.org/10.1037/met0000156

Heathcote, A., & Matzke, D. (2022). Winner takes all! What are race models, and why and how should psychologists use them? *Current Directions in Psychological Science*, *31*(5), 383–394. https://doi.org/10.1177/09637214221095852

Heck, D. W., Boehm, U., Böing-Messing, F., Bürkner, P. -C., Derks, K., Dienes, Z., Fu, Q., Gu, X., Karimova, D., Kiers, H. A., Klugkist, I., Kuiper, R. M., Lee, M. D., Leenders, R., Leplaa, H. J., Linde, M., Ly, A., Meijerink-Bosman, M., Moerbeek, M., … Palfi, B. (2022). A review of applications of the Bayes factor in psychological research. *Psychological Methods*, *28*(3), 558–579. https://doi.org/10.1037/met0000454

Heck, D. W., & Erdfelder, E. (2019). Maximizing the expected information gain of cognitive modeling via design optimization. *Computational Brain & Behavior*, *2*(3), 202–209. https://doi.org/10.1007/s42113-019-00035-0

Hermans, J., Banik, N., Weniger, C., Bertone, G., & Louppe, G. (2021). Towards constraining warm dark matter with stellar streams through neural simulation-based inference. *Monthly Notices of the Royal Astronomical Society*, *507*(2), 1999–2011. https://doi.org/10.1093/mnras/stab2181

Hinton, S. R., Davis, T., Kim, A. G., Brout, D., D'Andrea, C. B., Kessler, R., Lasker, J., Lidman, C., Macaulay, E., Möller, A., & Sako, M. (2019). Steve: A hierarchical Bayesian model for supernova cosmology. *The Astrophysical Journal*, *876*(1), Article 15. https://doi.org/10.3847/1538-4357/ab13a3

Hox, J. J., Moerbeek, M., & Van de Schoot, R. (2017). *Multilevel analysis: Techniques and applications*. Routledge.

Jalilian, A., & Mateu, J. (2021). A hierarchical spatio-temporal model to analyze relative risk variations of COVID-19: A focus on Spain, Italy and Germany. *Stochastic Environmental Research and Risk Assessment*, *35*(4), 797–812. https://doi.org/10.1007/s00477-021-02003-2

Kass, R. E., & Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association*, *90*(430), 773–795. https://doi.org/10.1080/01621459.1995.10476572

Kingma, D. P., & Ba, J. L. (2015). *Adam: A method for stochastic optimization* [Conference session]. In 3rd International Conference on Learning Representations.

Klauer, K. C. (2010). Hierarchical multinomial processing tree models: A latent-trait approach. *Psychometrika*, *75*(1), 70–98. https://doi.org/10.1007/s11336-009-9141-0

Lee, M. D. (2011). How cognitive modeling can benefit from hierarchical Bayesian models. *Journal of Mathematical Psychology*, *55*(1), 1–7. https://doi.org/10.1016/j.jmp.2010.08.013

Lerche, V., & Voss, A. (2016). Model complexity in diffusion modeling: Benefits of making the model more parsimonious. *Frontiers in Psychology*, *7*, Article 1324. https://doi.org/10.3389/fpsyg.2016.01324

Lodewyckx, T., Tuerlinckx, F., Kuppens, P., Allen, N. B., & Sheeber, L. (2011). A hierarchical state space approach to affective dynamics. *Journal of Mathematical Psychology*, *55*(1), 68–83. https://doi.org/10.1016/j.jmp.2010.08.004

Lotfi, S., Izmailov, P., Benton, G., Goldblum, M., & Wilson, A. G. (2022). *Bayesian model selection, the marginal likelihood, and generalization*. arXiv preprint arXiv:2202.11678.

MacKay, D. (2003). *Information theory, inference and learning algorithms*. Cambridge University Press.

Marin, J. M., Pudlo, P., Estoup, A., & Robert, C. (2018). *Likelihood-free model choice*. Chapman & Hall/CRC Press.

Marjoram, P., Molitor, J., Plagnol, V., & Tavaré, S. (2003). Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, *100*(26), 15324–15328. https://doi.org/10.1073/pnas.0306899100

McElreath, R. (2020). *Statistical rethinking: A Bayesian course with examples in R and Stan*. Chapman & Hall/CRC Press.

Meng, X. L., & Schilling, S. (2002). Warp bridge sampling. *Journal of Computational and Graphical Statistics*, *11*(3), 552–586. https://doi.org/10.1198/106186002457

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

Mertens, U. K., Voss, A., & Radev, S. (2018). ABrox—A user-friendly Python module for approximate Bayesian computation with a focus on model comparison. *PLoS ONE*, *13*(3), Article e0193981. https://doi.org/10.1371/journal.pone.0193981

Mestdagh, M., Verdonck, S., Meers, K., Loossens, T., & Tuerlinckx, F. (2019). Prepaid parameter estimation without likelihoods. *PLoS Computational Biology*, *15*(9), Article e1007181. https://doi.org/10.1371/journal.pcbi.1007181

Myung, J. I., & Pitt, M. A. (2009). Optimal experimental design for model discrimination. *Psychological Review*, *116*(3), 499–518. https://doi.org/10.1037/a0016104

Naeini, M. P., Cooper, G., & Hauskrecht, M. (2015). *Obtaining well calibrated probabilities using Bayesian binning*. In Proceedings of the twenty-ninth AAAI conference on artificial intelligence (pp. 2901–2907). AAAI Press.

Ng, H. W., Nguyen, V. D., Vonikakis, V., & Winkler, S. (2015). *Deep learning for emotion recognition on small datasets using transfer learning* [Conference session]. In Proceedings of the 2015 ACM on International Conference on Multimodal Interaction.

Nicenboim, B., Schad, D. J., & Vasishth, S. (2022). *An introduction to Bayesian data analysis for cognitive science*. https://vasishth.github.io/bayescogsci/book/.

Niculescu-Mizil, A., & Caruana, R. (2005). *Predicting good probabilities with supervised learning*. In Proceedings of the 22nd international conference on machine learning (pp. 625–632).

O'Hagan, A. (1995). Fractional Bayes factors for model comparison. *Journal of the Royal Statistical Society: Series B (Methodological)*, *57*(1), 99–118.

Pudlo, P., Marin, J. M., Estoup, A., Cornuet, J. M., Gautier, M., & Robert, C. P. (2016). Reliable ABC model choice via random forests. *Bioinformatics*, *32*(6), 859–866. https://doi.org/10.1093/bioinformatics/btv684

Radev, S. T., D'Alessandro, M., Mertens, U. K., Voss, A., Köthe, U., & Bürkner, P. C. (2021). Amortized Bayesian model comparison with evidential deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, *34*(8), 4903–4917. https://doi.org/10.1109/TNNLS.2021.3124052

Radev, S. T., Graw, F., Chen, S., Mutters, N. T., Eichel, V. M., Bärnighausen, T., & Köthe, U. (2021). OutbreakFlow: Model-based Bayesian inference of disease outbreak dynamics with invertible neural networks and its application to the COVID-19 pandemics in Germany. *PLoS Computational Biology*, *17*(10), Article e1009472. https://doi.org/10.1371/journal.pcbi.1009472

Radev, S. T., Mertens, U. K., Voss, A., Ardizzone, L., & Köthe, U. (2020). BayesFlow: Learning complex stochastic models with invertible neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, *33*(4), 1452–1466. https://doi.org/10.1109/TNNLS.2020.3042395

Radev, S. T., Schmitt, M., Schumacher, L., Elsemüller, L., Pratz, V., Schälte, Y., Köthe, U., & Bürkner, P. C. (2023). *BayesFlow: Amortized Bayesian workflows with neural networks*. arXiv preprint arXiv:2306.16015.

Ratcliff, R., Smith, P. L., Brown, S. D., & McKoon, G. (2016). Diffusion decision model: Current issues and history. *Trends in Cognitive Sciences*, *20*(4), 260–281. https://doi.org/10.1016/j.tics.2016.01.007

Riefer, D. M., & Batchelder, W. H. (1988). Multinomial modeling and the measurement of cognitive processes. *Psychological Review*, *95*(3), 318–339. https://doi.org/10.1037/0033-295X.95.3.318

Rouder, J. N., & Lu, J. (2005). An introduction to Bayesian hierarchical models with an application in the theory of signal detection. *Psychonomic Bulletin & Review*, *12*(4), 573–604. https://doi.org/10.3758/BF03196750

Rouder, J. N., & Morey, R. D. (2012). Default Bayes factors for model selection in regression. *Multivariate Behavioral Research*, *47*(6), 877–903. https://doi.org/10.1080/00273171.2012.734737

Rouder, J. N., Morey, R. D., & Pratte, M. S. (2017). Bayesian hierarchical models of cognition. In W. H. Batchelder, H. Colonius, E. N. Dzhafarov, & J. Myung (Eds.), *New handbook of mathematical psychology: Foundations and methodology* (pp. 504–551). Cambridge University Press.

Sarafoglou, A., Kuhlmann, B. G., Aust, F., & Haaf, J. M. (2022). *Theory-informed refinement of Bayesian hierarchical MPT modeling*. PsyArXiv preprint.

Schad, D. J., Betancourt, M., & Vasishth, S. (2021). Toward a principled Bayesian workflow in cognitive science. *Psychological Methods*, *26*(1), 103–126. https://doi.org/10.1037/met0000275

Schad, D. J., Nicenboim, B., Bürkner, P. C., Betancourt, M., & Vasishth, S. (2023). Workflow techniques for the robust use of Bayes factors. *Psychological Methods*, *28*(6), 1404–1426. https://doi.org/10.1037/met0000472

Schmitt, M., Radev, S. T., & Bürkner, P. C. (2022). *Meta-uncertainty in Bayesian model comparison*. arXiv preprint arXiv:2210.07278.

Schumacher, L., Bürkner, P. C., Voss, A., Köthe, U., & Radev, S. T. (2022). *Neural superstatistics: A Bayesian method for estimating dynamic models of cognition*. arXiv preprint arXiv:2211.13165.

Singmann, H., & Kellen, D. (2013). MPTinR: Analysis of multinomial processing tree models in R. *Behavior Research Methods*, *45*(2), 560–575. https://doi.org/10.3758/s13428-012-0259-0

Singmann, H., & Kellen, D. (2019). An introduction to mixed models for experimental psychology. In D. H. Spieler & E. Schumacher (Eds.), *New methods in cognitive psychology* (pp. 4–31). Routledge.

Sisson, S. A., Fan, Y., & Tanaka, M. M. (2007). Sequential Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, *104*(6), 1760–1765. https://doi.org/10.1073/pnas.0607208104

Stan Development Team. (2019). *Stan modeling language users guide and reference manual* (Version 2.21.0). https://mc-stan.org

Sunnåker, M., Busetto, A. G., Numminen, E., Corander, J., Foll, M., & Dessimoz, C. (2013). Approximate Bayesian computation. *PLoS Computational Biology*, *9*(1), Article e1002803. https://doi.org/10.1371/journal.pcbi.1002803

Talts, S., Betancourt, M., Simpson, D., Vehtari, A., & Gelman, A. (2018). *Validating Bayesian inference algorithms with simulation-based calibration*. arXiv preprint arXiv:1804.06788.

Tieleman, T., & Hinton, G. (2012). *Lecture 6.5-rmsprop, coursera: Neural networks for machine learning* (Technical Report 6). University of Toronto.

Torrey, L., & Shavlik, J. (2010). Transfer learning. In *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques* (pp. 242–264). IGI global.

Tran, N. H., van Maanen, L., Heathcote, A., & Matzke, D. (2021). Systematic parameter reviews in cognitive modeling: Towards a robust and

cumulative characterization of psychological processes in the diffusion decision model. *Frontiers in Psychology*, *11*, Article 608287. https://doi.org/10.3389/fpsyg.2020.608287

Turner, B. M., & Sederberg, P. B. (2014). A generalized, likelihood-free method for posterior estimation. *Psychonomic Bulletin & Review*, *21*(2), 227–250. https://doi.org/10.3758/s13423-013-0530-0

Turner, B. M., & Van Zandt, T. (2014). Hierarchical approximate Bayesian computation. *Psychometrika*, *79*(2), 185–209. https://doi.org/10.1007/s11336-013-9381-x

Ulitzsch, E., von Davier, M., & Pohl, S. (2020). A multiprocess item response model for not-reached items due to time limits and quitting. *Educational and Psychological Measurement*, *80*(3), 522–547. https://doi.org/10.1177/0013164419878241

Vandekerckhove, J., Tuerlinckx, F., & Lee, M. D. (2011). Hierarchical diffusion models for two-choice response times. *Psychological Methods*, *16*(1), 44–62. https://doi.org/10.1037/a0021765

Vanpaemel, W. (2010). Prior sensitivity in theory testing: An apologia for the Bayes factor. *Journal of Mathematical Psychology*, *54*(6), 491–498. https://doi.org/10.1016/j.jmp.2010.07.003

Van Rooij, I., Blokpoel, M., Kwisthout, J., & Wareham, T. (2019). *Cognition and intractability: A guide to classical and parameterized complexity analysis*. Cambridge University Press.

Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., & Bürkner, P. C. (2021). Rank-normalization, folding, and localization: an improved R for assessing convergence of MCMC (with discussion). *Bayesian Analysis*, *16*(2), 667–718. https://doi.org/10.1214/20-BA1221

Vehtari, A., Simpson, D. P., Yao, Y., & Gelman, A. (2019). Limitations of "Limitations of Bayesian leave-one-out cross-validation for model selection". *Computational Brain & Behavior*, *2*(1), 22–27. https://doi.org/10.1007/s42113-018-0020-6

Voss, A., Lerche, V., Mertens, U., & Voss, J. (2019). Sequential sampling models with variable boundaries and non-normal noise: A comparison of six models. *Psychonomic Bulletin & Review*, *26*(3), 813–832. https://doi.org/10.3758/s13423-018-1560-4

Voss, A., & Voss, J. (2007). Fast-dm: A free program for efficient diffusion model analysis. *Behavior Research Methods*, *39*(4), 767–775. https://doi.org/10.3758/BF03192967

Wagenmakers, E. J., Lodewyckx, T., Kuriyal, H., & Grasman, R. (2010). Bayesian hypothesis testing for psychologists: A tutorial on the Savage–Dickey method. *Cognitive Psychology*, *60*(3), 158–189. https://doi.org/10.1016/j.cogpsych.2009.12.001

Wiecki, T. V., Sofer, I., & Frank, M. J. (2013). HDDM: Hierarchical Bayesian estimation of the drift-diffusion model in Python. *Frontiers in Neuroinformatics*, *7*, Article 14. https://doi.org/10.3389/fninf.2013.00014

Wieschen, E. M., Voss, A., & Radev, S. (2020). Jumping to conclusion? A Lévy flight model of decision making. *The Quantitative Methods for Psychology*, *16*(2), 120–132. https://doi.org/10.20982/tqmp.16.2.p120

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., & Smola, A. J. (2017). Deep sets. *Advances in Neural Information Processing Systems*, *30*. https://papers.nips.cc/paper_files/paper/2017/hash/f22e4747da1aa27e363d86d40ff442fe-Abstract.html

Appendix A

Neural Network Implementation and Training

The neural networks are implemented in the Python library TensorFlow (Abadi et al., 2015) and jointly optimized via backpropagation. During training, we use mini-batch gradient descent with batches of size $B = 32$ per backpropagation update (training step). We employ the Adam optimizer (Kingma & Ba, 2015) with a cosine decay schedule in Validation Study 1 (initial learning rate of 5×10^{-4}) and the real-data application (initial learning rate of 5×10^{-4} for pretraining and 5×10^{-5} for fine-tuning. In Validation Study 2, we use the RMSprop optimizer (Tieleman & Hinton, 2012) with an initial learning rate of 2.5×10^{-4} and a cosine decay schedule, which we found to work better for the unusually sparse binary data. In all validation studies, we use online training, that is, simulate new training data sets flexibly right before each training step. In the real-data application, we simulate all data sets efficiently a priori in the Julia programing language and therefore use offline training, that is, training with a predetermined amount of data sets.

We use the following neural network architectures: the hierarchical summary network is composed of two deep invariant modules, each consisting of $K = 2$ equivariant modules followed by an invariant module. The inference network is realized via a standard feedforward network with three fully connected layers followed by a softmax output layer. We did not conduct a thorough search for optimal hyperparameter settings of the neural networks and the training process.

Appendix B

Validation Study 1 Details

Calibration

Additional results for the scenario containing data sets with varying number of observations are depicted in Figure B1. Accuracy and SBC (median of $\widehat{\text{SBC}} = -0.0006$) are stable across nearly all settings, only slightly dropping for data sets with few observations.

Concerning the scenario containing data sets with varying number of groups and nested observations, Figure B2 presents generally unbiased SBC results with a median of $\widehat{\text{SBC}} = 0.0004$. Figure B3 shows the marginal plots corresponding to the three-dimensional plots for all metrics.

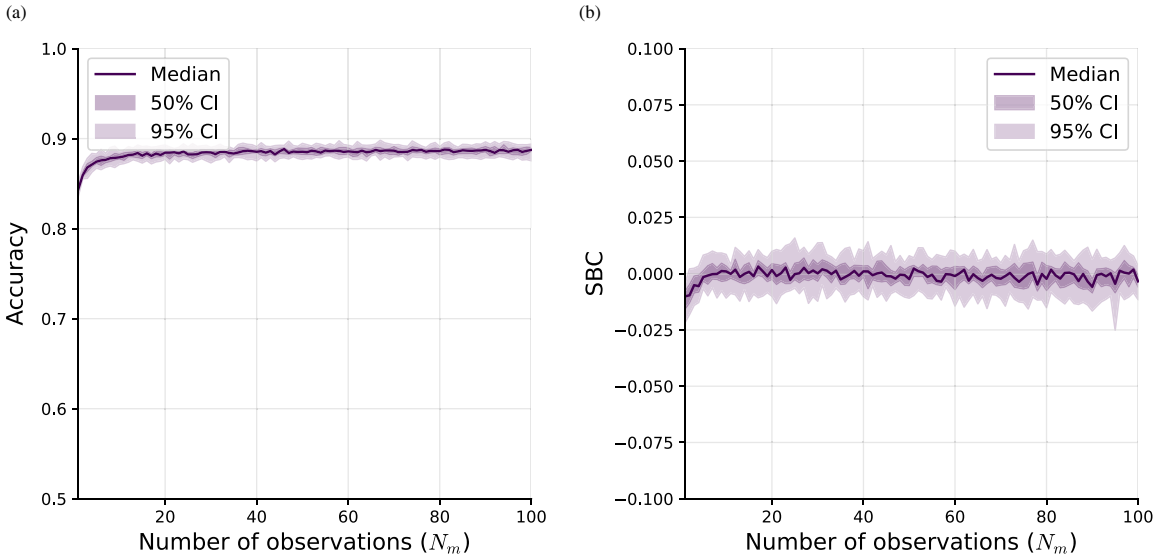
Bridge Sampling Comparison

Figure B4 displays the log BFs approximated by bridge sampling and the neural network variants for all 100 test data sets, including those 27 data sets for which bridge sampling approximated a BF $>$ 1,000,000 and that were therefore excluded in Figure 5 for visibility purposes.

(*Appendices continue*)

Figure B1

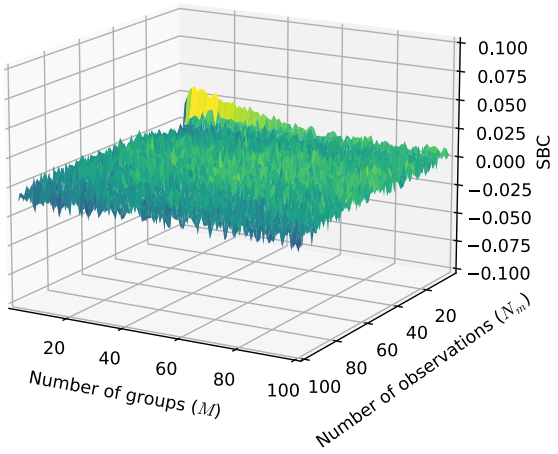
Validation Study 1: Additional Results for the Neural Network Trained and Tested on Data Sets With Varying Number of Observations



Note. (a) Accuracy of recovery and (b) SBC. SBC = simulation-based calibration; CI = confidence interval. See the online article for the color version of this figure.

Figure B2

Validation Study 1: SBC Results for the Neural Network Trained and Tested Over Variable Data Set Sizes

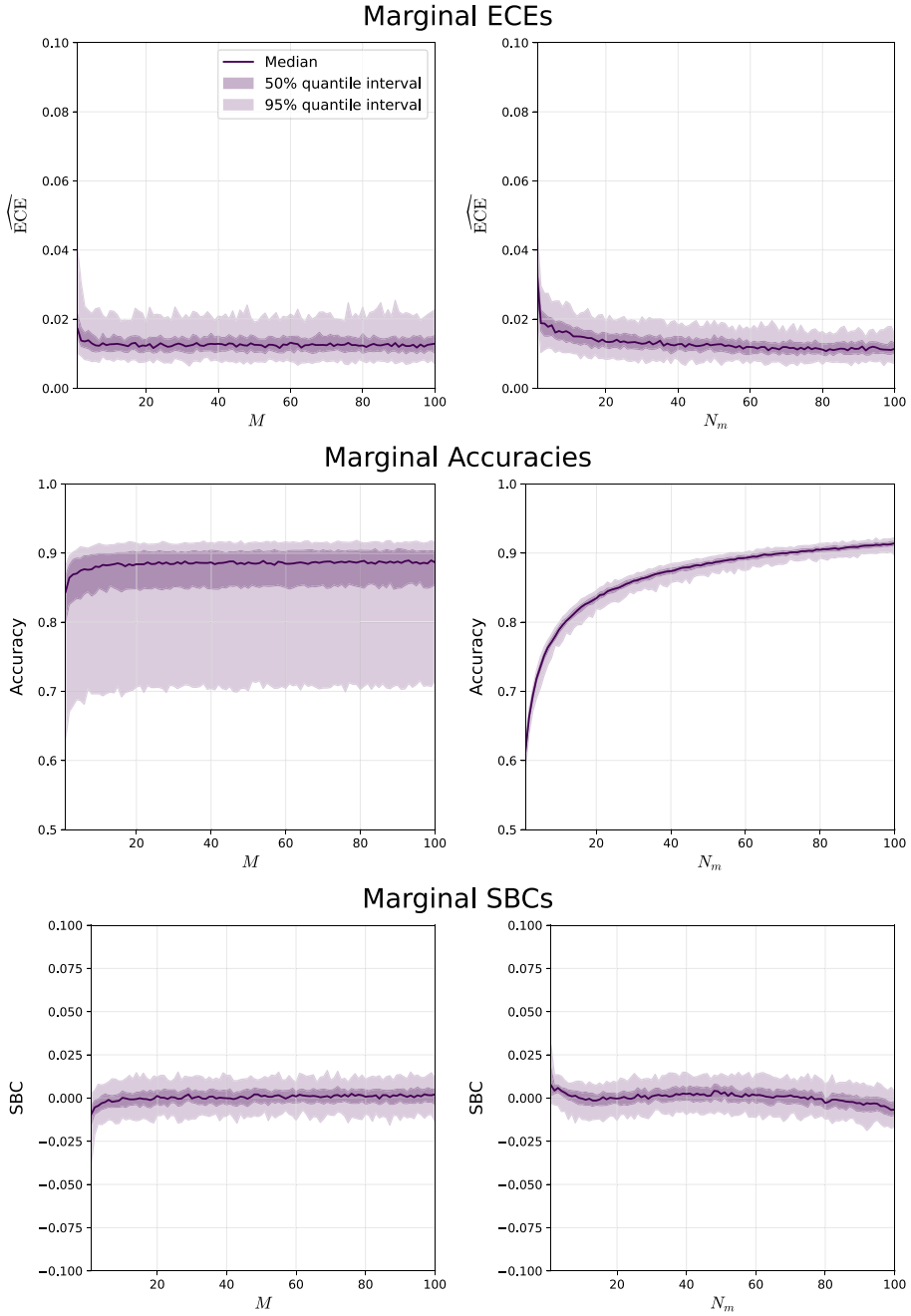


Note. SBC = simulation-based calibration. See the online article for the color version of this figure.

(*Appendices continue*)

Figure B3

Validation Study 1: Marginal Plots for the Neural Network Trained and Tested Over Variable Data Set Sizes

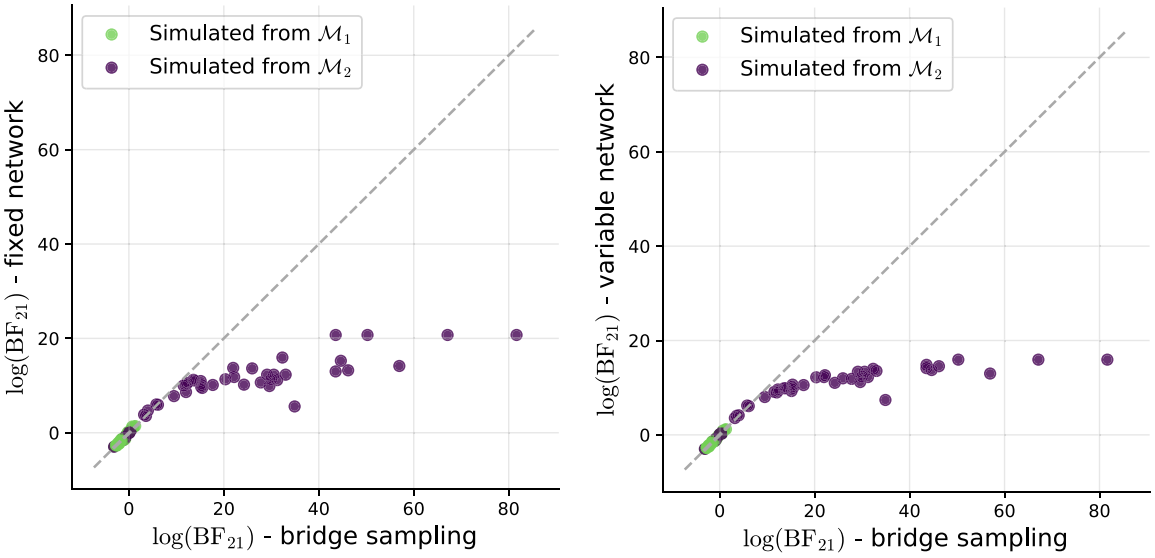


Note. ECE = expected calibration error; SBC = simulation-based calibration. See the online article for the color version of this figure.

(*Appendices continue*)

Figure B4

Validation Study 1: Full Comparison Results for the Log BFs (All 100 Test Data Sets)



Note. BFs = Bayes factors. See the online article for the color version of this figure.

Appendix C

Validation Study 2 Details

Here, we provide details on our model specifications and prior choices. We reformulate the observation-level structure of the MPT model as a binomial instead of a multinomial process to obtain identical response generation implementations for both models

$$x^h_{mn} \sim \text{Bernoulli}(h_m) \quad \text{for } n = 1, \ldots, N_m, \tag{C1}$$

$$x^f_{mn} \sim \text{Bernoulli}(f_m) \quad \text{for } n = 1, \ldots, N_m, \tag{C2}$$

where h_m denotes the probability of detecting an old item as old ("hit") and f_m denotes the probability of detecting a new item as old ("false alarm"). The generating processes of these probabilities with our distributional choices are described in Tables C1 and C2 for the SDT model and Tables C3 and C4 for the MPT models. Figure C1 shows the prior predictive patterns of hit rates and false alarm rates arising from 5,000 simulated data sets for each model.

Figure C2 presents the log BFs approximated by bridge sampling and the neural network, showing slight discrepancies in areas of extreme evidence. In contrast to the nested models in Validation Study 1, the SDT and MPT models being nonnested allows for extreme evidence for both models.

Table C1

Validation Study 2: Hyperprior Distributions of the SDT Model

Parameter	Symbol	Prior distribution
Probit-transformed hit probability	$\mu_{h'}$	Normal(1, 0.5)
	$\sigma_{h'}$	Gamma(1, 1)
Probit-transformed false alarm probability	$\mu_{f'}$	Normal(−1, 0.5)
	$\sigma_{f'}$	Gamma(1, 1)

Note. SDT = signal detection theory.

(*Appendices continue*)

Table C2
Validation Study 2: Group-Level Prior Distributions and Transformations of the SDT Model

Parameter	Symbol	Prior distribution/transformation
Probit-transformed hit probability	h'_m	Normal($\mu_{h'}$, $\sigma_{h'}$)
Probit-transformed false alarm probability	f'_m	Normal($\mu_{f'}$, $\sigma_{f'}$)
Hit probability	h_m	$\Phi(h'_m)$
False alarm probability	f_m	$\Phi(f'_m)$

Note. SDT = signal detection theory.

Table C3
Validation Study 2: Hyperprior Distributions and Transformations of the MPT Model

Parameter	Symbol	Prior distribution/transformation
Probit-transformed recognition probability	$h_{d'}$	Normal(0, 0.25)
Probit-transformed guessing probability	$h_{g'}$	Normal(0, 0.25)
Covariance matrix	$\lambda_{d'}$	Uniform(0, 2)
	$\lambda_{g'}$	Uniform(0, 2)
	Q	InvWishart(3, $\mathbb{I}$)
	Σ	Diag($\lambda_{d'}$, $\lambda_{g'}$) Q Diag($\lambda_{d'}$, $\lambda_{g'}$)

Note. MPT = multinomial processing tree.

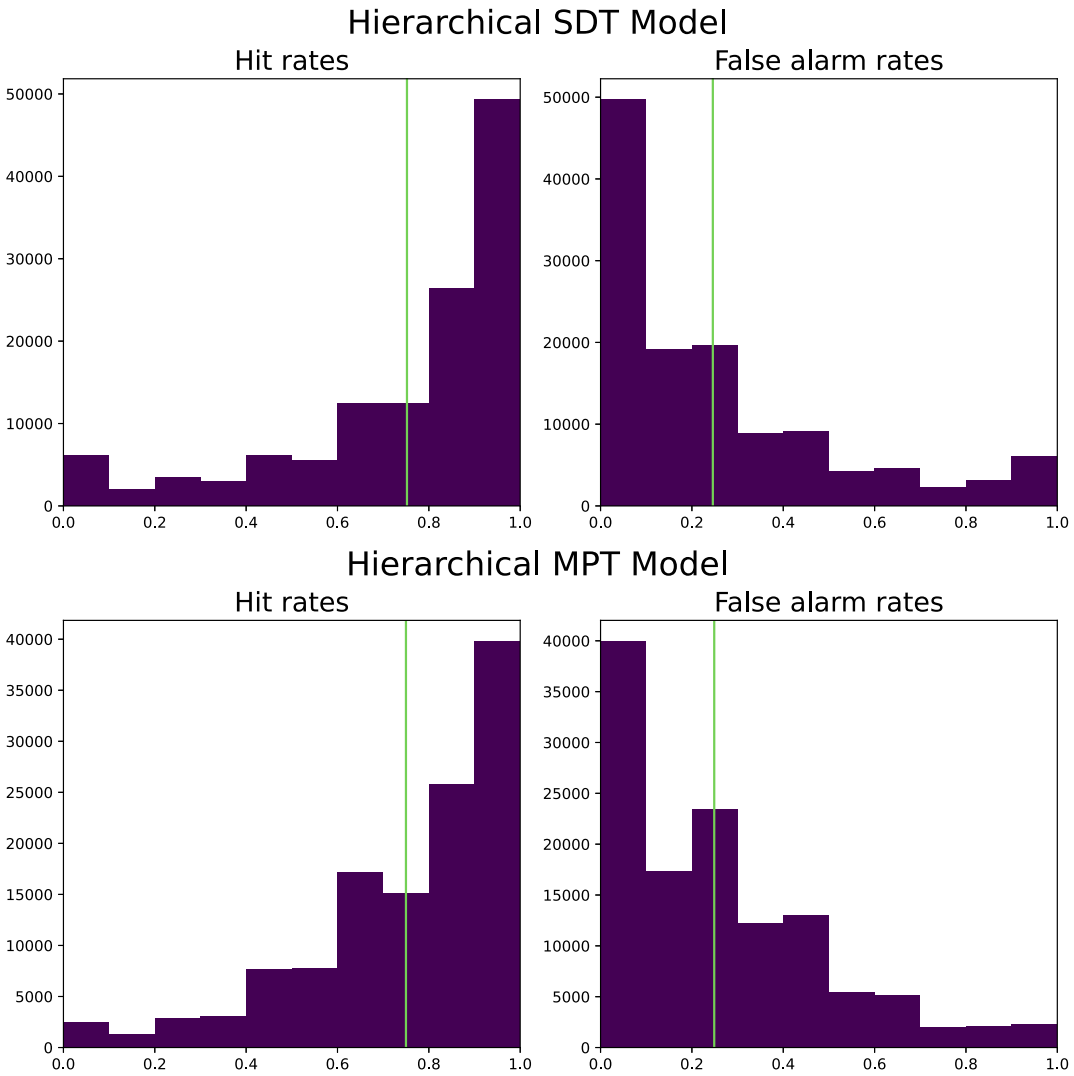
Table C4
Validation Study 2: Group-Level Prior Distributions and Transformations of the MPT Model

Parameter	Symbol	Prior distribution/transformation
Probit-transformed recognition probability	d'_m	$\text{Normal}\left(\begin{bmatrix}\mu_{d'}\\ \mu_{g'}\end{bmatrix}, \Sigma\right)$
Probit-transformed guessing probability	g'_m	
Recognition probability	d_m	$\Phi(d'_m)$
Guessing probability	g_m	$\Phi(g'_m)$
Hit probability	h_m	$d_m + (1 - d_m)*g_m$
False alarm probability	f_m	$(1 - d_m)*g_m$

Note. MPT = multinomial processing tree.

(*Appendices continue*)

Figure C1
Validation Study 2: Prior Predictive Checks for the SDT and the MPT Model

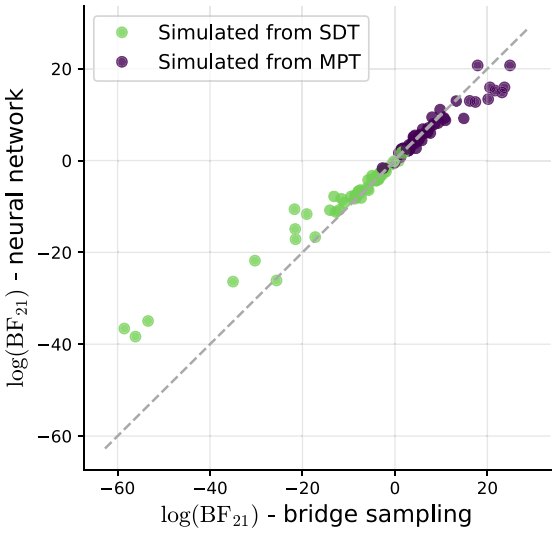


Note. The vertical (green) lines indicate the mean. SDT = signal detection theory; MPT = multinomial processing tree. See the online article for the color version of this figure.

(*Appendices continue*)

Figure C2

Validation Study 2: Full Comparison Results for the Log BFs (All 100 Test Data Sets)



Note. SDT = signal detection theory; MPT = multinomial processing tree; BFs= Bayes factors. See the online article for the color version of this figure.

Appendix D

Application Details

Parameter Priors and Prior Predictive Checks

We base our priors upon the comprehensive collection of diffusion model parameter estimates by Tran et al. (2021). For the Lévy flight models, $\mathcal{M}_2$ and $\mathcal{M}_4$, we inform the prior on the additional α parameter by the estimates for comparable tasks (those completed under speed instructions) by Voss et al. (2019). For the intertrial variability parameters included in $\mathcal{M}_3$ and $\mathcal{M}_4$, we follow the nonhierarchical priors that Wiecki et al. (2013) suggest to use in hierarchical drift-diffusion models, but choose a nonpooling approach with individual parameters instead of a complete-pooling approach. Table D1 contains the hyperprior choices and Table D2 the group-level priors.

To ensure that the informed priors for our HMs accurately reflect prior knowledge at both levels, we conduct prior predictive checks based on 10,000 simulations (displayed in Figures D1–D3).

Robustness Against Artificial Noise

Here, we inspect the stability of our neural network against additional noise injection. Figure D4 displays the model comparison results as increasing percentages of trials per participant are artificially masked as missing. We repeat the random masking of trials 100 times per percentage step to assess the sensitivity of the results to specific parts of the empirical data. Consistent with our main results, there is a clear separation between low evidence for $\mathcal{M}_1$ and $\mathcal{M}_2$ and substantial evidence for $\mathcal{M}_3$ and $\mathcal{M}_4$ across all settings. Despite our network being trained on the empirical amount of missing data, 3.17% over both tasks, we observe rank stability of the model comparison results up until 25% missing data per participant.

(*Appendices continue*)

Table D1

Real-Data Application: Hyperprior Distributions of the Evidence Accumulation Models

Parameter	Symbol	Prior distribution
Threshold separation	μ_a	Normal(5, 1)
	σ_a	$\text{Normal}_+(0.4, 0.15)$
Relative starting point	μ_{zr}	Normal(0, 0.25)
	σ_{zr}	$\text{Normal}_+(0, 0.05)$
Drift rate for blue/nonword stimuli	μ_{v_0}	Normal(5, 1)
	σ_{v_0}	$\text{Normal}_+(0.5, 0.25)$
Drift rate for orange/word stimuli	μ_{v_1}	Normal(5, 1)
	σ_{v_1}	$\text{Normal}_+(0.5, 0.25)$
Nondecision time	μ_{t_0}	Normal(5, 1)
	σ_{t_0}	$\text{Normal}_+(0.1, 0.05)$
Stability parameter of the noise distribution	μ_α	Normal(1.65, 0.15)
	σ_α	$\text{Normal}_+(0.3, 0.1)$

Table D2

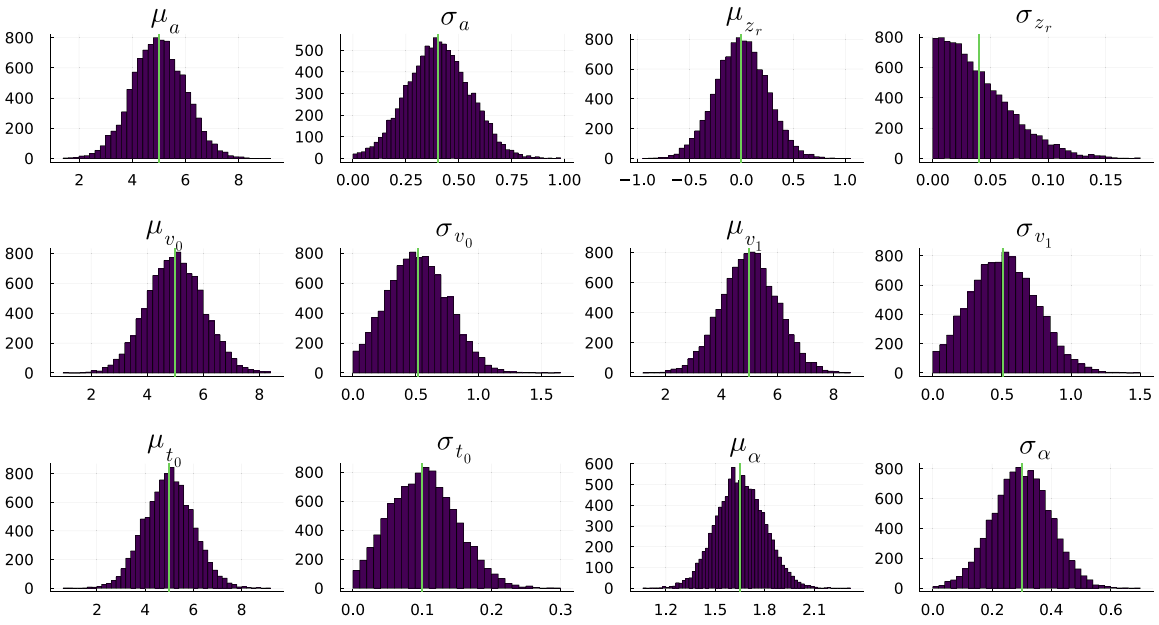
Real-Data Application: Group-Level Prior Distributions of the Evidence Accumulation Models

Parameter	Symbol	Prior distribution
Threshold separation	a_m	Gamma(μ_a, σ_a)
Relative starting point	zr_m	invlogit(Normal(μ_{zr}, σ_{zr}))
Drift rate for blue/nonword stimuli	$v0_m$	$-$Gamma(μ_{v_0}, σ_{v_0})
Drift rate for orange/word stimuli	$v1_m$	Gamma(μ_{v_1}, σ_{v_1})
Nondecision time	$t0_m$	Gamma(μ_{t_0}, σ_{t_0})
Stability parameter of the noise distribution	α_m	TruncatedNormal(μ_α, σ_α, 1, 2)
Intertrial variability of starting point	s_{z_m}	Beta(1, 3)
Intertrial variability of drift	s_{v_m}	$\text{Normal}_+(0, 2)$
Intertrial variability of nondecision time	s_{t_m}	$\text{Normal}_+(0, 0.3)$

Note. $\text{Normal}_+(\cdot)$ denotes a zero-truncated normal distribution that only allows for positive values. TruncatedNormal$(\cdot)$ denotes a truncated normal distribution with the lower and upper limits given by the last two values.

Figure D1

Real-Data Application: Prior Predictive Checks for the Hyperpriors in the Comparison of Evidence Accumulation Models

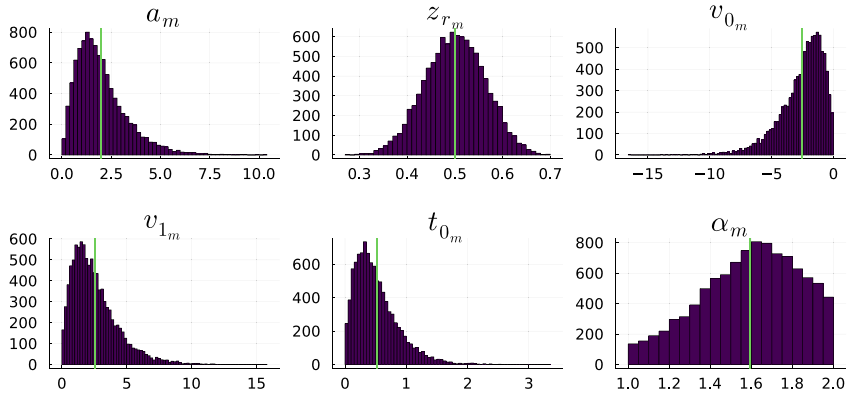


Note. The vertical (green) lines indicate the mean. See the online article for the color version of this figure.

(*Appendices continue*)

Figure D2

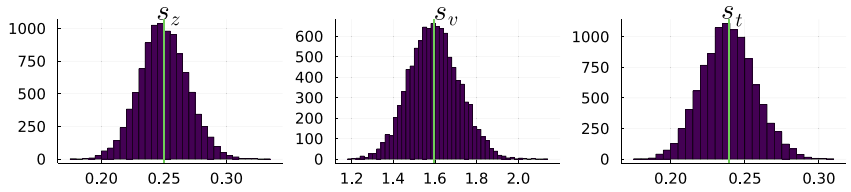
Real-Data Application: Prior Predictive Checks for the Hierarchical Group-Level Priors in the Comparison of Evidence Accumulation Models



Note. The vertical (green) lines indicate the mean. See the online article for the color version of this figure.

Figure D3

Real-Data Application: Prior Predictive Checks for the Nonhierarchical Group-Level Priors in the Comparison of Evidence Accumulation Models

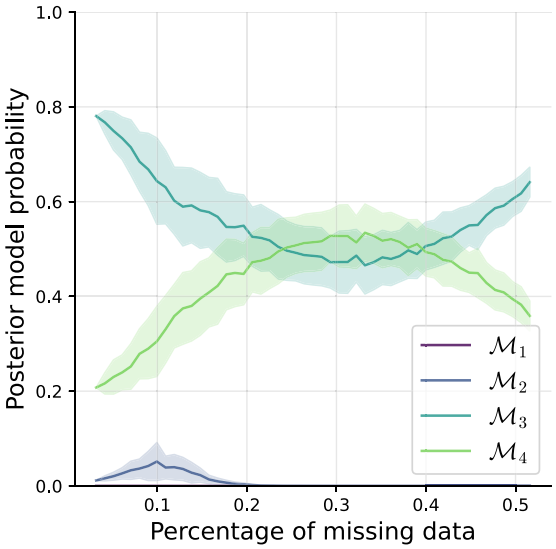


Note. The vertical (green) lines indicate the mean. See the online article for the color version of this figure.

(*Appendices continue*)

Figure D4

Real-Data Application: Robustness of the Model Comparison Results Against Increasing Amounts of Artificially Injected Random Noise



Note. The lines represent the average probabilities of 100 repetitions per percentage step (in each repetition masking a random subset of the empirical data), whereas the shaded areas indicate the standard deviation between these repetitions. See the online article for the color version of this figure.

Received February 6, 2023
Revision received July 11, 2023
Accepted December 14, 2023 ■

Publication II

B

Publication	**Link to online full-text**

Elsemüller, L., Olischläger, H., Schmitt, M., Bürkner, P. C., Köthe, U., & Radev, S. T. (2024). Sensitivity-aware amortized Bayesian inference. *Transactions on Machine Learning Research*.

The reprint of the publication can be found in the following.

Published in Transactions on Machine Learning Research (08/2024)

Sensitivity-Aware Amortized Bayesian Inference

Lasse Elsemüller *lasse.elsemueller@gmail.com*
Heidelberg University

Hans Olischläger
Heidelberg University

Marvin Schmitt
University of Stuttgart

Paul-Christian Bürkner
TU Dortmund University

Ullrich Köthe
Heidelberg University

Stefan T. Radev *stefan.radev93@gmail.com*
Rensselaer Polytechnic Institute

Reviewed on OpenReview: *https://openreview.net/forum?id=Kxtpa9rvM0*

Abstract

Sensitivity analyses reveal the influence of various modeling choices on the outcomes of statistical analyses. While theoretically appealing, they are overwhelmingly inefficient for complex Bayesian models. In this work, we propose sensitivity-aware amortized Bayesian inference (SA-ABI), a multifaceted approach to efficiently integrate sensitivity analyses into simulation-based inference with neural networks. First, we utilize weight sharing to encode the structural similarities between alternative likelihood and prior specifications in the training process with minimal computational overhead. Second, we leverage the rapid inference of neural networks to assess sensitivity to data perturbations and preprocessing steps. In contrast to most other Bayesian approaches, both steps circumvent the costly bottleneck of refitting the model for each choice of likelihood, prior, or data set. Finally, we propose to use deep ensembles to detect sensitivity arising from unreliable approximation (e.g., due to model misspecification). We demonstrate the effectiveness of our method in applied modeling problems, ranging from disease outbreak dynamics and global warming thresholds to human decision-making. Our results support sensitivity-aware inference as a default choice for amortized Bayesian workflows, automatically providing modelers with insights into otherwise hidden dimensions.

1 Introduction

Statistical inference aims to extract meaningful insights from empirical data through a series of analytical procedures. Acknowledging that each of these procedures involves a myriad of implicit choices and assumptions, *any single analysis hides an iceberg of uncertainty* (Wagenmakers et al., 2022). We consider *sensitivity analysis* as a formal approach to shed light on this very iceberg of uncertainty.

For instance, global warming forecasts can change depending on the assumed earth system model. In other words: Climate change analyses can be sensitive to the underlying observation model (i.e., *likelihood*; see **Experiment 2**). Yet, the likelihood is not the only model component that can induce sensitivity. The prior assumptions, the approximation algorithm, and the specifics of the collected data contribute further uncertainty to the results (Bürkner et al., 2022).

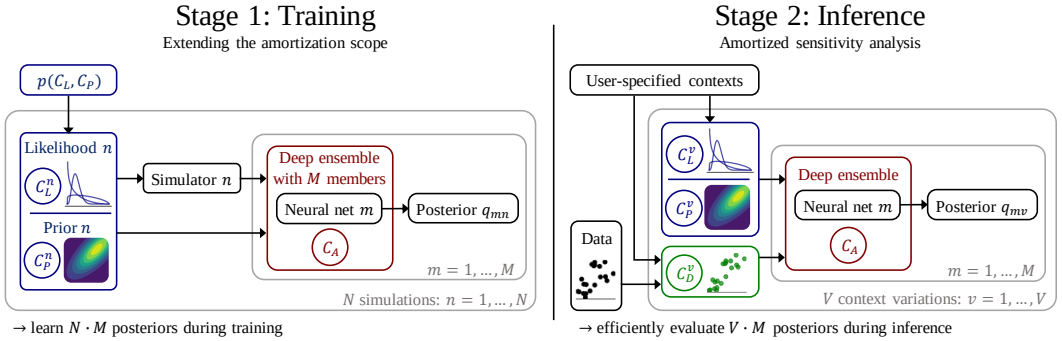


Figure 1: Our proposed approach for sensitivity-aware amortized Bayesian inference (SA-ABI). **Stage 1:** During training, a distribution $p(C_L, C_P)$ over plausible likelihood and prior choices is encoded via context variables C_L and C_P in a deep ensemble of neural approximators. **Stage 2:** During inference, we cast costly model refits as a near-instant neural network prediction task conditioned on user-specified context C. Our amortized neural approach unlocks fast large-scale sensitivity analyses of all components in a Bayesian model: likelihood (C_L), prior (C_P), data (C_D), and approximator (C_A). **Experiment 3** uses $V = 8\,100$ variations in prior and data alongside $M = 20$ deep ensemble members. The resulting amortized sensitivity analysis encompassing $V \cdot M = 162\,000$ approximate posteriors would have been infeasible with existing methods.

Classical sensitivity analyses rely on costly model refitting under each configuration and quickly become infeasible for both likelihood-based (e.g., MCMC; Neal, 2011) and simulation-based inference (SBI, Cranmer et al., 2020).

Recently, SBI methods have been accelerated through amortized Bayesian inference (ABI; Radev et al., 2020; Gonçalves et al., 2020; Avecilla et al., 2022), where neural networks learn probabilistic inference tasks and compensate for the training effort with rapid inference on many unseen data sets. As we demonstrate in this paper, this directly enables a large-scale assessment of data sensitivity. However, assessing likelihood, prior, and approximator sensitivity remains extremely challenging in standard ABI applications, which may be costly to train even for a single configuration. To address this gap, we investigate *sensitivity-aware amortized Bayesian inference* (SA-ABI) and demonstrate that it unlocks highly efficient and multifaceted sensitivity analyses in realistic ABI applications (cf. Figure 1). Our main contributions are:

1. We conceptualize sensitivity via implicit context variables and integrate established methods for sensitivity analysis into amortized Bayesian inference;
2. We investigate a context-aware neural architecture to quantify likelihood and prior sensitivity at inference time with minimal computational overhead and no notable loss of accuracy;
3. We assess approximator sensitivity via deep ensembles and data sensitivity via the near-instant ABI inference;
4. We demonstrate the utility of SA-ABI for Bayesian parameter estimation as well as Bayesian model comparison in three real-world scenarios of scientific interest, investigating sensitivity under up to 162 000 configurations.

2 Background

What follows is a brief overview of Bayesian parameter estimation, model comparison, amortized Bayesian inference, and learnable summary statistics. Readers familiar with these topics can safely jump directly to **Section 3**.

2.1 Bayesian Inference

Bayesian Parameter Estimation In Bayesian parameter estimation, the key quantity is the *posterior distribution*,

$$p(\boldsymbol{\theta} \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\boldsymbol{x})}, \tag{1}$$

which combines the likelihood $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ with prior information about parameter distributions $p(\boldsymbol{\theta})$, normalized by the (analytically intractable) marginal likelihood $p(\boldsymbol{x})$.

Bayesian Model Comparison In many scientific applications, no single generative model can provide *the* ultimate explanation for a data set $\boldsymbol{x}$. Instead, a set of models $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_J\}$ is plausible. Bayesian model comparison aims to find the "best" model within $\mathcal{M}$. In (prior-predictive) Bayesian model comparison, a model's *marginal likelihood* $p(\boldsymbol{x} \mid \mathcal{M}_j)$ now takes the central role:

$$p(\boldsymbol{x} \mid \mathcal{M}_j) = \int p(\boldsymbol{x} \mid \boldsymbol{\theta}, \mathcal{M}_j)\, p(\boldsymbol{\theta} \mid \mathcal{M}_j)\, \mathrm{d}\boldsymbol{\theta}. \tag{2}$$

Marginalizing the likelihood over the parameter space automatically encodes Occam's razor through a preference for models with limited prior predictive flexibility (MacKay, 2003). The *posterior model probabilities* of competing models can be computed as

$$p(\mathcal{M}_j \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid \mathcal{M}_j)\, p(\mathcal{M}_j)}{\sum_{\mathcal{M}} p(\boldsymbol{x} \mid \mathcal{M})\, p(\mathcal{M})}, \tag{3}$$

where $p(\mathcal{M})$ is the prior distribution over the model space.

2.2 Simulation-Based Inference

Both Bayesian parameter estimation and model comparison have traditionally been limited by the ability to efficiently evaluate a model's likelihood density $p(\boldsymbol{x} \mid \boldsymbol{\theta})$. Likelihood-based methods (e.g., MCMC) assume that the distributional family of the likelihood is explicitly known and can be evaluated analytically or numerically for any pair $(\boldsymbol{x}, \boldsymbol{\theta})$. Differently, simulation-based approaches only require simulations from a simulation program G,

$$\boldsymbol{x} = G(\boldsymbol{\theta}, \boldsymbol{\xi}) \quad \text{with} \quad \boldsymbol{\xi} \sim p(\boldsymbol{\xi} \mid \boldsymbol{\theta}), \boldsymbol{\theta} \sim p(\boldsymbol{\theta}), \tag{4}$$

with latent program states or "outsourced" noise $\boldsymbol{\xi}$. A single execution of such a program corresponds to generating samples from the Bayesian joint model $(\boldsymbol{\theta}, \boldsymbol{x}) \sim p(\boldsymbol{\theta}, \boldsymbol{x})$, since the execution paths of the simulation program define an *implicit likelihood* (e.g., Cranmer et al., 2020; Diggle & Gratton, 1984; Marin et al., 2012)

$$p(\boldsymbol{x} \mid \boldsymbol{\theta}) = \int \delta(\boldsymbol{x} - G(\boldsymbol{\theta}, \boldsymbol{\xi}))\, p(\boldsymbol{\xi} \mid \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\xi}, \tag{5}$$

where δ is the Dirac delta function. However, the above equation (5) is analytically intractable for any simulation program of practical interest, turning simulation-based inference into a computational challenge.

2.3 Amortized Bayesian Inference

Amortized Bayesian inference (ABI) leverages simulations from G to solve Bayesian inference tasks with neural networks in real time after an initial training phase. To achieve this, neural networks learn to encode the relationship between simulated data and model states during training. As a result, costly probabilistic inference is replaced with a neural network prediction task. For parameter estimation, generative neural networks act as conditional neural density approximators of the parameter posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ (Greenberg et al., 2019; Radev et al., 2020). Model comparison, on the other hand, can be framed as a probabilistic classification problem which is addressed with discriminative neural networks to approximate posterior model probabilities $p(\mathcal{M} \mid \boldsymbol{x})$ (Pudlo et al., 2016; Radev et al., 2021a).

2.4 End-to-end Summary Statistics

Common neural density estimators hinge on fixed-length vector-valued inputs – a requirement that is violated by widespread data formats such as sets of $i.i.d.$ observations or time series. To this end, previous research explored ways to learn end-to-end summary statistics for flexible adaption to the individual data structure and inference task (Chan et al., 2018; Wiqvist et al., 2019; Radev et al., 2020; Chen et al., 2020; 2023). In a nutshell, a summary network $h_{\boldsymbol{\psi}}$ compresses input data $\boldsymbol{x}$ of variable size to a fixed-length vector of learned summary statistics $h_{\boldsymbol{\psi}}(\boldsymbol{x})$ by exploiting probabilistic symmetries in the data (e.g., permutation-invariant networks for exchangeable data; Bloem-Reddy & Teh, 2020). The resulting *embedding* is fed to an inference network $f_{\boldsymbol{\phi}}$ that approximates the posterior (e.g., with a conditional normalizing flow). The summary network $h_{\boldsymbol{\psi}}$ and the inference network $f_{\boldsymbol{\phi}}$ are simultaneously trained end-to-end in order to learn optimal summary statistics for the inference task.

3 Methods

3.1 Extending the Amortization Scope

Sensitivity Sources as Context Variables The PAD framework (Bürkner et al., 2022) defines a Bayesian model as a combination of the joint **P**robability distribution $p(\boldsymbol{\theta}, \boldsymbol{x} \mid \mathcal{M})$, which can be factorized into likelihood $p(\boldsymbol{x} \mid \boldsymbol{\theta}, \mathcal{M})$ and prior $p(\boldsymbol{\theta} \mid \mathcal{M})$, the posterior **A**pproximator, and the observed **D**ata $\boldsymbol{x}_{\text{obs}}$. Building on this decomposition, we define *sensitivity* to a component of a Bayesian model as the extent of change in inferential results induced by perturbations in any of these components (Bürkner et al., 2022). We will refer to the opposite of sensitivity as *robustness*: When an inference procedure is robust, it is not sensitive to changes in model components.

To enable systematic and comprehensive investigations of sensitivity, we consider sources of perturbations as context variables that implicitly shape inferential results (see Figure 1) and can be cheaply varied as conditioning variables (or hyperparameters) in our simulation-based approach. In contrast, standard Bayesian workflows treat context variables as fixed and thus cannot typically investigate their effect explicitly without re-doing the entire analysis.

In the following, we denote context variables as C_L (likelihood), C_P (prior), C_A (approximator), C_D (data), and refer to the entirety of context variables as $C = (C_L, C_P, C_A, C_D)$. Accordingly, we refer to posteriors with explicitly encoded context variables as $p(\boldsymbol{\theta} \mid \boldsymbol{x}, C)$ for parameter estimation and $p(\mathcal{M}_j \mid \boldsymbol{x}, C)$ for model comparison.

Sensitivity-Aware Training Before discussing key facets of practical sensitivity analysis, we describe extensions to standard simulation-based training that allow these analyses to be performed *efficiently*. Specifically, we eliminate the necessity of retraining neural approximators for every aspect of a sensitivity analysis by incorporating the **T**raining contexts $C_T = (C_L, C_P)$ into the network's amortization scope (Radev, 2021). Since we realize C_A via deep ensembles and C_D only during inference (see **Section 3.2**), neither of these contexts influences the training objective of single ensemble members.

For sensitivity-aware parameter estimation (PE), we incorporate the context C_T into the standard negative log-posterior objective via

$$\mathcal{L}^{\text{PE}}(\boldsymbol{\phi}, \boldsymbol{\psi}; C_T) = \mathbb{E}\Big[- \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}), C_T)\Big]. \tag{6}$$

The expectation $\mathbb{E}$ is here taken over a contextualized joint Bayesian model $p(\boldsymbol{\theta}, \boldsymbol{x} \mid C_T)$, which produces tuples of training parameters and corresponding data $(\boldsymbol{\theta}, \boldsymbol{x})$ for the given context C_T.

Analogously, for sensitivity-aware Bayesian model comparison (BMC), we can target the approximate posterior model probability $q_{\boldsymbol{\phi}}(\mathcal{M} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}), C_T)$ via the cross-entropy

$$\mathcal{L}^{\text{BMC}}(\boldsymbol{\phi}, \boldsymbol{\psi}; C_T) = \mathbb{E}\Big[- \sum_{j=1}^{J} \mathbb{I}_{\mathcal{M}_j} \log q_{\boldsymbol{\phi}}(\mathcal{M}_j \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}), C_T)\Big], \tag{7}$$

where the expectation $\mathbb{E}$ is taken with respect to a contextualized generative mixture of Bayesian models $p(\mathcal{M}_j \mid C_T)\, p(\boldsymbol{x} \mid \mathcal{M}_j, C_T)$ producing tuples of model indices and associated simulated data $(\mathcal{M}_j, \boldsymbol{x})$. The indicator function $\mathbb{I}_{\mathcal{M}_j}$ denotes a one-hot encoding for the true model index, i.e., $\mathbb{I}_{\mathcal{M}_j} = 1$ if $\mathcal{M}_j$ is the true model.

To achieve the desired amortization over any set of context variables C_T, we define a prior distribution $p(C_T)$ over the domains of C_T and minimize the *context-aware* (CA) loss:

$$\mathcal{L}^{\text{CA}}(\boldsymbol{\phi}, \boldsymbol{\psi}) = \mathbb{E}\big[\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\psi}; C_T)\big], \tag{8}$$

where the (outer) expectation runs over $p(C_T)$ and $\mathcal{L}$ can be either $\mathcal{L}^{\text{PE}}$ or $\mathcal{L}^{\text{BMC}}$. We believe that uniform distributions are a reasonable choice of $p(C_T)$ for sensitivity analyses and employ them in all experiments. Nevertheless, $p(C_T)$ can be tailored to specific modeling needs, such as giving more weight to approximating a preferred baseline setting. In practice, we approximate Eq. 8 using standard mini-batch gradient descent over a finite data set $\mathcal{D} = \{C_T, \boldsymbol{\theta}, \boldsymbol{x}\}$ for parameter estimation, or $\mathcal{D} = \{C_T, \mathcal{M}_j, \boldsymbol{x}\}$ for model comparison.

Our approach seamlessly generalizes to other *strictly proper losses* (Gneiting & Raftery, 2007) which can be used as training objectives for amortized inference (Pacchiardi & Dutta, 2021). For the sake of generality, we can introduce a function S that quantifies the fidelity of a conditional distribution $q_{\boldsymbol{\phi}} \in \mathcal{Q}$ for predicting a target quantity $\boldsymbol{y} \in \mathcal{Y}$

Table 1: Overview of our taxonomy for sensitivity in Bayesian inference via context variables. The rightmost column conveys that our context-aware (CA) loss function $\mathcal{L}^{\text{CA}}$ in Eq. 8 enables the amortization over both likelihood (C_L) and prior (C_P) contexts during training.

	Context	**Sensitivity source example**	**Implementation**	$\mathcal{L}^{\text{CA}}$ **required?**
C_L	Likelihood	Structural model assumptions	Multiple simulator configurations	✓
C_P	Prior	Expert knowledge	Multiple prior configurations	✓
C_A	Approximator	Simulation gaps	Deep ensemble	✗
C_D	Data	Influential observations	Multiple data configurations	✗

(Gneiting & Raftery, 2007). Thus, $S : \mathcal{Q} \times \mathcal{Y} \to \mathbb{R}$ is a function of some $q_{\boldsymbol{\phi}}$ and $\boldsymbol{y}$ (e.g., $\boldsymbol{\theta}$ or $\mathcal{M}_j$) which can easily be written to incorporate arbitrary conditions for $q_{\boldsymbol{\phi}}$, such as summarized data $h_{\boldsymbol{\psi}}(\boldsymbol{x})$ and context C_T, resulting in $S(q_{\boldsymbol{\phi}}(\boldsymbol{y} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}), C_T), \boldsymbol{y})$. Ideally, we would like to treat the *expected score* as an optimization objective

$$\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\psi}; C_T) = \mathbb{E}_{p^*(\boldsymbol{x}, \boldsymbol{y})}\Big[S(q_{\boldsymbol{\phi}}(\boldsymbol{y} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}), C_T), \boldsymbol{y})\Big], \tag{9}$$

which for *strictly proper scoring functions* S would guarantee $q_{\boldsymbol{\phi}}(\boldsymbol{y} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}), C_T) = p^*(\boldsymbol{y} \mid \boldsymbol{x})$ under perfect convergence (Gneiting & Raftery, 2007; Pacchiardi & Dutta, 2021). However, we usually cannot directly access the analytic expectation over the unknown true data-generating distribution $p^*(\boldsymbol{x}, \boldsymbol{y})$. Thus, to achieve tractable amortization, we use the model-implied distribution $p(\boldsymbol{x}, \boldsymbol{y} \mid \mathcal{M})$ as a proxy for the (unknown) true data generating process $p^*(\boldsymbol{x}, \boldsymbol{y})$ and optimize the former objective in expectation over the simulator outputs (i.e., one or more Bayesian probabilistic models).

With this expansion of the amortization scope, we achieve amortization across all sensitivity dimensions $C = (C_L, C_P, C_A, C_D)$ during inference. What that means in practice is that, at inference time, we can simply "turn a knob" on any of the sensitivity dimensions and obtain the resulting posterior in an instant. A natural question that immediately arises is *whether the resulting sensitivity-aware posterior is less accurate than the corresponding fixed-context posterior*. Intuitively, the answer depends on the sampling diversity of the contextualized joint model $p(\boldsymbol{x}, \boldsymbol{y}, C_T)$ and the potential for reaping the benefits of weight sharing: If the associated likelihood and prior variations instantiate generative models with wildly different behaviors, weight sharing may not be advantageous, resulting in diminishing returns from amortized training over the context variables C_T. Fortunately, the set of plausible choices for a given modeling problem typically leads to similar generative patterns, so that weight sharing is much more efficient than separate approximation. Indeed, our experiments demonstrate this for several representative model families even under small simulation budgets. Nevertheless, if amortization over very different simulators is desired, we recommend increasing the expressiveness of the neural approximator, the simulation budget, and the allotted training time. The following section describes sensitivity sources and actionable manipulation strategies for each sensitivity dimension (see Table 1 for an overview).

3.2 Sources of Sensitivity

Likelihood and Prior Sensitivity Varying likelihood context variables are ubiquitous in simulation-based inference. Structural decisions within the simulator(s) may constitute context variables (e.g., the underlying scientific model, see **Experiment 2**) or exogenous experimental factors, such as design matrices, indicator variables, or time scales. Typical examples for prior context might be as simple as the scales of prior distributions, or as complex as different experts eliciting discrete sets of qualitative (e.g., non-probabilistic) domain knowledge. Moreover, both likelihood and prior can be continuously tempered to strengthen or weaken their influence. For example, power-scaling exponentiates densities with a parameter $\gamma > 0$, resulting in $p(\boldsymbol{\theta})^\gamma$ for prior power scaling and $p(\boldsymbol{y} \mid \boldsymbol{\theta})^\gamma$ for likelihood power scaling (Kallioinen et al., 2021).

We make all known pieces of likelihood and prior context explicit by incorporating C_L and C_P into the generative model. This enables amortization over these context variables, which drastically increases the generalization space of the trained neural approximator. During inference, the specific set of likelihood and prior can simply be selected by passing the respective C_L and C_P configurations. Amortization over the context space leverages structural similarities

between context configurations via weight sharing. Compared to separate training, this *minimizes the associated computational cost and is especially beneficial whenever only finite training data is available* (see **Experiment 2**).

Approximator Sensitivity We define approximator sensitivity as the variability of inferential results due to the approximation method employed. To isolate approximator sensitivity in ABI, it seems helpful to (i) distinguish between a *closed world* (i.e., simulations) and *open world* (i.e., empirical data) setting; and (ii) realize an approximator context C_A via a deep ensemble of M equally configured and independently trained neural networks $\{(\boldsymbol{\phi}^{(m)}), \boldsymbol{\psi}^{(m)}\}_{m=1}^{M}$. [1]

In the closed-world setting, ground truth values for the approximation targets $\boldsymbol{y}$ (i.e., $\boldsymbol{\theta}$ or $\mathcal{M}_j$) are available. Thus, we can readily validate amortized neural approximators on thousands of simulated data sets from the model(s) under consideration. We propose to additionally measure performance variability between the ensemble members to detect approximator sensitivity due to finite training or suboptimal convergence. After validating the approximator in the closed world, we consider the open-world setting, where the true data-generating process is unknown.

As a simulation-based method, ABI assumes that simulations are a faithful representation of a system's real behavior (Dellaporta et al., 2022). Hence, *simulation gaps*, where atypical data violate this assumption, threaten its credibility (Schmitt et al., 2023; Cannon et al., 2022). Simulation gaps can be considered to cause an out-of-distribution (OOD) setting at inference time: For example, Cannon et al. (2022) observed that misspecification-induced simulation gaps result in neural approximators exhibiting the typical OOD behavior of unstable predictions (Ji et al., 2022; Shamir et al., 2021; Liu et al., 2021). Therefore, we can leverage the proven OOD detection capabilities of deep ensembles (Lakshminarayanan et al., 2017; Fort et al., 2019; Yang et al., 2022) to detect simulation gaps in ABI. Specifically, we hypothesize that *variability* across the M ensemble members in the open world *despite consistent performance in the closed world* indicates a simulation gap. Concretely, we expect a simulation gap to translate into high variability in the predictive distribution of the unknown targets $\boldsymbol{y}$ given empirical data $\boldsymbol{x}_{\text{obs}}$,

$$\tilde{q}(\boldsymbol{y} \mid \boldsymbol{x}_{\text{obs}}) = \mathbb{E}_{p(\boldsymbol{\phi}, \boldsymbol{\psi} \mid \mathcal{D})} \left[q_{\boldsymbol{\phi}}(\boldsymbol{y} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}_{\text{obs}})) \right] \tag{10}$$

which is approximated by the deep ensemble (Lakshminarayanan et al., 2017) and can be augmented with arbitrary context C_T.

When we train a deep ensemble with M members, we clearly need to repeat the training loop M times. Crucially though, we can simulate a single training set upfront and then re-use the simulated training data for all ensemble members. This not only reduces the stochastic dependence by keeping the training data constant but also drastically reduces the computational cost in most realistic tasks where simulations are expensive. Finally, our ensemble approach can be easily extended to combine information from all ensemble members for potentially more accurate inference (e.g., via simulation-based stacking, Yao et al., 2023) or investigate hyperparameter sensitivity via hyperparameter ensembles (Wenzel et al., 2020). Hyperparameters that are particularly relevant in SBI include the architecture of the summary network (e.g., inductive bias induced by the architecture, number of learned summary statistics), the choice of inference network (e.g., architecture, number of trainable weights), and common hyperparameters that are ubiquitous in deep learning (e.g., learning rate, regularization).

Data Sensitivity Statistical inference relies on finite samples to draw conclusions about populations. As such, any analysis is influenced by the properties of this particular sample. For instance, analysis outcomes might radically change under different preprocessing choices, such as handling extreme or missing data points, even if these choices only affect a small subset (Simmons et al., 2011; Broderick et al., 2023).

There are two straightforward strategies to assess this data sensitivity: (i) To assess a disproportionally large influence of single data points (also known as influential observations), a context C_D of alternative data manifestations can be realized via small perturbations of the empirical data set, for instance via bootstrapping or leave-one-out folds; (ii) to analyze the effect of specific preprocessing decisions, we can generate data set variations for all combinations of reasonable decisions, which in turn constitutes C_D. For example, Kristanto et al. (2024) identify 17 debatable preprocessing steps with 102 choice options in graph-based fMRI analysis, resulting in hundreds of potential manifestations of the final data set.

[1] Although Bayesian neural networks offer appealing uncertainty quantification properties, we focus on deep ensembles for their practical implementation advantages (Lakshminarayanan et al., 2017; Wilson & Izmailov, 2020).

Both strategies require a large amount of model refits, which is computationally infeasible for MCMC or non-amortized simulation-based approximators. In contrast, ABI methods can amortize across data sets of variable sizes (Radev et al., 2020), enabling rapid inference on a large number of data set variations.

3.3 Evaluating Sensitivity

Quantitative Sensitivity We can easily *quantify* sensitivity via a divergence metric $\mathbb{D}$ between target probability densities (Kallioinen et al., 2021; Roos et al., 2015): For an acceptable upper bound ϑ based on domain knowledge, a model is robust if

$$\mathbb{D}\big[p(g(\boldsymbol{y}) \mid \boldsymbol{x}, C_i) \,\big|\big|\, p(g(\boldsymbol{y}) \mid \boldsymbol{x}, C_j))\big] < \vartheta, \tag{11}$$

for arbitrary context realizations C_i and C_j, where $g(\boldsymbol{y})$ is a pushforward variable (e.g., predicted quantities) or a projection of the full target posterior onto a subset $\boldsymbol{y}' \subseteq \boldsymbol{y}$.

Measures from the family of $\mathcal{F}$-divergences offer principled metrics for $\mathbb{D}$ (Csiszár, 1964; Ali & Silvey, 1966; Liese & Vajda, 2006). In Bayesian model comparison, the model posterior containing the probabilities for each $\mathcal{M}_j$ follows a discrete categorical distribution. Thus, obtaining $\mathcal{F}$-divergences, such as the KL divergence, is straightforward (see **Experiment 3**, Figure 5b). In Bayesian parameter estimation, the posterior is typically not available as a closed-form density but as random draws. Thus, we prefer a probability integral metric, such as the maximum mean discrepancy (MMD; Gretton et al., 2012), which can be efficiently estimated from posterior samples (see Bischoff et al., 2024, for a recent discussion of other suitable choices).

Qualitative Sensitivity Although quantitative sensitivity patterns provide detailed insights, sensitivity analysis is often ultimately interested in *qualitative* robustness, i.e., invariance of analytical conclusions to the context C (Kallioinen et al., 2021; Bürkner et al., 2022). For instance, an analyst might ask whether two choices of context variables C_1 and C_2 contain a certain parameter value within a specified highest density interval (HDI), or lead to the selection of the same model $\mathcal{M}_j$. Making decisions based on a posterior distribution can be formalized via a decision function $L : \mathcal{P} \to \mathcal{A}$ which maps distributions $p \in \mathcal{P}$ (or their approximations) to possible qualitative conclusions or actions for a given problem $a \in \mathcal{A}$. Formally, qualitative robustness is expressed with the indicator function

$$R(C_i, C_j) = \begin{cases} 1 & \text{if } L\big(p(\boldsymbol{y} \,|\, \boldsymbol{x}, C_i)\big) = L\big(p(\boldsymbol{y} \,|\, \boldsymbol{x}, C_j)\big) \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

which yields 1 if a conclusion is invariant to the choice of arbitrary context realizations C_i or C_j, and 0 otherwise. Note that our definition trivially generalizes to more than two choices of context variables C.

4 Related Work

Extending the amortization scope Wu et al. (2020) proposed a variational inference (VI) algorithm that amortizes over a family of probabilistic generative models. This meta-amortized VI approach learns transferable representations and generalizes to unseen distributions within the amortized family. The dependence on an analytically tractable likelihood function makes this approach inapplicable to simulation-based inference, while sequential approaches that enable likelihood-free VI (Wiqvist et al., 2021; Glöckler et al., 2022) lack the amortization properties essential for sensitivity analysis (see Table 2). Schröder & Macke (2023) perform amortized inference on a *set of models* by combining model comparison and parameter estimation into a single mixture generative model. On a related note, Dax et al. (2021) avoid training separate neural approximators for gravitational-wave parameter estimation by conditioning on detector-noise characteristics. Our SA-ABI method integrates ideas from amortization scope extension in a unified framework, enabling amortization over *any* plausible prior, likelihood, and data configurations while also assessing approximator sensitivity.

Likelihood and prior sensitivity The posterior distribution clearly depends on the likelihood and prior, and a large body of research has studied the sensitivity to both likelihood and prior (for an overview, see Insua & Ruggeri, 2012; Depaoli et al., 2020). In non-amortized Bayesian inference, several approaches aim to avoid costly model refits by estimating the effects of local likelihood or prior perturbations on a given posterior, for example, via the infinitesimal jackknife (Giordano et al., 2018; 2019) or Pareto-smoothed importance sampling (Kallioinen et al., 2021). However,

Table 2: Comparison of the suitability of posterior approximation methods for sensitivity analysis in Bayesian inference.

	VI	MAVI	SNVI	MCMC	IJ	IS	ABI	**SA-ABI**
Can handle intractable likelihoods	✗	✗	✓	✗	✗	✗	✓	✓
Amortized likelihood & prior sensitivity	✗	✓	✗	✗	✓	✓	✗	✓
Amortized data sensitivity	✗	✓	✗	✗	✗	✗	✓	✓

VI: Variational Inference; MAVI: Meta-Amortized Variational Inference; SNVI: Sequential Neural Variational Inference; MCMC: Markov Chain Monte Carlo; IJ: Infinitesimal Jackknife; IS: Importance Sampling; ABI: Amortized Bayesian Inference; SA-ABI: Sensitivity-Aware ABI (ours).

these approaches require an analytically tractable likelihood function and rather expensive refits to evaluate data sensitivity. SA-ABI, in contrast, allows for a *direct* assessment of posteriors under different C_L and C_P contexts while eliminating likelihood tractability restrictions and the computational burden of refits (see Table 2). Our approach allows sensitivity analyses under drastic perturbations, which is not possible via established methods that rely on MCMC and importance sampling (e.g., when the scaling factor γ approaches zero; Kallioinen et al., 2021). In **Experiment 3**, we demonstrate how our method enables prior sensitivity analyses up to a scaling factor of $\gamma = 0.1$.

Approximator sensitivity Recent work has employed deep ensembles for posterior approximation (Balabanov et al., 2023; Tiulpin & Blaschko, 2022) or, within the scope of simulation-based inference, for improving estimation performance (Modi et al., 2023; Cannon et al., 2022), but not for quantifying the sensitivity induced by the approximation procedure. Schmitt et al. (2023) developed a method to detect simulation gaps in amortized Bayesian parameter estimation via out-of-distribution detection. We adopt a similar perspective on simulation gaps but focus on quantifying the resulting sensitivity in both parameter estimation and model comparison based on the variability of ensemble members. Beyond the identification of simulation gaps, SA-ABI has the crucial advantage of directly assessing the *real-life impact* of a simulation gap in terms of unreliable approximation.

Data sensitivity The fact that analyses are sensitive to the input sample (i.e., data sensitivity) has wide-ranging implications across the sciences. First, the immoderate influence of single data points is closely related to traditional notions of robustness and simulation-based solutions thereof (Huang et al., 2023; Ward et al., 2022). Framed as a hostile scenario, adversarial attacks intend to exploit data sensitivity (Goodfellow et al., 2015; Biggio et al., 2012; Baruch et al., 2019), and adversarial robustness tries to prevent this (see Glöckler et al., 2023, for an ABI application). Second, the sensitivity to different preprocessing choices is directly linked to the reproducibility crisis in the empirical sciences (OSC, 2015; Wicherts et al., 2016). To render this sensitivity tangible, Steegen et al. (2016) introduced the *multiverse analysis*, which repeats an analysis across all alternatively processed data sets. The concept of a holistic analysis across plausible data configurations has been continually extended but is typically restricted by the computational feasibility of large-scale refits (Hall et al., 2022; Liu et al., 2020) or to specific estimation procedures (Broderick et al., 2023). In summary, our sensitivity-aware method unlocks (i) near-instant analyses of data sensitivity and adversarial susceptibility; and (ii) rapid multiverse analyses across a wide space of data processing decisions.

5 Experiments

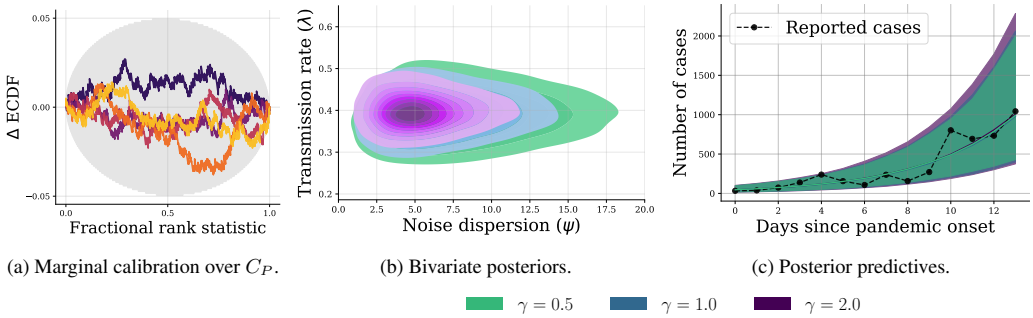
In the following, we demonstrate the utility of our SA-ABI approach on applied, real-data modeling problems of COVID-19 outbreak dynamics (**Experiment 1**; prior sensitivity), climate modeling (**Experiment 2**; prior and likelihood sensitivity), and human decision-making (**Experiment 3**; prior, approximator, and data sensitivity). In each experiment, we first ensure the trustworthiness of SA-ABI by benchmarking it against standard ABI as the state-of-the-art approach for amortized inference on simulation-based models. Afterward, we use our validated approach to obtain insights into sensitivity-induced uncertainties that would have been hardly feasible with existing methods.

All implementations use the BayesFlow library for amortized Bayesian workflows (Radev et al., 2023b). Details for all experiments, such as model setup, network architecture and training, and additional results are available in the **Supplementary Material**.

Table 3: **Experiment 1**: Benchmarking approximation quality and time between standard ABI and SA-ABI (ours).

Simulation budget	Method	MAE ↓ (± SD)	ECE ↓ (± SD)	PC ↑ (± SD)	Time by # of priors ↓ 1	3	1 000
$2^{12} = 4\,096$	ABI	$\mathbf{5.63} \pm 0.07$	0.009 ± 0.0001	0.27 ± 0.05	**2min**	6min	1 876min
	SA-ABI	5.69 ± 0.06	$\mathbf{0.005} \pm 0.001$	$\mathbf{0.28} \pm 0.01$	**2min**	**2min**	**22min**
$2^{14} = 16\,384$	ABI	$\mathbf{5.42} \pm 0.04$	0.008 ± 0.001	$\mathbf{0.38} \pm 0.006$	**6min**	17min	5 557min
	SA-ABI	5.53 ± 0.05	$\mathbf{0.006} \pm 0.002$	0.35 ± 0.005	**6min**	**6min**	**26min**
$2^{16} = 65\,536$	ABI	$\mathbf{5.37} \pm 0.005$	0.01 ± 0.001	$\mathbf{0.40} \pm 0.007$	**21**min	62min	20 721min
	SA-ABI	5.44 ± 0.006	$\mathbf{0.009} \pm 0.001$	0.39 ± 0.01	**21min**	**21min**	**41min**

Note. SD = Standard Deviation. MAE = Mean Absolute Error. ECE = Expected Calibration Error. PC = Posterior Contraction. Metrics are evaluated on the prior scaling setting $\boldsymbol{\gamma} = 1.0$ with $N = 1\,000$ held-out data sets and averaged over ensembles of size $M = 2$ for each method. Thus, SDs reflect the within-ensemble variability. Total times for training and inference for $M = 1$ are reported (extrapolated for 1 000 prior sensitivity evaluations).



(a) Marginal calibration over C_P. (b) Bivariate posteriors. (c) Posterior predictives.

Figure 2: **Experiment 1.** (a) All parameters obtained by SA-ABI are well-calibrated over the full context space C_P. (b) The bivariate posterior of best recoverable parameters λ and ψ indicates substantial sensitivity in terms of uncertainty reduction for ψ, but (c) the posterior predictive distribution appears robust (overlapping median prediction lines and 90% CIs).

5.1 Experiment 1: COVID-19 Outbreak Dynamics

We set the stage with a straightforward example: Modeling the very early stage of a disease outbreak via a SIR model (Dehning et al., 2020). We demonstrate that obtaining amortized prior sensitivity insights does not compromise the approximation performance compared to standard ABI with the same fixed simulation budget. We adopt the simulation model from Radev et al. (2021b) and use a comparable neural network architecture specialized for time-series data. During training, we use power-scaling, that is, element-wise exponentiation $p(\boldsymbol{\theta})^{\boldsymbol{\gamma}}$, to amortize over different priors. We sample the scaling factors in log space, $\boldsymbol{\gamma} \sim \exp(\mathcal{U}(\log(0.5), \log(2.0)))$, to ensure equal amounts of widening and shrinking. Thus, the prior context C_P comprises a vector of scaling powers $\boldsymbol{\gamma}$ for each of the model parameters.

We first benchmark our SA-ABI approach against individual ABI instances trained solely on the tested baseline setting $\gamma = 1.0$. This allows us to determine the performance trade-off for the amortization scope expansion over C_P. Table 3 shows mostly comparable performance of our SA-ABI approach and individual ABI instances with only little trade-offs across all simulation budget settings, *despite SA-ABI spending only a fraction of the simulation budget on the tested settings*. The small variability within the deep ensembles further indicates approximator robustness in the simulated setting. Lastly, Table 3 highlights the time advantage of our method even for sensitivity analyses that only consider C_P.

Figure 2 shows the prior sensitivity results of our SA-ABI approach for the medium simulation budget of $N = 2^{14} = 16\,384$ simulations: Complementing the low calibration error in the benchmark setting without prior scaling, we also

Table 4: **Experiment 2**: Benchmarking approximation quality and time between standard ABI and SA-ABI (ours) in a limited data setting.

Method	MAE ↓ ($\pm$ SD)	ECE ↓ ($\pm$ SD)	PC ↑ ($\pm$ SD)	Time ↓
ABI	4.2 ± 1.1	0.08 ± 0.07	0.980 ± 0.004	313min
SA-ABI	$\mathbf{3.8} \pm 1.3$	$\mathbf{0.04} \pm 0.04$	$\mathbf{0.982} \pm 0.015$	**67min**

Note. SD = Standard Deviation. MAE = Mean Absolute Error. ECE = Expected Calibration Error. PC = Posterior Contraction. Metrics are averaged over test data from all emission scenarios $\times$ climate model settings, resulting in 18 combinations with a total of $N = 2\,916$ held-out data sets. Thus, SDs reflect the variability of 18 individual results per method and metric. Total times for training and inference are reported. All networks use the uninformative prior context.

observe excellent calibration over the full prior space C_P in Figure 2a. The bivariate posteriors for the two parameters with the best recovery (i.e., transmission rate and noise dispersion) unveil that prior sensitivity only affects the noise dispersion (see Figure 2b). Despite prior sensitivity in terms of parameter recoverability, model-based predictive performance is robust to prior scaling (see Figure 2c).

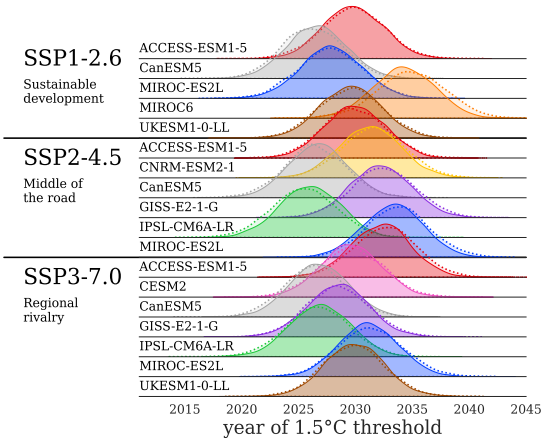
5.2 Experiment 2: Climate Trajectory Forecasting

In this experiment, we study whether model-based global warming forecasts are sensitive to the underlying climate model, emission scenario, and prior specification. Climate models estimate the solutions of differential equations for the fluid dynamics and thermodynamics of atmosphere, ocean, ice, and land masses. Since single forecasts can heavily depend on initial conditions, assumed emission scenarios, and the chosen climate model, modern global warming estimates build on a multitude of simulated trajectories (Riahi et al., 2017; Zelinka et al., 2020; Joshi et al., 2011).

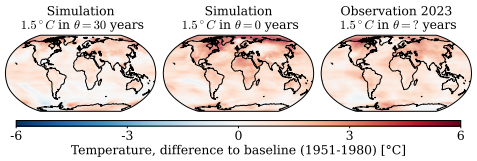
However, trajectories simulated from climate models typically start in pre-industrial times, are not explicitly conditioned on any information since 1850, and are only available in a limited number. In their pioneering work, Diffenbaugh & Barnes (2023) combine neural networks trained on simulated trajectories with recent observational data to predict global warming trends and forecast when critical thresholds are reached. Here, we demonstrate the utility of our approach for *efficiently assessing the sensitivity of model-based predictions* in terms of qualitatively different assumptions regarding the underlying models, emission scenarios, and prior distributions.

Given a high-dimensional spatial observation dataset of surface temperatures (see Figure 3b, right), we are interested in temperature development predictions of different climate models under different future emission scenarios, specifically the time until a global mean surface temperature threshold is exceeded. Framed as a Bayesian parameter estimation task, we model the time-to-threshold θ and explicitly incorporate the climate model and an emission scenario (i.e., SSP1-3) in a likelihood context C_L. Furthermore, we encode a weakly informative prior $\theta \sim \mathcal{U}(-40, 41)$ that encompasses the full range of values present in the training data vs. an informative Gaussian prior $\mathcal{N}_+(10, 10)$, truncated to positive values based on the IPCC sixth assessment report (Lee et al., 2021), in a prior context C_P (see Figure 3c. During training, the neural approximator learns to infer θ from simulated observations with the corresponding context. For each training example, we extract the ground truth θ from later stages of the simulated trajectory (see Figure 3b). At inference time, the network processes an unseen real observation $\boldsymbol{x}_{\text{obs}}$ from the year 2023 with a context that specifies an emission scenario, a climate model, and a prior configuration. The output is the contextualized approximate posterior $p(\theta \mid \boldsymbol{x}_{\text{obs}}, C)$.

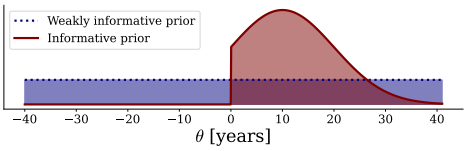
We reproduce the results of Diffenbaugh & Barnes (2023) without sacrificing predictive accuracy (see **Supplementary Material**) and reveal sensitivity to the climate model. Further, Table 4 highlights the advantages of our joint training method utilizing information from *all* context configurations via weight sharing, which is especially relevant in the present limited data setting.



(a) Sensitivity-aware global warming forecasts.



(b) Simulation snapshots (left, center) and the empirical observation (right).



(c) Prior distributions in C_P.

Figure 3: **Experiment 2**. (a) Global warming forecasts are sensitive to the assumed climate model (rows) but not the emission scenario (SSP; groups of rows) or the prior (dotted: weakly informative prior; solid: informative prior). (b) Two examples of simulated observations from the climate model ACCESS-ESM1-5 (SSP-3) with known time-to-threshold (training data) and the current empirical observation that we use for the forecasts. (c) Prior predictive distributions of the weakly informative prior and the informative prior constituting the prior context C_P.

Finally, we study the sensitivity of climate forecasts for a real-world question of societal impact: *When will we reach the* $1.5°C$ *global warming threshold?* As depicted in Figure 3a, the answer to this question is sensitive to the underlying climate model but robust to the assumed emission scenario and informativeness of the prior. This finding is in accordance with Hawkins & Sutton (2009) who argue that the delayed effects of emission scenarios are unlikely to show on a short time scale such as the 1.5°C average warming threshold.

5.3 Experiment 3: Hierarchical Models of Decision-Making

This experiment extends our method to Bayesian model comparison of complex hierarchical models with analytically intractable likelihoods. The *drift-diffusion model* (DDM) is a popular stochastic model of decision-making widely used in cognitive science and neuroscience (Ratcliff et al., 2016). Elsemüller et al. (2023) recently compared four hierarchical models with ABI to test two proposed improvements of the DDM (referred to as $\mathcal{M}_1$): First, allowing model parameters to vary between experimental trials ($\mathcal{M}_3$ and $\mathcal{M}_4$) and second, allowing for evidence accumulation "jumps" via an additional parameter α ($\mathcal{M}_2$ and $\mathcal{M}_4$), which renders the likelihood function intractable.

Elsemüller et al. (2023) found clear evidence for inter-trial variabilities but unclear results concerning the utility of α. In this experiment, we examine the sensitivity of these results to (i) the prior (via inference under 81 power-scaling perturbations of the hierarchical prior on α), (ii) the approximator (via an ensemble of 20 equally configured neural networks), and (iii) the data (via 100 bootstrap samples). Thus, our comprehensive sensitivity analysis is based on 162 000 posterior model probabilities that are challenging to recover even *once* using existing methods. We now use the flexibility of our simulation-based approach to investigate the effects of shrinking and widening the hierarchical prior on α up to a factor of 10 and thus sample the C_P scaling factors during training from $\boldsymbol{\gamma} \sim \exp(\mathcal{U}(\log(0.1), \log(10.0)))$. The complexity of amortizing over the prior space is balanced by two aspects: While the scaling only affects the hyperpriors of the additional α parameter in $\mathcal{M}_2$ and $\mathcal{M}_4$, scaling up to a factor of 10 leads to much more extreme variations than the usual perturbations in likelihood-based settings of up to a factor of 2 (Kallioinen et al., 2021).

As before, we benchmark our SA-ABI approach against individual ABI instances trained solely on the tested baseline setting $\gamma = 1.0$, that is, without varying C_P during training. Despite amortizing over a wide C_P range, we observe little trade-offs in Table 5, with all ensemble members of both ABI and SA-ABI exhibiting near-perfect performance on

Table 5: **Experiment 3**: Benchmarking approximation quality and time between standard ABI and SA-ABI (ours) in a model comparison setting.

Method	MAE ↓ (± SD)	ECE ↓ (± SD)	Accuracy ↑ (± SD)	Time by # of priors ↓ 1	1 000
ABI	$\mathbf{0.012} \pm 0.01$	$\mathbf{0.005} \pm 0.002$	$\mathbf{0.99} \pm 0.01$	**66min**	66 349min
SA-ABI	0.017 ± 0.01	0.01 ± 0.002	0.985 ± 0.01	**66min**	**415min**

Note. SD = Standard Deviation. MAE = Mean Absolute Error. ECE = Expected Calibration Error. Metrics are evaluated on the prior scaling setting $\gamma = 1.0$ with $N = 8\,000$ held-out data sets (2 000 per model) and averaged over ensembles of size $M = 20$ for each method. Thus, SDs reflect the within-ensemble variability. Total times for training and inference for $M = 1$ are reported (extrapolated for 1 000 prior sensitivity evaluations).

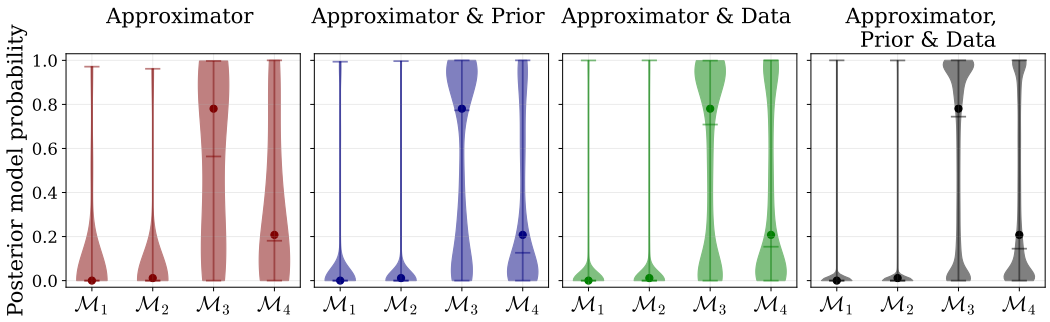


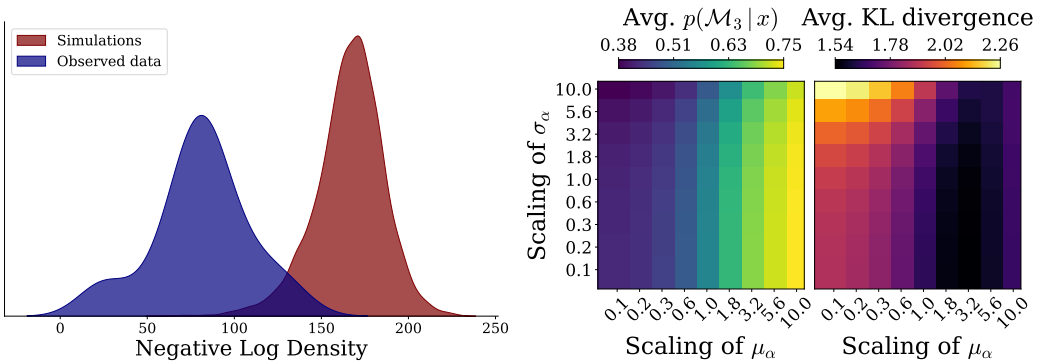
Figure 4: **Experiment 3**. Our sensitivity-aware posterior model probabilities indicate substantial approximator sensitivity but robustness to additional prior scaling and data perturbations. Dots represent the original results by Elsemüller et al. (2023).

simulated data. Strikingly, Figure 4 reveals highly inconsistent predictions of the ensemble members on the empirical data (also for ABI, see **Supplementary Material**). [2] This stark discrepancy implies the presence of a simulation gap. Indeed, contrasting the empirical data with the typical set of model simulations (Nalisnick et al., 2019; Morningstar et al., 2021) in Figure 5a flags the empirical data as out-of-distribution for the deep ensemble. Further, Figure 4 shows a comparatively low sensitivity against perturbing the hierarchical prior on α or the empirical data, with the medians under all perturbations close to the results by Elsemüller et al. (2023). Viewing deep ensemble predictions as approximate Bayesian model averaging (Wilson & Izmailov, 2020), we can conclude that the original results hold, but with substantial OOD uncertainty due to the simulation gap. A closer inspection of prior sensitivity in Figure 5b reveals (i) quantitative sensitivity to wide specifications of the hierarchical location μ_α, which increases under narrow specifications of the hierarchical scale σ_α, and (ii) qualitative sensitivity to settings of μ_α concerning the model with the highest probability, $\mathcal{M}_3$.

6 Conclusion

We proposed SA-ABI, an approach for large-scale sensitivity analyses with a keen emphasis on managing uncertainty in critical, high-impact scenarios. By leveraging amortized inference, our method causes minimal computational overhead during inference and can be directly integrated into software toolkits for amortized Bayesian workflows (such as Radev et al., 2023b; Tejero-Cantero et al., 2020). Future work should investigate more efficient approaches to quantify approximator sensitivity, with specific attention to Bayesian neural networks (Izmailov et al., 2021). Ad-

[2] Recall that the approximation targets in Bayesian model comparison are (categorical) posterior model probabilities, not posterior distributions over parameters. Thus, the variability shown in Figure 4 directly reflects the sensitivity caused by perturbing the respective model component(s).



(a) Approximator sensitivity

(b) Prior sensitivity.

Figure 5: **Experiment 3**. (a) The learned summary statistics for the observed data are out-of-distribution (OOD) relative to the typical set of summary statistics for the model simulations (both distributions marginalized over C_P and C_A). (b) The posteriors are quantitatively sensitive to power-scaling of the prior location μ_α, as indexed by the ensemble-averaged probability for $\mathcal{M}_3$ (left) as well as the KL divergence between the original results by Elsemüller et al. (2023) vs. scaled model posteriors (right). Notable qualitative sensitivity is present mainly due to different μ_α values.

ditionally, whereas arbitrary data and approximator configurations can be explored at any time, likelihood and prior configurations have to be integrated into the training process. We believe that transfer learning (Bengio et al., 2009; Zhuang et al., 2021) is a promising tool to resolve this constraint, unlocking further flexibility on all sensitivity facets. We extend neural Bayesian inference (parameter estimation and model comparison) by amortizing over families of probabilistic models, as characterized by context variables C. This drastically expands the amortization scope of the employed neural approximators and constitutes a major leap towards foundation models for probabilistic (Bayesian) inference. Follow-up research in this direction might further increase the probabilistic model space during the training stage to facilitate near-universal amortized inference with pre-trained neural networks.

Acknowledgments

We thank the anonymous reviewers and the action editor for improving the manuscript with their constructive and thoughtful feedback. Additionally, we thank Noah S. Diffenbaugh and Elizabeth A. Barnes for helping us build upon their pioneering work in Experiment 2. LE was supported by a grant from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation; GRK 2277) to the research training group Statistical Modeling in Psychology (SMiP) and the Google Cloud Research Credits program with the award GCP19980904. HO was supported by the state of Baden-Württemberg through bwHPC. MS was supported by the Cyber Valley Research Fund (grant number: CyVy-RF-2021-16) and the DFG under Germany's Excellence Strategy EXC-2075 - 390740016 (the Stuttgart Cluster of Excellence SimTech). UK was supported by the Informatics for Life initiative funded by the Klaus Tschira Foundation and the DFG under Germany's Excellence Strategy EXC-2181 - 390900948 (the Heidelberg Cluster of Excellence STRUCTURES).

References

Syed Mumtaz Ali and Samuel D Silvey. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society: Series B (Methodological)*, 28(1):131–142, 1966.

Grace Avecilla, Julie N Chuong, Fangfei Li, Gavin Sherlock, David Gresham, and Yoav Ram. Neural networks enable efficient and accurate simulation-based inference of evolutionary parameters from adaptation dynamics. *PLoS Biology*, 20(5):e3001633, 2022.

Oleksandr Balabanov, Bernhard Mehlig, and Hampus Linander. Bayesian posterior approximation with stochastic ensembles. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2023. doi: 10.1109/cvpr52729.2023.01317. URL https://doi.org/10.1109/cvpr52729.2023.01317.

Gilad Baruch, Moran Baruch, and Yoav Goldberg. A little is enough: Circumventing defenses for distributed learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/ec1c59141046cd1866bbbcdfb6ae31d4-Paper.pdf.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

Michael Betancourt. Calibrating model-based inferences and decisions. *arXiv preprint arXiv:1803.08393*, 2018.

Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM review*, 59(1):65–98, 2017.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pp. 1467–1474, Madison, WI, USA, 2012. Omnipress.

Sebastian Bischoff, Alana Darcher, Michael Deistler, Richard Gao, Franziska Gerken, Manuel Gloeckler, Lisa Haxel, Jaivardhan Kapoor, Janne K Lappalainen, Jakob H Macke, et al. A practical guide to statistical distances for evaluating generative models in science. *arXiv preprint arXiv:2403.12636*, 2024.

Benjamin Bloem-Reddy and Yee Whye Teh. Probabilistic symmetries and invariant neural networks. *J. Mach. Learn. Res.*, 21:90–1, 2020.

Seth Blumberg, Sebastian Funk, and Juliet RC Pulliam. Detecting differential transmissibilities that affect the size of self-limited outbreaks. *PLoS pathogens*, 10(10):e1004452, 2014.

Udo Boehm, Jeffrey Annis, Michael J. Frank, Guy E. Hawkins, Andrew Heathcote, David Kellen, Angelos-Miltiadis Krypotos, Veronika Lerche, Gordon D. Logan, Thomas J. Palmeri, Don van Ravenzwaaij, Mathieu Servant, Henrik Singmann, Jeffrey J. Starns, Andreas Voss, Thomas V. Wiecki, Dora Matzke, and Eric-Jan Wagenmakers. Estimating across-trial variability parameters of the diffusion decision model: Expert advice and recommendations. *Journal of Mathematical Psychology*, 87:46–75, 2018.

Alexander Braumann, Jonas Krampe, Jens-Peter Kreiss, and Efstathios Paparoditis. Estimation of the distribution of the individual reproduction number: The case of the covid-19 pandemic. *arXiv preprint arXiv:2101.07919*, 2021.

Tamara Broderick, Ryan Giordano, and Rachael Meager. An automatic finite-sample robustness metric: when can dropping a little data make a big difference? *arXiv preprint arXiv:2011.14999*, 2023.

Paul-Christian Bürkner, Maximilian Scholz, and Stefan T Radev. Some models are useful, but how do we know which ones? towards a unified bayesian model taxonomy. *arXiv preprint arXiv:2209.02439*, 2022.

Patrick Cannon, Daniel Ward, and Sebastian M Schmon. Investigating the impact of model misspecification in neural simulation-based inference. *arXiv preprint arXiv:2209.01845*, 2022.

Jeffrey Chan, Valerio Perrone, Jeffrey Spence, Paul Jenkins, Sara Mathieson, and Yun Song. A likelihood-free inference framework for population genetic data using exchangeable neural networks. *Advances in neural information processing systems*, 31, 2018.

Yanzhi Chen, Dinghuai Zhang, Michael Gutmann, Aaron Courville, and Zhanxing Zhu. Neural approximate sufficient statistics for implicit models. *arXiv preprint arXiv:2010.10079*, 2020.

Yanzhi Chen, Michael U Gutmann, and Adrian Weller. Is learning summary statistics necessary for likelihood-free inference? In *International Conference on Machine Learning*, pp. 4529–4544. PMLR, 2023.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

Imre Csiszár. Eine informationstheoretische ungleichung und ihre anwendung auf beweis der ergodizitaet von markoffschen ketten. *Magyer Tud. Akad. Mat. Kutato Int. Koezl.*, 8:85–108, 1964.

Maximilian Dax, Stephen R Green, Jonathan Gair, Jakob H Macke, Alessandra Buonanno, and Bernhard Schölkopf. Real-time gravitational wave science with neural posterior estimation. *Physical review letters*, 127(24):241103, 2021.

Jonas Dehning, Johannes Zierenberg, F Paul Spitzner, Michael Wibral, Joao Pinheiro Neto, Michael Wilczek, and Viola Priesemann. Inferring change points in the spread of covid-19 reveals the effectiveness of interventions. *Science*, 369(6500):eabb9789, 2020.

Charita Dellaporta, Jeremias Knoblauch, Theodoros Damoulas, and François-Xavier Briol. Robust bayesian inference for simulator-based models via the mmd posterior bootstrap. In *International Conference on Artificial Intelligence and Statistics*, pp. 943–970. PMLR, 2022.

Sarah Depaoli, Sonja D. Winter, and Marieke Visser. The importance of prior sensitivity analysis in bayesian statistics: Demonstrations using an interactive shiny app. *Frontiers in Psychology*, 11, 2020. doi: 10.3389/fpsyg.2020.608045. URL https://doi.org/10.3389%2Ffpsyg.2020.608045.

Noah S. Diffenbaugh and Elizabeth A. Barnes. Data-driven predictions of the time remaining until critical global warming thresholds are reached. *Proceedings of the National Academy of Sciences*, 120(6):e2207183120, February 2023. doi: 10.1073/pnas.2207183120. URL https://www.pnas.org/doi/10.1073/pnas.2207183120.

Peter J Diggle and Richard J Gratton. Monte carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 46(2):193–212, 1984.

Lasse Elsemüller, Martin Schnuerch, Paul-Christian Bürkner, and Stefan T Radev. A deep learning method for comparing bayesian hierarchical models. *arXiv preprint arXiv:2301.11873*, 2023.

Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. *arXiv preprint arXiv:1912.02757*, 2019.

Andrew Gelman, Jessica Hwang, and Aki Vehtari. Understanding predictive information criteria for bayesian models. *Statistics and computing*, 24(6):997–1016, 2014.

Ryan Giordano, Tamara Broderick, and Michael I. Jordan. Covariances, robustness and variational bayes. *Journal of Machine Learning Research*, 19(1):1981–2029, 2018.

Ryan Giordano, William Stephenson, Runjing Liu, Michael Jordan, and Tamara Broderick. A swiss army infinitesimal jackknife. In Kamalika Chaudhuri and Masashi Sugiyama (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 1139–1147. PMLR, 2019.

Manuel Glöckler, Michael Deistler, and Jakob H Macke. Variational methods for simulation-based inference. *arXiv preprint arXiv:2203.04176*, 2022.

Manuel Glöckler, Michael Deistler, and Jakob H Macke. Adversarial robustness of amortized bayesian inference. *arXiv preprint arXiv:2305.14984*, 2023.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

Pedro J Gonçalves, Jan-Matthis Lueckmann, Michael Deistler, Marcel Nonnenmacher, Kaan Öcal, Giacomo Bassetto, Chaitanya Chintaluri, William F Podlaski, Sara A Haddad, Tim P Vogels, et al. Training deep neural density estimators to identify mechanistic models of neural dynamics. *Elife*, 9:e56261, 2020.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. URL http://arxiv.org/abs/1412.6572.

David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic posterior transformation for likelihood-free inference. In *International Conference on Machine Learning*, pp. 2404–2414. PMLR, 2019.

A Gretton, K. Borgwardt, Malte Rasch, Bernhard Schölkopf, and AJ Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13:723–773, 03 2012.

Brian D. Hall, Yang Liu, Yvonne Jansen, Pierre Dragicevic, Fanny Chevalier, and Matthew Kay. A survey of tasks and visualizations in multiverse analysis reports. *Computer Graphics Forum*, 41(1):402–426, 2022. doi: 10.1111/cgf.14443. URL https://doi.org/10.1111/cgf.14443.

Ed Hawkins and Rowan Sutton. The Potential to Narrow Uncertainty in Regional Climate Predictions. *Bulletin of the American Meteorological Society*, 90(8):1095–1108, August 2009. ISSN 0003-0007, 1520-0477. doi: 10.1175/2009BAMS2607.1. URL https://journals.ametsoc.org/view/journals/bams/90/8/2009bams2607_1.xml.

Daolang Huang, Ayush Bharti, Amauri Souza, Luigi Acerbi, and Samuel Kaski. Learning robust statistics for simulation-based inference under model misspecification. *arXiv preprint arXiv:2305.15871*, 2023.

David Rios Insua and Fabrizio Ruggeri. *Robust Bayesian Analysis*. Springer Science & Business Media, Berlin Heidelberg, 2012. ISBN 978-1-461-21306-2.

Pavel Izmailov, Sharad Vikram, Matthew D Hoffman, and Andrew Gordon Gordon Wilson. What are bayesian neural network posteriors really like? In *International conference on machine learning*, pp. 4629–4640. PMLR, 2021.

Xu Ji, Razvan Pascanu, R Devon Hjelm, Balaji Lakshminarayanan, and Andrea Vedaldi. Test sample accuracy scales with training sample density in neural networks. In *Conference on Lifelong Learning Agents*, pp. 629–646. PMLR, 2022.

Manoj Joshi, Ed Hawkins, Rowan Sutton, Jason Lowe, and David Frame. Projections of when temperature change will exceed 2 °C above pre-industrial levels. *Nature Climate Change*, 1(8):407–412, November 2011. ISSN 1758-678X, 1758-6798. doi: 10.1038/nclimate1261. URL https://www.nature.com/articles/nclimate1261.

Noa Kallioinen, Topi Paananen, Paul-Christian Bürkner, and Aki Vehtari. Detecting and diagnosing prior and likelihood sensitivity with power-scaling. *arXiv preprint arXiv:2107.14054*, 2021.

Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations*, pp. 1–15, 2015.

Daniel Kristanto, Micha Burkhardt, Christiane Thiel, Stefan Debener, Carsten Giessing, and Andrea Hildebrandt. The multiverse of data preprocessing and analysis in graph-based fmri: A systematic literature review of analytical choices fed into a decision support tool for informed analysis. *bioRxiv*, pp. 2024–01, 2024.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

June-Yi Lee, Jochem Marotzke, Govindasamy Bala, Long Cao, Susanna Corti, John P Dunne, Francois Engelbrecht, Erich Fischer, John C Fyfe, Christopher Jones, et al. Future global climate: scenario-based projections and near-term information. In *Climate change 2021: The physical science basis. Contribution of working group I to the sixth assessment report of the intergovernmental panel on climate change*, pp. 553–672. Cambridge University Press, 2021.

Veronika Lerche and Andreas Voss. Model complexity in diffusion modeling: Benefits of making the model more parsimonious. *Frontiers in Psychology*, 7(1324):1–14, 2016.

Friedrich Liese and Igor Vajda. On divergences and informations in statistics and information theory. *IEEE Transactions on Information Theory*, 52(10):4394–4412, 2006.

Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

Yang Liu, Tim Althoff, and Jeffrey Heer. Paths explored, paths omitted, paths obscured: Decision points & selective reporting in end-to-end data analysis. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, 2020. doi: 10.1145/3313831.3376533. URL https://doi.org/10.1145/3313831.3376533.

David MacKay. *Information theory, inference and learning algorithms*. Cambridge University Press, 2003.

Jean-Michel Marin, Pierre Pudlo, Christian P Robert, and Robin J Ryder. Approximate bayesian computational methods. *Statistics and computing*, 22(6):1167–1180, 2012.

Chirag Modi, Shivam Pandey, Matthew Ho, ChangHoon Hahn, Bruno Blancard, and Benjamin Wandelt. Sensitivity analysis of simulation-based inference for galaxy clustering. *arXiv preprint arXiv:2309.15071*, 2023.

Warren Morningstar, Cusuh Ham, Andrew Gallagher, Balaji Lakshminarayanan, Alex Alemi, and Joshua Dillon. Density of states estimation for out of distribution detection. In *International Conference on Artificial Intelligence and Statistics*, pp. 3232–3240. PMLR, 2021.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 2901—2907. AAAI Press, 2015.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, and Balaji Lakshminarayanan. Detecting out-of-distribution inputs to deep generative models using typicality. *arXiv preprint arXiv:1906.02994*, 2019.

Radford M Neal. Mcmc using hamiltonian dynamics. In Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng (eds.), *Handbook of markov chain monte carlo*, chapter 5. Chapman and Hall/CRC, 2011.

OSC. Estimating the reproducibility of psychological science. *Science*, 349(6251), 2015. doi: 10.1126/science.aac4716. URL https://doi.org/10.1126/science.aac4716. Published by the Open Science Collaboration (OSC).

Lorenzo Pacchiardi and Ritabrata Dutta. Generalized bayesian likelihood-free inference using scoring rules estimators. *arXiv preprint arXiv:2104.03889*, 2021.

Pierre Pudlo, Jean-Michel Marin, Arnaud Estoup, Jean-Marie Cornuet, Mathieu Gautier, and Christian P Robert. Reliable abc model choice via random forests. *Bioinformatics*, 32(6):859–866, 2016.

Stefan Radev. *Deep Learning Architectures for Amortized Bayesian Inference in Cognitive Modeling*. PhD thesis, Heidelberg University, 2021.

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*, 2020.

Stefan T Radev, Marco D'Alessandro, Ulf K Mertens, Andreas Voss, Ullrich Köthe, and Paul-Christian Bürkner. Amortized bayesian model comparison with evidential deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2021a.

Stefan T Radev, Frederik Graw, Simiao Chen, Nico T Mutters, Vanessa M Eichel, Till Bärnighausen, and Ullrich Köthe. OutbreakFlow: Model-based Bayesian inference of disease outbreak dynamics with invertible neural networks and its application to the COVID-19 pandemics in germany. *PLoS computational biology*, 2021b.

Stefan T. Radev, Marvin Schmitt, Valentin Pratz, Umberto Picchini, Ullrich Köthe, and Paul-Christian Bürkner. JANA: Jointly Amortized Neural Approximation of Complex Bayesian Models. In Robin J. Evans and Ilya Shpitser (eds.), *Proceedings of the 39th Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pp. 1695–1706. PMLR, 2023a.

Stefan T Radev, Marvin Schmitt, Lukas Schumacher, Lasse Elsemüller, Valentin Pratz, Yannik Schälte, Ullrich Köthe, and Paul-Christian Bürkner. Bayesflow: Amortized bayesian workflows with neural networks. *arXiv preprint arXiv:2306.16015*, 2023b.

Roger Ratcliff. A theory of memory retrieval. *Psychological review*, 85(2):59, 1978.

Roger Ratcliff, Philip L Smith, Scott D Brown, and Gail McKoon. Diffusion decision model: Current issues and history. *Trends in cognitive sciences*, 20(4):260–281, 2016.

Keywan Riahi, Detlef P. Van Vuuren, Elmar Kriegler, Jae Edmonds, Brian C. O'Neill, Shinichiro Fujimori, Nico Bauer, Katherine Calvin, Rob Dellink, Oliver Fricko, Wolfgang Lutz, Alexander Popp, Jesus Crespo Cuaresma, Samir Kc, Marian Leimbach, Leiwen Jiang, Tom Kram, Shilpa Rao, Johannes Emmerling, Kristie Ebi, Tomoko Hasegawa, Petr Havlik, Florian Humpenöder, Lara Aleluia Da Silva, Steve Smith, Elke Stehfest, Valentina Bosetti, Jiyong Eom, David Gernaat, Toshihiko Masui, Joeri Rogelj, Jessica Strefler, Laurent Drouet, Volker Krey, Gunnar Luderer, Mathijs Harmsen, Kiyoshi Takahashi, Lavinia Baumstark, Jonathan C. Doelman, Mikiko Kainuma, Zbigniew Klimont, Giacomo Marangoni, Hermann Lotze-Campen, Michael Obersteiner, Andrzej Tabeau, and Massimo Tavoni. The Shared Socioeconomic Pathways and their energy, land use, and greenhouse gas emissions implications: An overview. *Global Environmental Change*, 42:153–168, January 2017. ISSN 09593780. doi: 10.1016/j.gloenvcha.2016.05.009. URL https://linkinghub.elsevier.com/retrieve/pii/S0959378016300681.

Malgorzata Roos, Thiago G Martins, Leonhard Held, and Håvard Rue. Sensitivity analysis for bayesian hierarchical models. *Bayesian Analysis*, 10(2):321–349, 2015.

Marvin Schmitt, Paul-Christian Bürkner, and Köthe. Detecting model misspecification in amortized bayesian inference with neural networks. *Proceedings of the German Conference on Pattern Recognition (GCPR)*, 2023.

Cornelius Schröder and Jakob H Macke. Simultaneous identification of models and parameters of scientific simulators. *arXiv preprint arXiv:2305.15174*, 2023.

Adi Shamir, Odelia Melamed, and Oriel BenShmuel. The dimpled manifold model of adversarial examples in machine learning. *arXiv preprint arXiv:2106.10151*, 2021.

Joseph P. Simmons, Leif D. Nelson, and Uri Simonsohn. False-positive psychology. *Psychological Science*, 22(11):1359–1366, 2011. doi: 10.1177/0956797611417632. URL https://doi.org/10.1177/0956797611417632.

Sara Steegen, Francis Tuerlinckx, Andrew Gelman, and Wolf Vanpaemel. Increasing transparency through a multiverse analysis. *Perspectives on Psychological Science*, 11(5):702–712, 2016. doi: 10.1177/1745691616658637. URL https://doi.org/10.1177/1745691616658637.

Sean Talts, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. Validating bayesian inference algorithms with simulation-based calibration. *arXiv preprint arXiv:1804.06788*, 2018.

Alvaro Tejero-Cantero, Jan Boelts, Michael Deistler, Jan-Matthis Lueckmann, Conor Durkan, Pedro J Gonçalves, David S Greenberg, and Jakob H Macke. Sbi–a toolkit for simulation-based inference. *arXiv preprint arXiv:2007.09114*, 2020.

Aleksei Tiulpin and Matthew B. Blaschko. Greedy bayesian posterior approximation with deep ensembles. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=P1DuPJzVTN.

Aki Vehtari, Andrew Gelman, and Jonah Gabry. Practical bayesian model evaluation using leave-one-out cross-validation and waic. *Statistics and computing*, 27:1413–1432, 2017.

Andreas Voss, Veronika Lerche, Ulf Mertens, and Jochen Voss. Sequential sampling models with variable boundaries and non-normal noise: A comparison of six models. *Psychonomic bulletin & review*, 26(3):813–832, 2019.

Eric-Jan Wagenmakers, Alexandra Sarafoglou, and Balazs Aczel. One statistical analysis must not rule them all. *Nature*, 605(7910):423–425, 2022.

Daniel Ward, Patrick Cannon, Mark Beaumont, Matteo Fasiolo, and Sebastian Schmon. Robust neural posterior estimation and statistical model criticism. *Advances in Neural Information Processing Systems*, 35:33845–33859, 2022.

Florian Wenzel, Jasper Snoek, Dustin Tran, and Rodolphe Jenatton. Hyperparameter ensembles for robustness and uncertainty quantification. *Advances in Neural Information Processing Systems*, 33:6514–6527, 2020.

Jelte M. Wicherts, Coosje L. S. Veldkamp, Hilde E. M. Augusteijn, Marjan Bakker, Robbie C. M. van Aert, and Marcel A. L. M. van Assen. Degrees of freedom in planning, running, analyzing, and reporting psychological studies: A checklist to avoid p-hacking. *Frontiers in Psychology*, 7, 2016. doi: 10.3389/fpsyg.2016.01832. URL https://doi.org/10.3389/fpsyg.2016.01832.

Eva Marie Wieschen, Andreas Voss, and Stefan Radev. Jumping to conclusion? a lévy flight model of decision making. *The Quantitative Methods for Psychology*, 16(2):120–132, 2020.

Andrew G Wilson and Pavel Izmailov. Bayesian deep learning and a probabilistic perspective of generalization. *Advances in neural information processing systems*, 33:4697–4708, 2020.

Samuel Wiqvist, Pierre-Alexandre Mattei, Umberto Picchini, and Jes Frellsen. Partially exchangeable networks and architectures for learning summary statistics in approximate bayesian computation. In *International Conference on Machine Learning*, pp. 6798–6807. PMLR, 2019.

Samuel Wiqvist, Jes Frellsen, and Umberto Picchini. Sequential neural posterior and likelihood approximation. *arXiv preprint arXiv:2102.06522*, 2021.

Mike Wu, Kristy Choi, Noah Goodman, and Stefano Ermon. Meta-amortized variational inference and learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):6404–6412, 2020. doi: 10.1609/aaai.v34i04.6111. URL https://doi.org/10.1609/aaai.v34i04.6111.

Jingkang Yang, Pengyun Wang, Dejian Zou, Zitang Zhou, Kunyuan Ding, Wenxuan Peng, Haoqi Wang, Guangyao Chen, Bo Li, Yiyou Sun, et al. Openood: Benchmarking generalized out-of-distribution detection. *Advances in Neural Information Processing Systems*, 35:32598–32611, 2022.

Yuling Yao, Bruno Régaldo-Saint Blancard, and Justin Domke. Simulation based stacking, 2023.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. *Advances in neural information processing systems*, 30, 2017.

Mark D. Zelinka, Timothy A. Myers, Daniel T. McCoy, Stephen Po-Chedley, Peter M. Caldwell, Paulo Ceppi, Stephen A. Klein, and Karl E. Taylor. Causes of Higher Climate Sensitivity in CMIP6 Models. *Geophysical Research Letters*, 47(1):e2019GL085782, 2020. ISSN 1944-8007. doi: 10.1029/2019GL085782. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2019GL085782.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2021. doi: 10.1109/jproc.2020.3004555. URL https://doi.org/10.1109/jproc.2020.3004555.

Supplementary Material

A Frequently Asked Questions (FAQ)

Q: How can I reproduce the results?

Code for reproducing all results from this paper is freely available at https://github.com/bayesflow-org/SA-ABI.

Q: Can I apply your sensitivity-aware approach to posterior predictive model comparison as well?

Yes! In this work, we focus on prior predictive model comparison, but our ideas directly apply to posterior predictive metrics (Gelman et al., 2014), such as leave-one-out cross-validation (Vehtari et al., 2017). We recommend the joint usage of a posterior and a likelihood network as proposed in Radev et al. (2023a) to achieve amortization in this application.

Q: Are there limits to the distributional shapes that can be explored in C_L and C_P?

The only requirement for distributions in the likelihood and prior context is being able to simulate data from the resulting model. Besides that, SA-ABI gives modelers full flexibility in specifying any theoretically meaningful alternative formulations without concerns about analytically tractable likelihoods, unlike MCMC methods.

Q: How exactly did you encode the context variables C_L and C_P in a suitable format for the neural network in your experiments?

In **Experiment 1** and **Experiment 3**, each prior distribution over a parameter is continuously tempered by power-scaling. Consequently, C_P is encoded by a vector holding the scaling factors for each prior component. In **Experiment 2**, both C_L and C_P consist of discrete choices and are therefore passed to the inference network in one-hot-encoded vectors.

B Methods: Additional Details

B.1 Hypothesis Testing for Quantitative Sensitivity

We can employ a sampling-based (frequentist) hypothesis test to determine the probability of observed $\mathbb{D}(\cdot \,||\, \cdot)$ estimates (hitherto referred to as $\widehat{\mathbb{D}}$) under the null hypothesis of zero difference between $p(\boldsymbol{\theta} \,|\, \boldsymbol{x}, C_i)$ and $p(\boldsymbol{\theta} \,|\, \boldsymbol{x}, C_j)$. For this, we can construct an approximate sampling distribution of $\widehat{\mathbb{D}}$ under the null hypothesis via bootstrap or permutation tests based on multiple draws from $p(\boldsymbol{\theta} \,|\, \boldsymbol{x}, C_i)$. Based on the approximate sampling distribution, we can then obtain a critical $\widehat{\mathbb{D}}$ value for a fixed Type I error probability δ and compare it to the observed one. The power of such a test will generally be high when having access to many draws from $p(\boldsymbol{\theta} \,|\, \boldsymbol{x}, C_i)$ and $p(\boldsymbol{\theta} \,|\, \boldsymbol{x}, C_j)$, a requirement that is easily met in the context of ABI.

C Experiments: Implementation Details and Additional Results

C.1 Benchmarking Metrics

For the benchmarks conducted in **Experiment 1** and **Experiment 2**, we measure three complementary performance metrics on J unseen test data sets $\{\mathcal{D}_o^{(j)}\}_{j=1}^J$ with known ground-truth parameters $\{\boldsymbol{\theta}_*^{(j)}\}_{j=1}^J$. For each data set $\mathcal{D}_o^{(j)}$, we obtain a set $\{\boldsymbol{\theta}_s^{(j)}\}_{s=1}^S$ of S posterior draws from the neural approximator $q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid \mathcal{D}_o^{(j)})$. We summarize each metric into a single measure across all test data sets and parameters for a given neural approximator and test setting (e.g., $\gamma = 0.5$ in **Experiment 1**).

We use the *Mean Absolute Error* (MAE) to measure the overall error between posterior draws $\boldsymbol{\theta}_s^{(j)}$ and ground-truth parameters $\boldsymbol{\theta}_*^{(j)}$:

$$\text{MAE} = \frac{1}{J}\sum_{j=1}^{J}\left|\frac{1}{S}\sum_{s=1}^{S}\left(\boldsymbol{\theta}_s^{(j)} - \boldsymbol{\theta}_*^{(j)}\right)\right|. \tag{13}$$

Further, we assess uncertainty calibration via the *Expected Calibration Error* (ECE). In Bayesian parameter estimation, all uncertainty regions $U_q(\boldsymbol{\theta} \mid \mathcal{D})$ of the true posterior $p(\boldsymbol{\theta} \mid \mathcal{D})$ are by definition well-calibrated for any quantile $q \in (0, 1)$ (Bürkner et al., 2022), such that:

$$q = \iint \mathbf{I}[\boldsymbol{\theta}_* \in U_q(\boldsymbol{\theta} \mid \mathcal{D})]\, p(\mathcal{D} \mid \boldsymbol{\theta}_*)\, p(\boldsymbol{\theta}_*) \mathrm{d}\boldsymbol{\theta}_* \mathrm{d}\mathcal{D}, \tag{14}$$

with $\mathbf{I}[\cdot]$ denoting the indicator function. Simulation-based calibration (SBC; Talts et al., 2018) measures miscalibration via deviations from this equality. We estimate the ECE via the median SBC error of 20 linearly spaced credible intervals with quantiles of $q \in [0.5\%, 99.5\%]$. [3] Lastly, we measure Bayesian information gain via the median *Posterior Contraction* (PC) across data sets, defined as $1 - \text{Var}(\text{Posterior})/\text{Var}(\text{Prior})$ (Betancourt, 2018).

C.2 Experiment 1: COVID-19 Outbreak Dynamics

Model Setup: We consider a simple SIR model where individuals are either susceptible, S, infected, I, or recovered, R. Both infection and recovery are modeled with a constant transmission rate λ and recovery rate μ, respectively. The model is described by a system of ordinary differential equations (ODEs),

$$\frac{dS}{dt} = -\lambda\left(\frac{S\,I}{N}\right), \tag{15}$$

$$\frac{dI}{dt} = \lambda\left(\frac{S\,I}{N}\right) - \mu\,I, \tag{16}$$

$$\frac{dR}{dt} = \mu\,I, \tag{17}$$

with $N = S + I + R$ denoting the total population size. In addition to the ODE parameters λ and μ, our model includes a reporting delay parameter D and a noise dispersion parameter ψ, which jointly influence the (noisy) number of reported infected individuals via

$$I_t^{(obs)} \sim \text{NegBinomial}(I_{t-D}^{(new)}, \psi), \tag{18}$$

with $I^{(new)} = \lambda(S_t I_t / N)$. The negative binomial distribution allows for modeling dispersion, i.e., variation of the variability independent of the mean, which is considered likely for early phases of the COVID-19 pandemic (Blumberg et al., 2014; Braumann et al., 2021). For our implementation, we transform the parameterization in Equation 18 with mean $I_{t-D}^{(new)}$ and dispersion ψ to the `numpy` library's implementation of the negative binomial distribution with number of successes n and probability of success p:

$$n = \psi \tag{19}$$

$$\sigma = \frac{I_{t-D}^{(new)} + 1}{\psi (I_{t-D}^{(new)})^2} \tag{20}$$

$$p = \frac{\sigma - I_{t-D}^{(new)}}{\sigma}. \tag{21}$$

The fifth estimated model parameter is the initial number of infected individuals I_0.

We use the same prior specification as Radev et al. (2021b), which is displayed in Table 6 along with the respective power-scaling scheme. Figure 6 shows the behavior of the prior predictive distributions under different power-scaling values $\boldsymbol{\gamma}$.

[3] In **Experiment 3**, we use the ECE formulation by Naeini et al. (2015) for probabilistic classification.

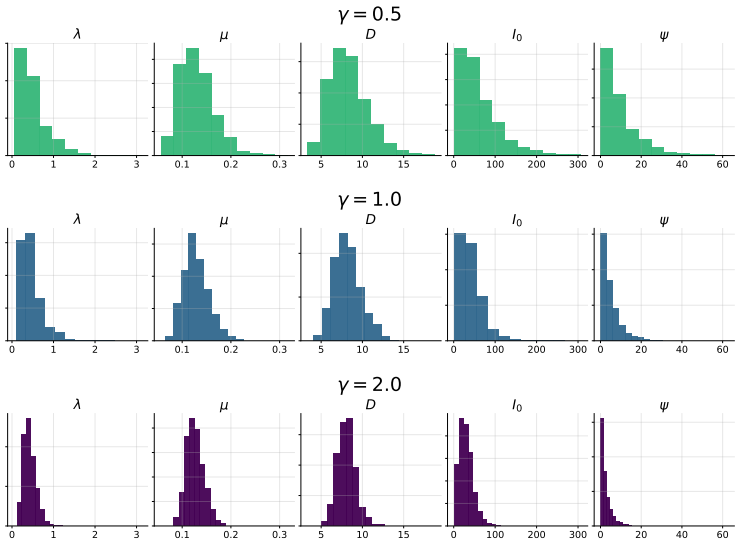


Figure 6: **Experiment 1**. Prior predictive distributions under different scaling parameters $\boldsymbol{\gamma}$.

Table 6: **Experiment 1**. Power-scaled prior distributions for all parameters.

Parameter	Symbol	Power-scaled prior distribution
Transmission rate	λ	LogNormal$(\log(0.4), 0.5/\sqrt{\gamma_1})$
Recovery rate of infected individuals	μ	LogNormal$(\log(1/8), 0.2/\sqrt{\gamma_2})$
Reporting delay (lag)	D	LogNormal$(\log(8), 0.2/\sqrt{\gamma_3})$
Initial number of infected individuals	I_0	Gamma$(2\gamma_4 - \gamma_4 + 1, 20/\gamma_4)$
Noise dispersion	ψ	Exponential$(5/\gamma_5)$

Note. Our parameterization follows the `numpy` library's implementation of the respective distribution.

We use time-series data from the first two weeks of the COVID-19 pandemic in Germany provided by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University, licensed under CC BY 4.0. [4]

Neural Network and Training: Our neural network architecture follows a simplified version of the design implemented by Radev et al. (2021b): We use a recurrent network with gated recurrent units as summary network and a conditional invertible network as inference network.

All computations for this experiment were performed on a single-GPU machine with an NVIDIA RTX 3070 graphics card and an AMD Ryzen 5 5600X processor. Simulating 16 384 training data sets took 8 seconds and subsequent offline training for 75 epochs took 6 minutes.

Additional Results: We further investigate SA-ABI and ABI for potential reductions in approximation performance due to amortized prior sensitivity in the medium simulation budget setting of $2^{14} = 16\,384$. Figure 7 and Figure 8 show similar parameter recovery in the baseline $\gamma = 1.0$ setting, both limited by the small number of $T = 14$ data points available. Figure 9 and Figure 10 demonstrate that calibrated predictions are nevertheless mostly achievable, with equal patterns between SA-ABI (Figure 9) and standard ABI (Figure 10) for the baseline setting. Figure 11 additionally contains MMD hypothesis tests that clearly show sensitivity of the parameter posterior to the prior specification.

[4] https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv

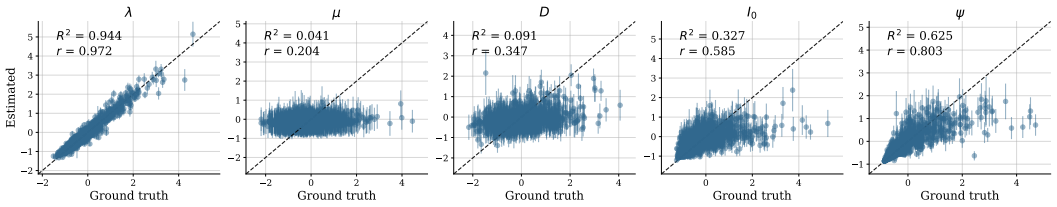


Figure 7: **Experiment 1**. Simulation-based recovery of the context-aware neural approximator used in our experiment for $\boldsymbol{\gamma} = 1.0$ (simulation budget of $2^{14} = 16\,384$).

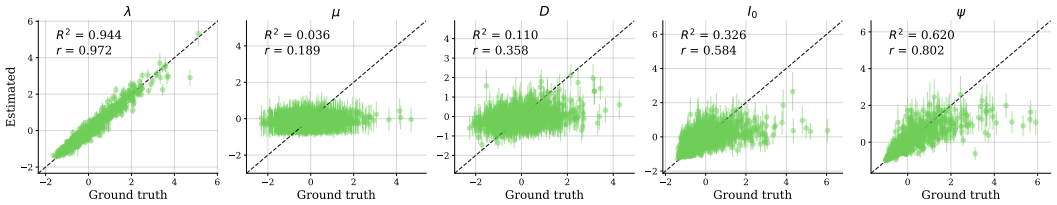


Figure 8: **Experiment 1**. Simulation-based parameter recovery of a single-prior neural approximator trained with the same configuration and simulation budget ($2^{14} = 16\,384$) as in our experiment but without C_P (i.e., on the baseline $\boldsymbol{\gamma} = 1.0$ setting).

Benchmark Details: All results of Experiment 1 except the benchmark operate in a standardized parameter space to align the different parameter scales. To eliminate the influence of standardization mismatches across the tested settings, the networks trained for the benchmark use the original (unstandardized) parameters. We further employ ensembles of size $M = 2$ to check for potential approximator sensitivity affecting the stability of the benchmarking results and observe stable results within the ensemble members (i.e., no approximator sensitivity). All networks are trained for 75 epochs.

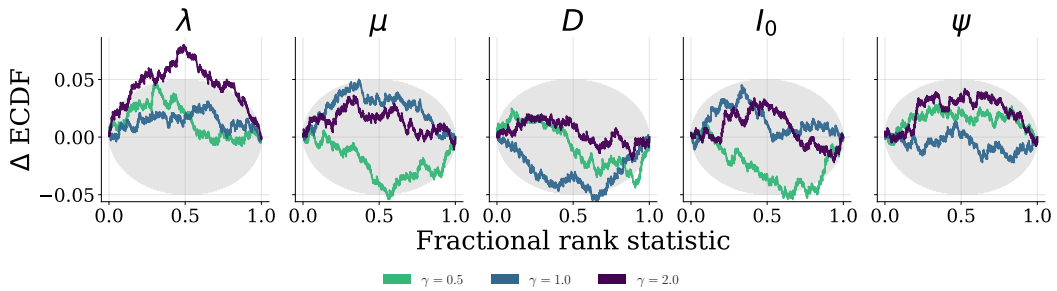


Figure 9: **Experiment 1**. Simulation-based calibration of the context-aware neural approximator used in our experiment for contexts $\gamma \in \{0.5, 1.0, 2.0\}$ (simulation budget of $2^{14} = 16\,384$).

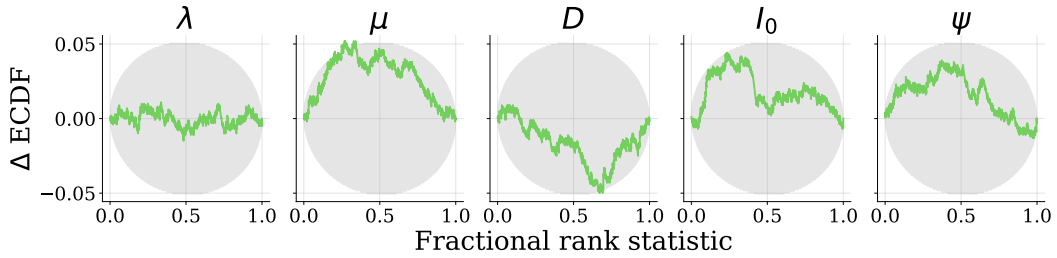
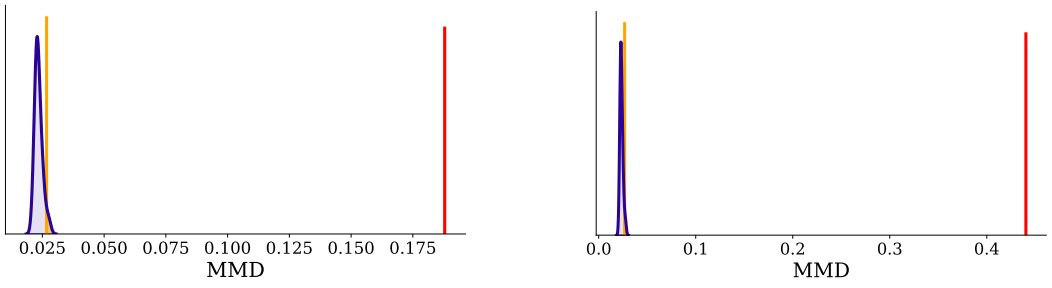


Figure 10: **Experiment 1**. Simulation-based calibration of a single-prior neural approximator trained with the same configuration and simulation budget ($2^{14} = 16\,384$) as in our experiment but without C_P (i.e., on the baseline $\gamma = 1.0$ setting).



(a) $\gamma = 0.5$ (red) vs. $\gamma = 1.0$ (blue).

(b) $\gamma = 2.0$ (red) vs $\gamma = 1.0$ (blue).

Figure 11: **Experiment 1**. MMD hypothesis tests confirm sensitivity to prior specification in the parameter space. The blue density represents samples under the null distribution of zero difference between $\gamma = 1.0$ and the scaled posteriors (red; $\gamma = 0.5$ or $\gamma = 2.0$.), the yellow lines mark the area with a $\delta = 5\%$ rejection probability.

C.3 Experiment 2: Climate Trajectory Forecasting

Data: Climate models (CM) are typically differential equations describing all relevant components of the Earth system and their couplings. Socioeconomic pathways (SSPs) provide comprehensive scenarios of future developments by describing qualitatively different trajectories of future emissions. Following standard practice, we append the associated radiative forcing in the year 2100 to the SSP identifier (e.g., SSP1-2.6 refers to the SSP1 scenario with radiative forcing of $2.6 W/m^2$). Given a CM, an SSP, and an initial condition (IC), high-dimensional trajectories of many observables can be simulated from the underlying model. Here we restrict the analysis to the air temperature at the surface (TAS).

Simulations produce trajectories $\text{TAS}^{\text{CM, SSP, IC}}(\vec{x}, t)$ where $\vec{x}$ is a spatial coordinate on the Earth's surface and t is a time between 1850 and 2100. From this, the area-weighted global mean surface air temperature (GSAT) can be computed, which is the key indicator of global warming over pre-industrial levels.

All data used in this experiment is freely available to download: For the climate model simulation outputs, we use data from the Earth System Grid Federation.[5] For the observational data set for 2022, we use data from Berkeley Earth, licensed under CC BY 4.0.[6] We include all climate models that have archived at least 10 trajectories for the future emission scenarios SSP1, SSP2, and SSP3 in our analysis. We reshape all data to a 2.5×2.5 longitude-latitude grid and compute yearly differences to the baseline period (1951 to 1980). The warming threshold is defined relative to the pre-industrial period 1850 to 1900, from which we can directly obtain the time-to-threshold parameter θ for each tuple of year, model, and trajectory ensemble.

Model Setup: Our model setup focuses on the time θ until the $1.5°$C warming threshold is reached, but can easily be adapted to arbitrary temperature thresholds. We realize a prior context C_P for θ by two discrete prior choices: First, a weakly informative prior, $\theta \sim \mathcal{U}(-40, 41)$, that encompasses the full range of values present in the training data of simulated climate warming trajectories. Second, a more informative Gaussian prior, $\mathcal{N}_+(10, 10)$, truncated to include only positive values. This prior is based on the IPCC sixth assessment report stating that the central estimate of crossing the $1.5°$C threshold lies in the early 2030s (Lee et al., 2021).

Table 7: Overview of the climate models included in each SSP emission scenario.

Climate Models	SSP1-2.6	SSP2-4.5	SSP3-7.0
ACCESS-ESM1-5	✓	✓	✓
CanESM5	✓	✓	✓
CESM2			✓
CNRM-ESM2-1		✓	
GISS-E2-1-G		✓	✓
IPSL-CM6A-LR		✓	✓
MIROC-ES2L	✓	✓	✓
MIROC6	✓		
UKESM1-0-LL	✓		✓

The likelihood is obtained from the simulated trajectories of the climate models. For a given time-to-threshold θ and climate model (encoded in the likelihood context C_L), trajectories of the respective climate model are selected in ensembles of 10. We first identify the year of threshold exceedance of the mean global surface temperature across a trajectory ensemble. [7] Afterwards, a random ensemble and trajectory are chosen. Finally, the likelihood algorithm returns the spatial temperature pattern that is θ years prior to the year of threshold exceedance in the simulated trajectory.

[5] https://esgf.llnl.gov/

[6] https://berkeleyearth.org/data/

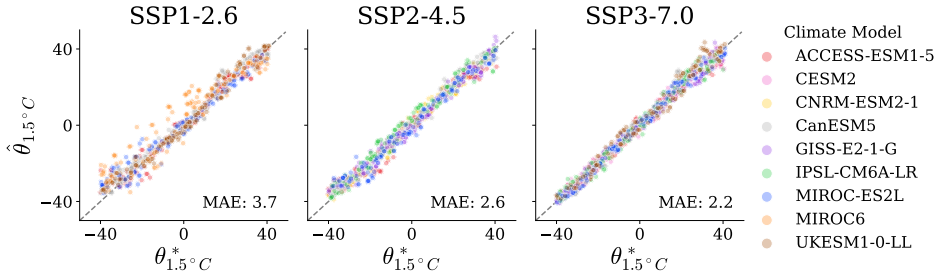
[7] Averaging over trajectory ensembles smoothes out the chaotic internal variability to obtain a more stable estimate. In contrast, the IPCC defines the year of threshold exceedance as the middle of a 20-year averaging window. Diffenbaugh & Barnes (2023) show that resulting forecasts are insensitive to the chosen definition.

Neural Network and Training: We z-standardize data and parameters before passing them to the neural approximator. As summary network, we use a dense network that parallels the architecture used in Diffenbaugh & Barnes (2023): Inputs come in the form of 72x144 temperature grids and are flattened. Two hidden layers of 25 units with ReLU activation are followed by 8 output units of learned summary statistics. During training, we additionally employ dropout regularization with a dropout probability of 0.4 on the initial layer of the summary network to mitigate overfitting. As inference network, we use a conditional invertible neural network. Since normalizing flows require more than one dimension, we add a dummy standard normal parameter.

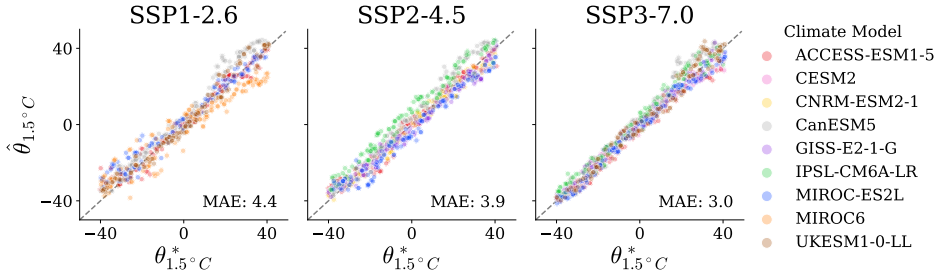
Training the neural approximator took approximately 70 minutes on a consumer notebook with a 6-core AMD Ryzen 5 5625U CPU and without a dedicated graphics card, underscoring the wide applicability of our method.

For the individual ABI instances trained for the benchmark, each instance was trained on data of a specific scenario $\times$ climate model setting. We used the same network architecture for separate and joint training to enhance comparability, but note that the hyperparameters may not be optimal for the respective data size settings. Joint training was conducted on 80 epochs, whereas we chose a smaller number of 15 epochs for the separate training to mitigate overfitting.

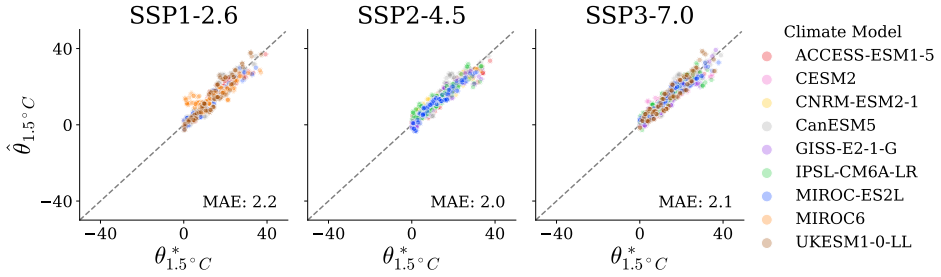
Additional Results: To validate our approach, we compute the mean absolute error (MAE) of the point estimate (posterior mean) $\hat{\theta}$ to the true value θ^* on a held-out validation set. Figure 12a shows good recovery across climate models, true time-to-threshold θ^*, and SSPs. To parallel the evaluation procedure of Diffenbaugh & Barnes (2023), who did not differentiate between the climate models, we use a flat prior via the prior context C_P and marginalize predictions over all climate models contained in C_L. Figure 12b shows that our sensitivity-aware approach does not sacrifice predictive accuracy and further enables the identification of biased estimates for MIROC6 in the SSP1-2.6 scenario as the main limitation to better performance. Figure 13 provides an additional perspective via standard simulation-based recovery and calibration plots. Overall recovery is high, but calibration plots indicate notable underconfidence of the posteriors, implying that the networks systematically overestimate the variance of time-to-threshold θ predictions. We hypothesize that this is due to an underperforming summary network which we nevertheless keep the same as in Diffenbaugh & Barnes (2023) for the sake of comparability.



(a) Weakly informative prior. Recovery with C_L information about the respective climate model given is good (total MAE of 2.0) across climate models, true time-to-threshold θ^*, and SSPs.

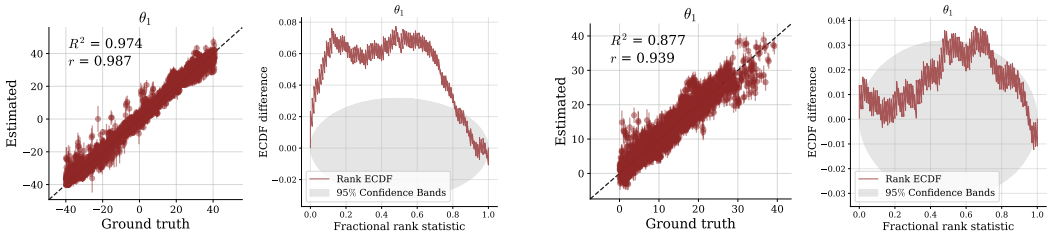


(b) Weakly informative prior. Recovery without C_L information about the respective climate model given – all predictions are not only obtained for the appropriate climate model, but all climate models contained in C_L and afterwards averaged. This leads to a validation setup comparable to Diffenbaugh & Barnes (2023). The mean absolute error (MAE) of 3.0 years for SSP3-7.0 indicates that our approach does not sacrifice predictive accuracy in comparison to Diffenbaugh & Barnes (2023), who report MAE between 2.7 and 3.8 years for the same task (eyeballed values from boxplots reported in the appendix).



(c) Informative prior. Recovery with C_L information about the respective climate model given, here restricted to positive θ values due to the truncation of the prior. Recovery is good with a total MAE of 1.2.

Figure 12: **Experiment 2**. Recovery of time-to-threshold on held-out validation data.



(a) Weakly informative prior.

(b) Informative prior.

Figure 13: **Experiment 2**. Standard simulation-based metrics of recovery and calibration on held-out validation data for both prior contexts C_P. All results are marginalized over the likelihood context space C_L (i.e., climate models and emission scenarios).

C.4 Experiment 3: Comparing Hierarchical Models of Decision-Making

Model Setup: The drift-diffusion model (DDM; Ratcliff et al., 2016; Ratcliff, 1978) models binary decision outcomes and their associated response times with the following stochastic ordinary differential equation:

$$\mathrm{d}x = v\,\mathrm{d}t + \xi\sqrt{\mathrm{d}t} \tag{22}$$

$$\xi \sim \mathcal{N}(0, 1), \tag{23}$$

with $\mathrm{d}x$ denoting the change in evidence accumulation, v denoting the rate of evidence accumulation, and ξ the noise of evidence accumulation. Additional parameters of the model include the non-decision time t_0 (e.g., encoding and motor response), the decision threshold a, and the bias towards a decision option z_r.

The Lévy flight model (LFM; Voss et al., 2019) relaxes the Gaussian noise assumption by using the more general α-stable distribution, leading to an unknown analytical form of the likelihood:

$$\mathrm{d}x = v\,\mathrm{d}t + \sigma\mathrm{d}\xi \tag{24}$$

$$\xi \sim \text{AlphaStable}(\alpha, \mu = 0, \sigma = \frac{1}{\sqrt{2}}, \beta = 0), \tag{25}$$

with the additional stability parameter α which shall also be estimated.

Additionally, there is a debate about whether the model parameters v, z_r, and t_o should have a fixed value over the course of an experiment (basic models) or be allowed to vary (full models) throughout the experiment (Lerche & Voss, 2016; Boehm et al., 2018). Therefore, we compare the following four models in this experiment:

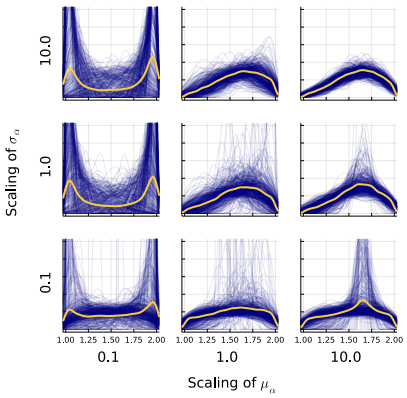
- Basic DDM ($\mathcal{M}_1$): The classic four-parameter formulation with the model parameters v, t_0, a, and z_r.
- Basic LFM ($\mathcal{M}_2$): Equals the basic DDM plus the additional stability parameter α controlling the tail behavior of the noise distribution.
- Full DDM ($\mathcal{M}_3$): Equals the basic DDM plus inter-trial variability parameters s_v, s_{t_0}, and s_{z_r}.
- Full LFM ($\mathcal{M}_4$): Equals the full DDM plus the additional stability parameter α.

As in Elsemüller et al. (2023), we reanalyze data by Wieschen et al. (2020) (provided by the original authors) containing 40 participants with 900 decision trials each and assume a uniform model prior (i.e., equal prior model probabilities). We use the same hierarchical priors as Elsemüller et al. (2023), who provided a detailed table with all prior choices. Since they are central to our experiment, we reiterate the priors leading to α_m for each participant m here:

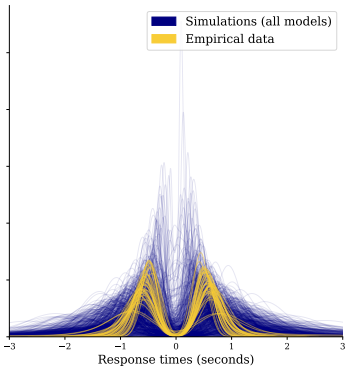
$$\begin{aligned} \mu_\alpha &\sim \mathcal{N}(1.65, 0.15/\gamma_1) \\ \sigma_\alpha &\sim \mathcal{N}_+(0.3, 0.1/\gamma_2) \\ \alpha_m &\sim \mathcal{N}_{Truncated}(\mu_\alpha, \sigma_\alpha, 1, 2) \text{ for } m = 1, \ldots, M. \end{aligned} \tag{26}$$

Our experiment investigates the sensitivity to different C_P, C_A, and C_D realizations. For C_P, we power-scale the hierarchical prior of α with a wide range of $\boldsymbol{\gamma} \in [0.1, 10]$, which would have been infeasible with existing methods since they require values close to the mid-range value of no scaling, i.e., $\boldsymbol{\gamma} = 1$ (Kallioinen et al., 2021). To ensure equal amounts of widening and shrinking of the respective distributions during training, we draw the scaling factors from independent uniform distributions in the log space $\boldsymbol{\gamma} \sim \exp(\mathcal{U}(\log(0.1), \log(10)))$. Figure 14a displays the prior predictive distribution of α under different $\boldsymbol{\gamma}$ configurations. We use 100 bootstrap samples on the trial level (i.e., within the observations of each participant) for the data context C_D and describe the details of the approximator context C_A in the following section.

Neural Network and Training: All 20 members of the employed deep ensemble constituting C_A are set up and trained independently and identically. Each network uses a hierarchical summary network consisting of two permutation invariant deep set networks (Zaheer et al., 2017) and a standard feedforward network with a softmax output layer as an inference network.



(a) Prior predictive distributions of the α parameter under power-scaling. The yellow line represents the marginal density for each configuration.



(b) Prior pushforward distributions of simulated response times per person contrasted with the empirical distributions per person. Negative response times indicate a decision for the lower decision boundary and positive times for the upper decision boundary.

Figure 14: **Experiment 3**. Prior distributions. a) Prior predictive distributions of the α parameter under maximum widening ($\gamma = 0.1$), no scaling ($\gamma = 1$), and maximum shrinkage ($\gamma = 10$) of the hyperpriors μ_α and σ_α. (b) Prior pushforward distributions of the response times simulated by the four compared models (marginalized over γ) and the empirical response times.

As in Elsemüller et al. (2023), we first pre-train each network on smaller data sets of 40 simulated participants with 100 observations each, and afterwards fine-tune on the full data size of 40 simulated participants with 900 observations. We use 30 epochs for both phases and an Adam optimizer (Kingma & Ba, 2015) with a cosine decay schedule (initial learning rates of 5×10^{-4} for pre-training and 5×10^{-5} for fine-tuning).

All computations for this experiment were performed on a single-GPU machine with an NVIDIA RTX 3070 graphics card and an AMD Ryzen 5 5600X processor. Simulating 40 000 pre-training and 8 000 fine-tuning data sets in the Julia programming language (Bezanson et al., 2017) took 23 minutes. Training the deep ensemble took 21 minutes for pre-training and 45 minutes for fine-tuning per network.

Additional Results: Figure 14b displays the prior pushforward distribution of simulated response times and the empirical response times distributions. The informative priors from Elsemüller et al. (2023) assign high densities to the central regions of the empirical distribution. Nevertheless, the results of the typical set approach (Nalisnick et al., 2019; Morningstar et al., 2021) in Figure 15 flag the empirical data as out-of-distribution of each deep ensemble member.

To ensure that the inclusion of C_P in the amortization scope does not lead to a substantially worsened approximation performance, we trained an additional deep ensemble without C_P. Table 8 displays validation performance and empirical approximations for the ensemble including C_P and Table 9 for the ensemble without C_P. Including C_P does neither lead to a substantial drop in performance nor qualitatively different model comparison results despite power-scaling over a wide range.

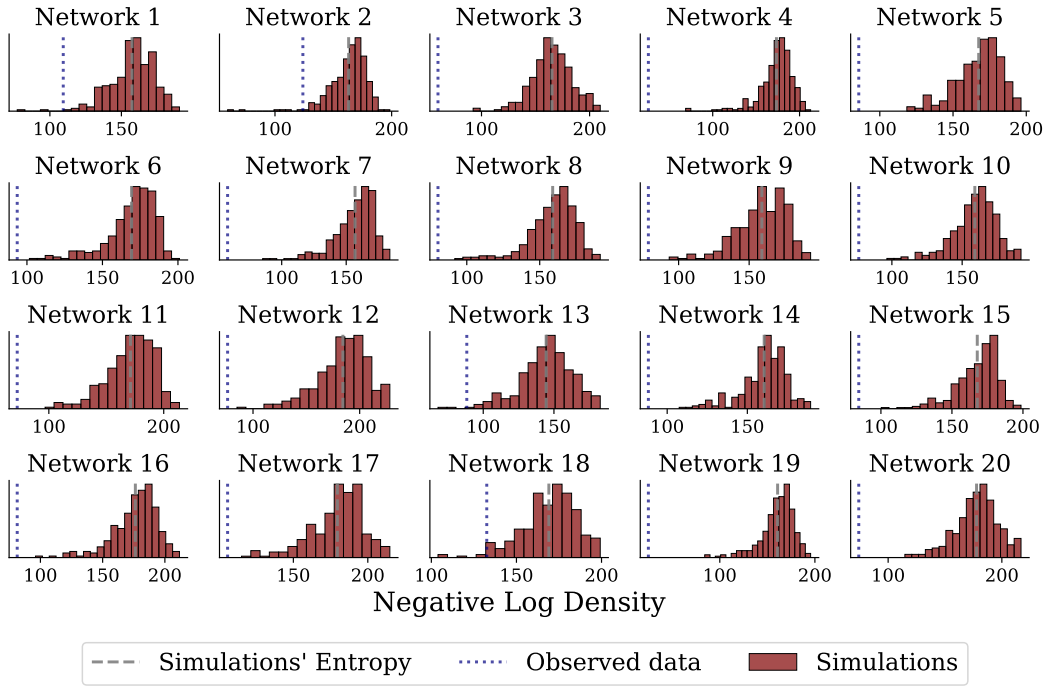


Figure 15: **Experiment 3**. The typical set of learned summary statistics from the model simulations contrasted with the density of the learned summary statistics from the observed data (both distributions are marginalized over C_P but separated for each ensemble member of C_A). The empirical data is flagged as out-of-distribution for each ensemble member.

Table 8: **Experiment 3**. SA-ABI validation performance on 8 000 held-out simulations and predictions on the empirical data of the deep ensemble trained on power-scaled C_P context from 0.1 to 10.

	Validation Performance				**Empirical Predictions**			
	ECE	Brier Score	MAE	Accuracy	$\mathcal{M}_1$	$\mathcal{M}_2$	$\mathcal{M}_3$	$\mathcal{M}_4$
Network 1	0.01	0.01	0.02	0.99	0.00	0.00	1.00	0.00
Network 2	0.01	0.03	0.05	0.96	0.00	0.00	0.92	0.08
Network 3	0.01	0.01	0.01	0.99	0.00	0.00	0.17	0.83
Network 4	0.01	0.01	0.01	0.99	0.00	0.00	0.50	0.50
Network 5	0.01	0.01	0.01	0.99	0.00	0.00	0.96	0.04
Network 6	0.01	0.01	0.01	0.99	0.00	0.00	0.02	0.98
Network 7	0.01	0.01	0.01	0.99	0.00	0.00	0.75	0.25
Network 8	0.01	0.01	0.02	0.99	0.00	0.00	1.00	0.00
Network 9	0.01	0.01	0.01	0.99	0.52	0.03	0.25	0.21
Network 10	0.01	0.02	0.03	0.97	0.00	0.00	0.85	0.15
Network 11	0.01	0.01	0.01	0.99	0.00	0.00	0.37	0.63
Network 12	0.01	0.01	0.01	0.99	0.00	0.00	0.00	1.00
Network 13	0.01	0.01	0.01	0.99	0.00	0.00	0.99	0.01
Network 14	0.01	0.02	0.03	0.98	0.00	0.00	0.99	0.01
Network 15	0.01	0.01	0.01	0.99	0.00	0.96	0.00	0.04
Network 16	0.00	0.00	0.01	0.99	0.97	0.00	0.03	0.00
Network 17	0.01	0.02	0.02	0.98	0.00	0.00	0.62	0.38
Network 18	0.01	0.02	0.03	0.98	0.00	0.00	0.91	0.09
Network 19	0.01	0.01	0.01	0.99	0.00	0.04	0.49	0.46
Network 20	0.01	0.01	0.01	0.99	0.00	0.00	0.08	0.92
Average	0.01	0.01	0.02	0.99	0.07	0.05	0.54	0.33
Std. Deviation	0.00	0.01	0.01	0.01	0.24	0.21	0.40	0.36

Table 9: **Experiment 3**. ABI validation performance on 8 000 held-out simulations and predictions on the empirical data of the deep ensemble trained without C_P context.

	Validation Performance				**Empirical Predictions**			
	ECE	Brier Score	MAE	Accuracy	$\mathcal{M}_1$	$\mathcal{M}_2$	$\mathcal{M}_3$	$\mathcal{M}_4$
Network 1	0.00	0.00	0.00	1.00	0.00	0.00	0.01	0.99
Network 2	0.00	0.01	0.01	0.99	0.00	0.00	0.00	1.00
Network 3	0.01	0.01	0.02	0.98	0.00	0.00	0.99	0.01
Network 4	0.01	0.02	0.03	0.98	0.00	0.00	0.60	0.40
Network 5	0.01	0.01	0.03	0.98	0.00	0.00	0.88	0.12
Network 6	0.00	0.00	0.00	1.00	1.00	0.00	0.00	0.00
Network 7	0.00	0.00	0.00	1.00	0.00	0.00	0.00	1.00
Network 8	0.01	0.01	0.02	0.98	0.00	0.00	0.00	1.00
Network 9	0.00	0.00	0.00	1.00	0.00	1.00	0.00	0.00
Network 10	0.00	0.00	0.00	1.00	0.72	0.16	0.11	0.00
Network 11	0.00	0.00	0.00	1.00	0.01	0.42	0.00	0.56
Network 12	0.01	0.02	0.04	0.97	0.00	0.00	0.91	0.09
Network 13	0.01	0.01	0.01	0.99	0.00	0.00	0.00	1.00
Network 14	0.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00
Network 15	0.01	0.01	0.01	0.99	0.00	0.00	1.00	0.00
Network 16	0.00	0.00	0.00	1.00	0.00	0.00	0.98	0.02
Network 17	0.00	0.00	0.00	1.00	0.00	0.25	0.01	0.74
Network 18	0.00	0.00	0.00	1.00	0.00	0.00	0.86	0.14
Network 19	0.01	0.01	0.01	0.99	0.00	0.00	0.00	1.00
Network 20	0.01	0.02	0.04	0.97	0.00	0.00	0.88	0.11
Average	0.00	0.01	0.01	0.99	0.09	0.09	0.41	0.41
Std. Deviation	0.01	0.01	0.01	0.01	0.27	0.24	0.46	0.44

Publication III

C

Publication	**Link to online full-text**

Elsemüller, L., Pratz, V., von Krause, M., Voss, A., Bürkner, P. C., & Radev, S. T. (2025). Does unsupervised domain adaptation improve the robustness of amortized Bayesian inference? A systematic evaluation. Under review at *Transactions on Machine Learning Research*.

The reprint of the publication can be found in the following.

Under review as submission to TMLR

Does Unsupervised Domain Adaptation Improve the Robustness of Amortized Bayesian Inference? A Systematic Evaluation

Anonymous authors
Paper under double-blind review

Abstract

Neural networks are fragile when confronted with data that significantly deviates from their training distribution. This is true in particular for simulation-based inference methods, such as neural amortized Bayesian inference (ABI), where models trained on simulated data are deployed on noisy real-world observations. Recent robust approaches employ unsupervised domain adaptation (UDA) to match the embedding spaces of simulated and observed data. However, the lack of comprehensive evaluations across different domain mismatches raises concerns about the reliability in high-stakes applications. We address this gap by systematically testing UDA approaches across a wide range of misspecification scenarios in silico and practice. We demonstrate that aligning summary spaces between domains effectively mitigates the impact of unmodeled phenomena or noise. However, the same alignment mechanism can lead to failures under prior misspecifications – a critical finding with practical consequences. Our results underscore the need for careful consideration of misspecification types when using UDA to increase the robustness of ABI.

1 Introduction

Synthetic data can augment numerous real-world applications (Savage, 2023), including complex statistical workflows (Bürkner et al., 2025). In line with this perspective, amortized Bayesian inference (ABI; Gershman & Goodman, 2014) redefines the classical sampling problem in Bayesian estimation by training generative neural networks on simulations derived from computational models (Bürkner et al., 2023; Cranmer et al., 2020). The trained neural networks are then deployed to efficiently solve inference tasks as diverse as inferring evolutionary parameters (Avecilla et al., 2022) or gravitational waves (Pacilio et al., 2024).

Evidently, the faithfulness of any simulation-based method rests on the critical assumption that statistical patterns learned from simulated data can be extrapolated to real observations. This assumption inevitably situates ABI in a domain-shift regime, exacerbated by the degree of potential mismatch between model simulations and reality. As such, *robustness to model misspecification* has been identified as the primary challenge for amortized methods in different fields (Dingeldein et al., 2024; Rainforth et al., 2024; Cannon et al., 2022).

Unsupervised Domain Adaptation (UDA) studies the transfer of knowledge from a labeled source domain to an unlabeled target domain. It aims to mitigate domain shifts by aligning the *embedding spaces* of the two domains. This property makes UDA a promising approach for addressing domain shifts in ABI, as the latter typically combines inference with embedding high-dimensional data into *learned summary statistics* (Radev et al., 2020; Chan et al., 2018). Indeed, recent research has underscored the critical role of *in-distribution* summary statistics for achieving robust simulation-based inference (Schmitt et al., 2023; Frazier et al., 2024; Huang et al., 2023; Wehenkel et al., 2024).

So far, only two pioneering studies (Swierc et al., 2024; Huang et al., 2023) have explored the potential of UDA methods for robustifying simulation-based inference. Both approaches align the embedding spaces by minimizing the maximum mean discrepancy (MMD; Gretton et al., 2012) between simulated and observed

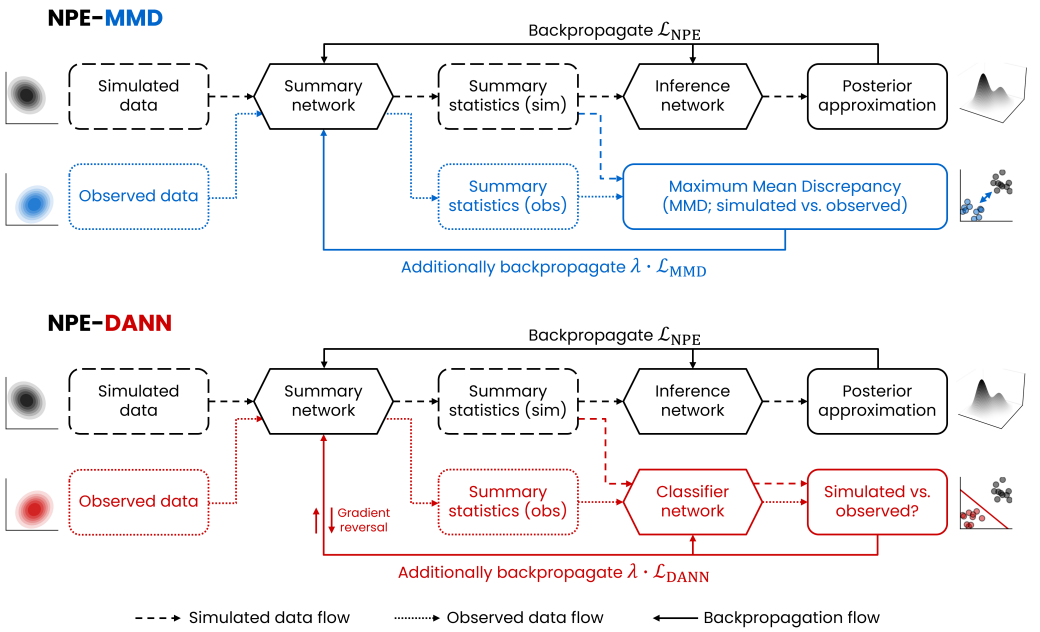


Figure 1: Schematic overview of NPE-UDA methods that combine neural posterior estimation (NPE) with unsupervised domain adaptation (UDA). Standard NPE training optimizes posterior approximation in a simulation-based training loop. NPE-UDA approaches introduce observed data into the training procedure, targeting performance improvements in the (possibly shifted) observed domain via *domain alignment in summary space*. NPE-MMD (maximum mean discrepancy) directly minimizes the distance between distributions, whereas NPE-DANN (domain-adversarial neural networks) uses adversarial competition between an auxiliary domain classifier and the summary network.

summary statistics (see Figure 1). However, despite their promising results, several gaps remain. In particular, Huang et al. (2023) did not make an explicit connection to UDA and explored a non-amortized approach. While Swierc et al. (2024) acknowledged the connection to UDA, their work focused on a specific gravitational lensing application. Both works mainly evaluated likelihood misspecification, leaving the behavior under prior shifts largely untapped. Finally, the utility of the widely used UDA method *domain-adversarial neural networks* (DANN; Ganin et al., 2016) remains completely unexplored. To address these gaps, we make the following contributions:

1. We adapt domain-adversarial neural networks for neural posterior estimation (NPE; see Figure 1) and evaluate their utility for robust amortized Bayesian inference.

2. We categorize robust methods by inference targets, enabling a theoretical assessment of their strengths and limitations based on the source of misspecification.

3. We evaluate the robustness of UDA-based ABI methods across multiple misspecification scenarios in several benchmarks, confirming the central role of the source of misspecification: whereas UDA improves performance under likelihood shifts, it is detrimental under prior shifts.

2 Background

Amortized Bayesian Inference (ABI) Amortized methods (Gershman & Goodman, 2014; Ritchie et al., 2016; Le et al., 2017) are a subset of the simulation-based inference (SBI; Cranmer et al., 2020) family. Their defining characteristic is the ability to perform zero-shot inference on model parameters $\boldsymbol{\theta}$ by learning a conditional distribution $q(\boldsymbol{\theta} \mid \boldsymbol{x})$ that requires no further training or auxiliary algorithms for new data $\boldsymbol{x}$ (see Appendix A for details). The *amortized distribution* $q(\boldsymbol{\theta} \mid \boldsymbol{x})$ is typically parameterized by an inference network, a generative neural network that can generate random samples $\boldsymbol{\theta} \sim q(\boldsymbol{\theta} \mid \boldsymbol{x})$ – akin to a standard Markov chain Monte Carlo (MCMC) sampler, but orders of magnitude faster. Often, the inference network is preceded by a summary network ϕ that compresses raw observations to learned summary statistics $\phi(\boldsymbol{x})$, leveraging probabilistic symmetries in the data (Radev et al., 2020; Chen et al., 2021). Following a potentially expensive simulation-based training phase, the network can be queried with any *new* data $\boldsymbol{x}_{\text{new}}$ to rapidly approximate the target distribution $p(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{new}})$. Initially dismissed as inefficient compared to sequential methods optimized for a specific data set $\boldsymbol{x}_{\text{obs}}$ (Papamakarios & Murray, 2016), amortized methods have since achieved notable successes across various domains (Bürkner et al., 2023; Zammit-Mangion et al., 2024).

Unsupervised Domain Adaptation (UDA) UDA is a subfield of transductive transfer learning where labeled data is only available for the source domain $\mathcal{D}_S = \{(\boldsymbol{x}_S^i, \boldsymbol{y}_S^i)\}_{i=1}^{N_S}$, distributed according to $p_S(\boldsymbol{x}, \boldsymbol{y})$, but not for the target domain $\mathcal{D}_T = \{\boldsymbol{x}_T^i\}_{i=1}^{N_T}$, distributed according to $p_T(\boldsymbol{x}_T, \boldsymbol{y}_T)$ (Johansson et al., 2019). UDA methods are based on the seminal theoretical works of Ben-David et al. (2006; 2010), who introduced generalization bounds for binary classification tasks that bound the risk in the target domain R_T of a hypothesis $h \in \mathcal{H}$:

$$R_T(h) \leq R_S(h) + d_{\mathcal{H}\Delta\mathcal{H}}(p_S, p_T) + \lambda_{\mathcal{H}}, \tag{1}$$

where $R_S(h)$ is the source domain risk, $d_{\mathcal{H}\Delta\mathcal{H}}(p_S, p_T)$ measures the divergence between the domain distributions, and $\lambda_{\mathcal{H}}$ is the minimum combined risk of the optimal hypothesis, $\lambda_{\mathcal{H}} = \inf_{h \in \mathcal{H}} |R_S(h) + R_T(h)|$ (Johansson et al., 2019). This suggests that domain adaptation from $\mathcal{D}_S$ to $\mathcal{D}_T$ can be facilitated by minimizing the divergence between the marginal domain distributions. Although the domain distribution divergence cannot be reduced directly, the representation divergence $d(\phi(\boldsymbol{x}_S), \phi(\boldsymbol{x}_T))$ from a transformation $\phi : \mathcal{X} \to \mathcal{Z}$ can be readily minimized (Ben-David et al., 2006). The core idea of UDA is thus twofold: (i) to minimize the source-domain error $R_S(h)$ during training, and (ii) to align the domain representations $\phi(\boldsymbol{x}_S)$ and $\phi(\boldsymbol{x}_T)$ to achieve *domain-invariant* embeddings that generalize to the target domain. UDA methods include discrepancy-based approaches, which minimize statistical divergences like the MMD between source and target embeddings (Tzeng et al., 2014), and, most prominently, adversarial-based approaches, such as domain-adversarial neural networks (DANN) (Ganin et al., 2016), which learn domain-invariant embeddings via a minimax game between a feature extractor and a domain classifier.

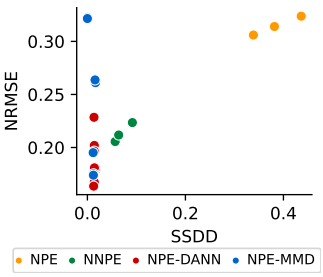


Figure 2: **Experiment 3**: Summary space domain distance (SSDD; MMD) vs. normalized root mean squared error (NRMSE) for row deletions. We observe a sweet spot of domain alignment without losing important information.

The vast majority of UDA research, including its theoretical foundations, focuses on classification tasks (Redko et al., 2022; Ben-David et al., 2010; Liu et al., 2022), with some works on regression tasks (Cortes & Mohri, 2014; Mansour et al., 2009) and only a few on generative tasks (Uppaal et al., 2024). More recently, UDA methods have been successfully applied to address simulation-to-reality (sim2real) problems (Ćiprijanović et al., 2020; Swierc et al., 2023; Kong et al., 2023) which seek to generalize patterns learned in a simulated source domain to a real-world target domain. These problems seem pertinent to any simulation-based method relying on data generation from imperfect models.

From Simulated to Real Domains The preceding discussion makes the connection between UDA and ABI immediately apparent: When the distance between the data distribution $p(\boldsymbol{x}_{\text{obs}})$ and the model-implied distribution $p(\boldsymbol{x}) = \mathbb{E}_{p(\boldsymbol{\theta})}[p(\boldsymbol{x} \mid \boldsymbol{\theta})]$ is non-zero, the risk of extrapolation error for atypical data $\boldsymbol{x}_{\text{obs}}$ may increase. Indeed, this behavior has been observed repeatedly in the context of SBI (Ward et al., 2022;

Schmitt et al., 2023; Huang et al., 2023; Frazier et al., 2024; Kelly et al., 2025). In particular, Frazier et al. (2024) notes that ABI is especially prone to "extrapolation bias" for observed summary statistics $\phi(\boldsymbol{x})$ that are far in the tails of the model-implied (i.e., prior predictive) density $p(\boldsymbol{x})$. The scenario can be equivalently stated by invoking the notion of a *typical set* (Cover & Thomas, 2012), which denotes a subset of the support of $p(\boldsymbol{x})$ where most of the probability mass concentrates around the entropy $H(p)$:

$$A_\epsilon = \{\boldsymbol{x} \in \mathcal{X} : |-\log p(\boldsymbol{x}) - H(p)| \leq \epsilon\}\,. \tag{2}$$

Accordingly, for any problem-specific ϵ, observed data $\boldsymbol{x}_{\text{obs}} \notin A_\epsilon$ may result in a biased posterior approximation $q(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{obs}})$. As further noted in the comprehensive theoretical exposition by Frazier et al. (2024), matching summary statistics $\phi(\boldsymbol{x}_{\text{obs}})$ to the model-implied distribution of $\phi(\boldsymbol{x})$ can be a useful heuristic for reducing extrapolation bias. This observation harmonizes with the UDA literature as well (Ben-David et al., 2010). Pre-asymptotically, the success of such matching depends on multiple factors, including (i) the type and hyperparameters of the matching method (see Figure 2); (ii) the degree and nature of domain mismatch; (iii) the complexity of the learning problem; and (iv) even the choice of success metric. Thus, a primary goal of this work is to systematically examine the effects of these factors on a variety of metrics that can index potential robustness gains.

3 Methods

3.1 Unsupervised Domain Adaptation for Amortized Bayesian Inference

We start with the observation that model misspecification in ABI (Schmitt et al., 2023), and also more generally in neural SBI, can naturally be framed as an UDA problem: Ground-truth parameter values are only available for the simulated source domain $\mathcal{D} = \{(\boldsymbol{x}^i, \boldsymbol{\theta}^i)\}_{i=1}^N$ but not the observed target domain $\mathcal{D}_{\text{obs}} = \{\boldsymbol{x}_{\text{obs}}^i\}_{i=1}^{N_{\text{obs}}}$. In most machine learning applications, the collection of reliable ground-truth values is costly but feasible, whereas in SBI, collecting ground-truth parameter values $\boldsymbol{\theta}_{\text{obs}}$ of observed data is typically impossible. A general optimization objective for NPE-UDA methods can be formulated by extending the standard negative log-posterior NPE objective:

$$\begin{aligned}\mathcal{L}_{\text{NPE-UDA}}(q, \phi) :&= \mathcal{L}_{\text{NPE}} + \lambda \cdot \mathcal{L}_{\text{UDA}} &(3)\\ &= \mathbb{E}_{p(\boldsymbol{\theta},\boldsymbol{x})p(\boldsymbol{x}_{\text{obs}})}\big[-\log q(\boldsymbol{\theta} \mid \phi(\boldsymbol{x})) + \lambda \cdot d(\phi(\boldsymbol{x}), \phi(\boldsymbol{x}_{\text{obs}}))\big], &(4)\end{aligned}$$

where λ controls the regularization weight of the UDA loss and $d(\cdot, \cdot)$ is a divergence measure that attains its global minimum if and only if $\phi(\boldsymbol{x}) = \phi(\boldsymbol{x}_{\text{obs}})$.

$\mathcal{L}_{\text{NPE-UDA}}$ incurs a trade-off between approximation performance in the simulated domain and domain difference in the summary space, depending on the degree of domain mismatch. In the well-specified case, $p(\boldsymbol{x}) = p(\boldsymbol{x}_{\text{obs}})$, $\mathcal{L}_{\text{NPE-UDA}}$ reduces to the standard NPE loss. In the misspecified case, $p(\boldsymbol{x}) \neq p(\boldsymbol{x}_{\text{obs}})$, the summary network ϕ optimizes the summary statistics to both *maximize information extraction* in the simulated domain and *minimize domain shift* in summary space. Thereby, the inference network $q(\boldsymbol{\theta} \mid \phi(\boldsymbol{x}))$ needs to rely on domain-invariant information shared between the simulated and the observed domain.

The common UDA assumption that there exists a low-error hypothesis for both domains (Redko et al., 2022, cf. Eq.1) suggests a λ-dependent upper bound on the amount of domain shift that can be rectified by NPE-UDA methods. However, finding an optimal value for λ despite missing target labels (i.e., ground-truth parameters in SBI) at test time is an open problem in UDA research (Zellinger et al., 2021; Musgrave et al., 2022). Thus, we expect this problem to carry over to NPE-UDA methods as well. Next, we formulate two NPE-UDA variants based on popular UDA methods with strong performance on established benchmarks (Musgrave et al., 2021).

3.2 NPE-MMD

The maximum mean discrepancy (MMD; Gretton et al., 2012) is a popular probability integral metric in SBI, since it can be efficiently estimated from a finite number of samples (Bischoff et al., 2024; Schmitt et al., 2023). For the same reason, it has been employed by various UDA works (Pan et al., 2010; Tzeng et al.,

2014; Long et al., 2015) to measure the divergence between (transformed) samples from different domains. We categorize the combination of NPE and UDA based on MMD, such as the variants of Huang et al. (2023) and Swierc et al. (2024), as NPE-MMD. Choosing the MMD as $\mathcal{L}_{\text{UDA}}$, Eq. 3 becomes

$$\mathcal{L}_{\text{NPE-MMD}}(q, \phi) := \mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})}\big[- \log q(\boldsymbol{\theta} \,|\, \phi(\boldsymbol{x}))\big] + \lambda \cdot \text{MMD}^2\big[\phi(\boldsymbol{x}) \,||\, \phi(\boldsymbol{x}_{\text{obs}})\big]. \tag{5}$$

The most important hyperparameter of NPE-MMD is the choice of kernel in the sample-based MMD estimator. In our experiments, using a sum of inverse multiquadric kernels (Ardizzone et al., 2018) led to the most stable training dynamics, but other choices have been explored in the context of robust ABI as well, such as (sums of) Gaussian kernels (Schmitt et al., 2023; Huang et al., 2023).

3.3 NPE-DANN

Domain-adversarial neural networks (DANN; Ganin et al., 2016), which have not been considered for NPE to date, introduce a domain classifier $\psi(\cdot)$ to reduce domain distance. Unlike typical adversarial training, which alternates between objectives, DANN achieves minimax optimization in a single-step update via a gradient reversal layer (Ganin et al., 2016). This layer flips the gradient sign from the classifier to the feature extractor (e.g., summary network) ϕ during backpropagation, encouraging the feature extractor to generate less domain-specific summary statistics. Similarly to NPE-MMD, DANN can be integrated into Eq. 3 to achieve NPE-DANN:

$$\mathcal{L}_{\text{NPE-DANN}}(q, \phi, \psi) := \mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})}\big[- \log q(\boldsymbol{\theta} \,|\, \phi(\boldsymbol{x}))\big] + \lambda \cdot \mathcal{L}_D(\psi, \phi). \tag{6}$$

The discriminator loss $\mathcal{L}_D$ is given by:

$$\mathcal{L}_D(\psi, \phi) := -\mathbb{E}_{p(\boldsymbol{x})}\big[\log(p(\psi(\phi(\boldsymbol{x}))))\big] - \mathbb{E}_{p(\boldsymbol{x}_{\text{obs}})}\big[\log(1 - p(\psi(\phi(\boldsymbol{x}_{\text{obs}}))))\big], \tag{7}$$

where ψ is the domain classifier and the equation represents the binary cross-entropy loss on the domains, where a gradient reversal layer enables updating ϕ and ψ in opposing directions.

While DANN is a powerful and popular UDA method (Zhou et al., 2022), it has two important drawbacks. First, the unstable training dynamics and convergence issues generally associated with adversarial learning can also occur with DANN (Sener et al., 2016; Sun et al., 2019). Second, adversarial training adds new hyperparameters, including the domain classifier architecture, an optional weight for gradient reversal balance (Ganin et al., 2016), and stabilization techniques like label smoothing (Zhang et al., 2023). Notably, although λ is a shared hyperparameter in NPE-MMD and NPE-DANN, its effect on training dynamics will vary across applications due to differing $\mathcal{L}_{\text{UDA}}$ scales.

3.4 What Is the Target of Robustness?

To better understand the strengths and limitations of robust methods, including NPE-UDA, we suggest to distinguish between the following inference goals:

- **Target 1**: The analytic (true) posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{obs}}) \propto p(\boldsymbol{x}_{\text{obs}} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$ of the assumed probabilistic model given the observed data $\boldsymbol{x}_{\text{obs}}$.
- **Target 2**: A posterior $p(\boldsymbol{\theta} \mid \widetilde{\boldsymbol{x}}_{\text{obs}}) \propto p(\widetilde{\boldsymbol{x}}_{\text{obs}} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$ of the assumed probabilistic model given *adjusted data* $\widetilde{\boldsymbol{x}}_{\text{obs}}$.
- **Target 3**: A posterior $\widetilde{p}(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{obs}}) \propto p(\boldsymbol{x}_{\text{obs}} \mid \boldsymbol{\theta})\, \widetilde{p}(\boldsymbol{\theta})$ from an *adjusted prior* $\widetilde{p}(\boldsymbol{\theta})$ given the observed data $\boldsymbol{x}_{\text{obs}}$.

Target 1 is the most common target in Bayesian inference. Classical approximation methods such as MCMC almost always consider this target (Carpenter et al., 2017). **Target 2**, an explicit deviation from the true posterior, is often targeted by methods that seek to improve the robustness of Bayesian inference (see 4). Their goal is to reduce the influence of unmodeled phenomena in $\boldsymbol{x}_{\text{obs}}$, such as additional noise or external

contamination, by approximating a target posterior $p(\boldsymbol{\theta} \mid \widetilde{\boldsymbol{x}}_{\text{obs}})$ based on denoised or uncontaminated data $\widetilde{\boldsymbol{x}}_{\text{obs}}$. This can be achieved either explicitly, by transforming $\boldsymbol{x}_{\text{obs}}$ into $\widetilde{\boldsymbol{x}}_{\text{obs}}$, or implicitly, by using an *adjusted (implicit) likelihood* $\widetilde{p}(\boldsymbol{x}_{\text{obs}} \mid \boldsymbol{\theta})$.

Since **Target 2** implies ignoring parts of the data that are in disagreement with the assumed probabilistic model, we expect corresponding methods to perform worse under prior misspecification: When a data-generating parameter $\boldsymbol{\theta}^*$ is impossible or highly unlikely under the assumed prior, *ignoring conflicting information effectively reduces the amount of information available to counteract a poorly chosen prior*. Generalized Bayes approaches (Bissiri et al., 2016) also aim to reduce the influence of undesired parts of the data. They move away from the classical Bayes rule by replacing the likelihood with a loss function, which can be interpreted as an adjusted likelihood $\widetilde{p}(\boldsymbol{x}_{\text{obs}} \mid \boldsymbol{\theta})$ according to **Target 2**. Lastly, **Target 3** can directly reduce the impact of prior misspecification by adjusting the prior based on $\boldsymbol{x}_{\text{obs}}$. However, compared to **Target 2**, it is more challenging to conceptualize the desired target priors $\widetilde{p}(\boldsymbol{\theta})$ and posteriors $\widetilde{p}(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{obs}})$ under model misspecification.

Given this categorization, what is the target of NPE-UDA? Unsurprisingly, the classic NPE loss $\mathcal{L}_{\text{NPE}}$ aims at **Target 1**. In contrast, the additional $\mathcal{L}_{\text{UDA}}$ loss governs the alignment of the summary space between simulated and observed data, effectively adjusting the observed data seen by the model. Thus, $\mathcal{L}_{\text{UDA}}$ introduces a shift towards **Target 2**, with λ governing its relative importance compared to **Target 1**. As hypothesized above, methods aiming at **Target 2** may not perform well under prior misspecification, which is confirmed for the NPE-UDA methods throughout our experiments. While Huang et al. (2023) suggested that their NPE-MMD variant is robust to prior mean shift, this conclusion was based on a single tested $\boldsymbol{x}_{\text{obs}}$ and our comprehensive evaluation could not replicate the result.

In line with our hypothesis and empirical results, Huang et al. (2023) observed that increasing values of λ encourage trading off the information content of $\boldsymbol{x}$ to minimize the domain distance in summary space, leading the posterior to converge to the assumed prior $p(\boldsymbol{\theta})$. Thus, the critical importance of the tunable hyperparameter λ in UDA contexts (Zellinger et al., 2021) directly translates to ABI applications, where λ controls a trade-off between *improving* approximation under likelihood misspecification and *degrading* approximation under prior misspecification.

4 Related Work

Robust Neural SBI Robustness in neural SBI has become a rapidly growing area of research, with most approaches enhancing robustness for a single data set at the cost of amortization, e.g., due to additional MCMC runs or post-hoc corrections. The majority of these approaches focuses on **Target 2** by incorporating an misspecification model (Ward et al., 2022), shifting observed summary statistics with low support (Kelly et al., 2023), reducing the influence of unmodeled data shifts via generalized SBI (Gao et al., 2023), or using the single-data-set NPE-MMD variant previously discussed (Huang et al., 2023). Focusing on **Target 1**, Siahkoohi et al. (2023) highlighted the role of the inference network's latent space in domain shifts and proposed a latent space correction based on the observed data $\boldsymbol{x}_{\text{obs}}$. Differently, Wang et al. (2024) focus on **Target 3** by using an upfront ABC run to filter the part of the parameter space causing the highest discrepancy between $\boldsymbol{x}$ and $\boldsymbol{x}_{\text{obs}}$.

Robust ABI In contrast, research on robustifying inference while retaining amortization has been sparse. Extending the scope of the training data via additive noise (Cranmer et al., 2020; Bernaerts et al., 2023), such as the spike-and-slab noise approach of Noisy NPE (NNPE; Ward et al., 2022), can be seen as a light modification to the simulator-implied likelihood of **Target 2**, but requires strong assumptions about the corruption process. Wehenkel et al. (2024) also approach **Target 2** by framing domain shift as an optimal transport problem in summary space, but this requires *observed* "ground-truth" parameters $\boldsymbol{\theta}^*_{\text{obs}}$ that are difficult to obtain in most ABI settings. Swierc et al. (2024) provided evidence for the potential of NPE-MMD for robust ABI but focused their evaluation solely on a gravitational lensing application with synthetically added noise. Finally, Glöckler et al. (2023) proposed an efficient regularization technique that can increase robustness against adversarial attacks and attain more reliable performance under **Target 1**.

5 Experiments

In all experiments, we benchmark NPE-MMD and NPE-DANN against an NPE baseline as well as NNPE (Ward et al., 2022) as an instantiation of a simple additive noise training modification (Cranmer et al., 2020; Bernaerts et al., 2023). The two existing works on NPE-MMD approaches mainly evaluated performance against contamination (Huang et al., 2023), where a fraction of the sample is replaced with corrupted observations (Huber, 1981), or noise applied to all observations (Swierc et al., 2024). Both of these scenarios are cases of likelihood misspecification where ignoring noise is desirable (**Target 2**). That is, the inference target is the posterior $p(\boldsymbol{\theta} \mid \widetilde{\boldsymbol{x}})$ of the assumed probabilistic model given the uncontaminated or (implicitly) denoised data set $\widetilde{\boldsymbol{x}}$.

Experiment 1 introduces a canonical contamination setting, in which we explore the sensitivity of NPE-UDA methods' to hyperparameters using Bayesian optimization. Afterwards, we expand the scope by comprehensively evaluating various likelihood/data and prior misspecification scenarios to obtain clearer insights into the strengths and limitations of the robust methods. Experiment 2 starts with a simple and controllable setting that allows for comparing the NPE-UDA methods not only against standard NPE and NNPE (Ward et al., 2022), but also to the analytic posterior under **Target 1**. Experiment 3 explores whether the result patterns can be replicated in a challenging setting with a high-dimensional parameter space. Lastly, since we are ultimately interested in the robustness of NPE in genuine scientific applications, Experiment 4 tests the methods on a massive real-world data set of human decision-making (von Krause et al., 2022).

We evaluate a range of metrics to enable a holistic assessment of the compared methods:

- **Parameter space performance metrics**: (i) Negative log likelihood (NLL) as a standard measure of posterior density estimation (ii); normalized root mean squared error (NRMSE) to measure approximation error; (iii) expected calibration error (ECE) to measure the fidelity of credible intervalst; (iv) posterior contraction (PC) to measure information gain from prior to posterior.
- **Data space performance metrics**: (i) Posterior predictive distance (PPD) to the observed data, which is the standard approach but has the disadvantage that the observed data contains the noise that a robust method should ignore, and (ii) PPD to data *resimulated* from the ground-truth parameters, a modification that allows PPD to represent the distance to well-specified (e.g., denoised) data. In Experiment 5.4, we approximate the denoised reference via intensive data pre-processing.
- **Network space metrics**: (i) Summary space domain distance (SSDD) measuring domain alignment via MMD, (ii) SSDD via the classifier two-sample test (C2ST), and (iii) inference network latent distance (INLD) to the base distribution of the generative neural network (diagonal Gaussian in all our experiments), which has recently been highlighted as the central mediator of posterior errors (Siahkoohi et al., 2023).

Appendix B provides details on metrics, experimental setups, network architectures, training and evaluation procedures, as well as additional results. Code for reproducing all results from this paper is available at `[ANONYMIZED DURING REVIEW]`.

5.1 Experiment 1 - Ricker: Hyperparameter Exploration

Setup Our first experiment explores the properties of amortized NPE-UDA methods in a classic contamination misspecification scenario. The inference task consists in inferring two parameters of the popular Ricker model of population dynamics (Wood, 2010; Ricker, 1954). Using the same setting, previous work by Huang et al. (2023) demonstrated the potential of non-amortized NPE-MMD, which specializes inference for a single seen test data set.

Here, we evaluate both NPE-MMD and NPE-DANN in an *amortized* setting on $N_{\text{obs}} = 1\,000$ *unseen* observed (contaminated) test data sets. All methods are trained on a low budget of $N = 5\,000$ uncontamined training

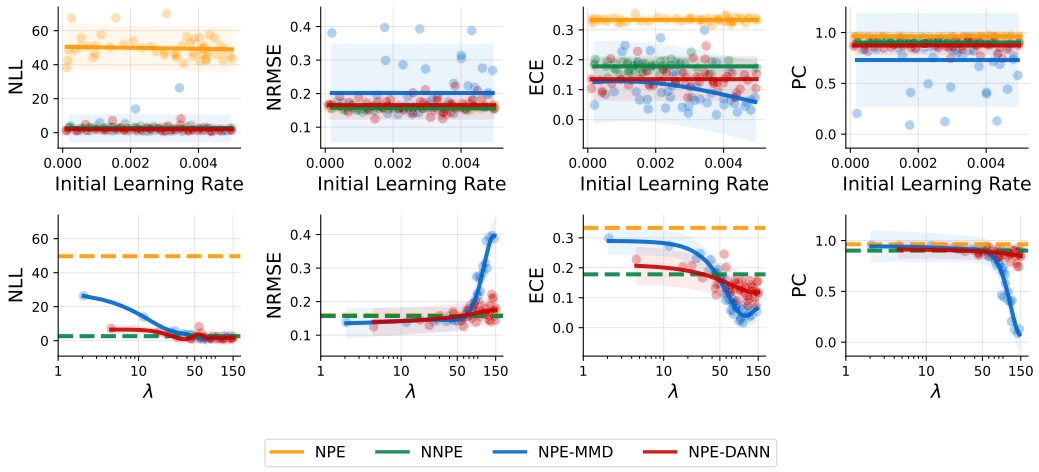


Figure 3: **Experiment 1**. Parameter space performance metrics resulting from 50 separate Bayesian hyperparameter optimization runs per method. The solid trend lines represent the predictive mean of a Gaussian process regression fitted to the individual run results, with the shaded areas representing 95% confidence intervals of the predictive distribution. If a parameter was not optimized, the methods average performance is depicted by a dashed horizontal line. Lower values indicate better performance for all metrics but PC. NLL = Negative Log Likelihood. NRMSE = Normalized Root Mean Squared Error. ECE = Expected Calibration Error. PC = Posterior Contraction. Whereas learning rate optimization is mostly ineffective for improving performance under contamination misspecification, the domain alignment regularization parameter λ controls a trade-off between error (NRMSE) vs. calibration (ECE) and contraction (PC) for NPE-UDA methods.

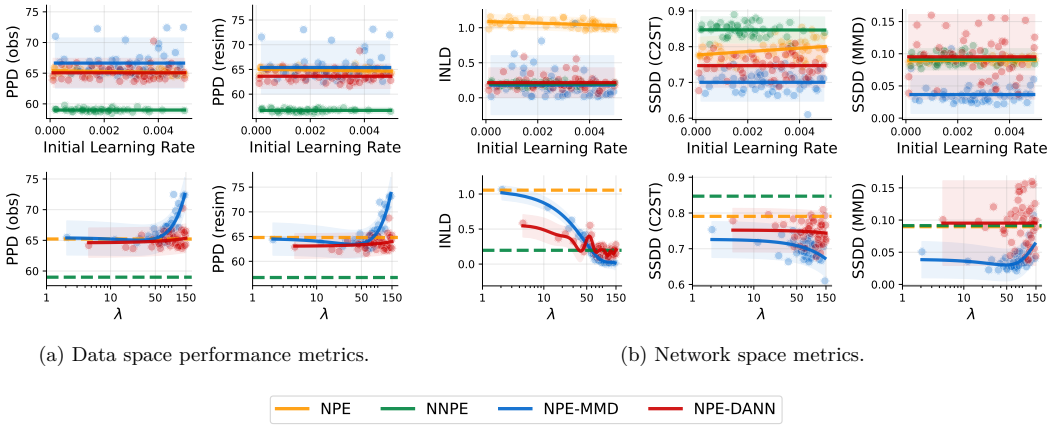
data sets. The NPE-UDA methods are trained with additional $N_{\text{obs}} = 1\,000$ unlabeled data sets from the observed domain, non-overlapping with the validation and test set.

To achieve a comprehensive evaluation of the trade-offs associated with the additional hyperparameters of the NPE-UDA methods, we optimize for the NLL on $N_{\text{obs}} = 1\,000$ contaminated validation data sets via Bayesian hyperparameter optimization (Akiba et al., 2019) using 50 separate training runs per method. The fixed training run budget per method automatically accounts for the complexity of the hyperparameter space, since less hyperparameters allow for a more thorough exploration of a method's hyperparameter space. We optimize the learning rate for all methods and the λ hyperparameter controlling the alignment strength for NPE-MMD and NPE-DANN (see Table B.1 for the search ranges for all hyperparameters).

For NPE-DANN, we additionally optimize (i) the width and depth of the feedforward discriminator architecture; (ii) the weight λ_{grl} balancing the strength of the adversarial summary network updates relative to the discriminator network (with $\lambda_{grl} < 1$ diminishing and $\lambda_{grl} > 1$ amplifying the reversed classification gradients passed to the summary network); and (iii) label smoothing, which has been found to be helpful for stabilizing the dynamics of domain-adversarial training (Zhang et al., 2023). [1]

Results The parameter space performance metrics results are displayed in Figure 3, the data space performance metrics results in Figure 4a, and the network space metrics in Figure 4b. Overall, optimizing the learning rate does not improve approximation performance on contaminated data, with the only exception

[1] We also explored scaling the gradient reversal weight λ_{grl} during training as suggested by Ganin et al. (2016), but found consistently worse results for the tested NPE setting. The same applies to the kernel choice in NPE-MMD, where the popular choice of using sums of Gaussian kernels with different bandwidths (Muandet et al., 2017; Schmitt et al., 2023) destabilized training compared to the sum of inverse multiquadratic kernels. Thus, we did not include these hyperparameters in the systematic hyperparameter assessment.



(a) Data space performance metrics.

(b) Network space metrics.

Figure 4: **Experiment 1**. Further metrics resulting from 50 separate Bayesian hyperparameter optimization runs per method. The solid trend lines represent the predictive mean of a Gaussian process regression fitted to the individual run results, with the shaded areas representing 95% confidence intervals of the predictive distribution. If a parameter was not optimized, the methods average performance is depicted by a dashed horizontal line. Lower values indicate better performance for all metrics but SSDD. PPD = Posterior Predictive Distance (RMSE). INLD = Inference Network Latent Distance (MMD). SSDD = Summary Space Domain Distance.

of NPE-MMD performance improving at higher learning rates. As expected, NNPE exhibits consistently robust performance as its data-generating process resembles the contamination misspecification scenario, whereas the success of the NPE-UDA methods depends on the domain alignment regularization parameter λ.

Concerning the parameter space performance metrics (Figure 3), we observe a lower NLL for all robust methods compared to NPE. However, taking into account the other metrics unveils that robust methods tend to improve calibration, but not estimation error. For the NPE-UDA methods, increasing the domain alignment regularization parameter λ improves calibration at the cost of approximation error and posterior contraction. We also find NPE-MMD to be much more sensitive to λ, with inference breaking down at very large values.

With regard to data space performance metrics (Figure 4a), both approaches of obtaining the posterior predictive distance lead to similar results in this setting. While the PPD mostly mirrors the NRMSE in the parameter space, we find a surprising advantage of NNPE in the data space. Since we find a close correspondence between PPD and NRMSE in the other experiments, NNPE's unique advantage in the current setting could be caused by a peculiar sensitivity of the Ricker simulator to certain parameter constellations that NNPE is less likely to infer.

Considering network space metrics (Figure 4b), we find that the deformation of the inference network's latent space, as measured by INLD, directly corresponds to density estimation quality as measured by NLL. This is unsurprising, given the change-of-variable mechanics of normalizing flows (Papamakarios et al., 2021). Further, we observe an overall lower summary space domain distance (SSDD) for both NPE-UDA methods in terms of C2ST but only for NPE-MMD in terms of MMD, and we do not observe the expected decrease in SSDD with increasing λ. We suspect these patterns to be caused by two factors limiting SSDD variability: First, the overall domain distance being relatively small and second, the hyperparameter optimization search concentrating on higher λ values. Our next experiments inspect the lower range of λ settings more closely and reveal it to be decisive for SSDD. Lastly, we find an overall little impact of the additional NPE-DANN hyperparameters (see Figure B1).

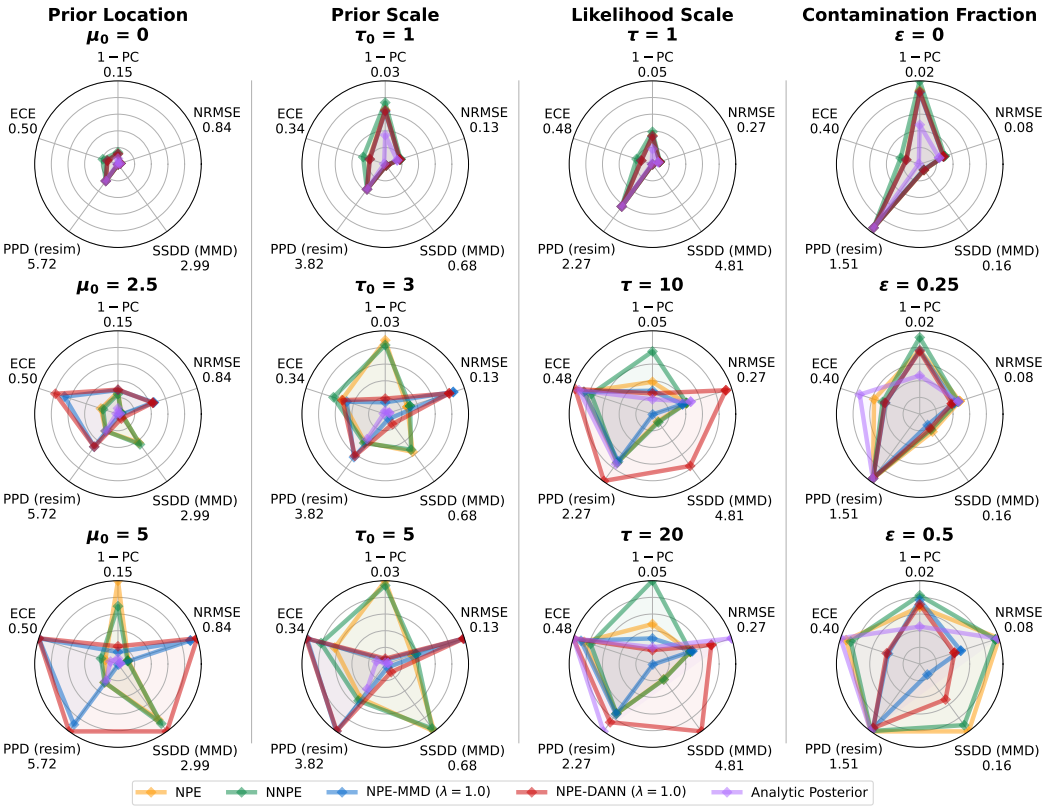


Figure 5: **Experiment 2**. Performance metrics and summary space domain distance (SSDD) of the methods in all misspecification scenarios (columns), aggregated via the median of 10 runs. The first row shows the well-specified setting, with misspecification increasing from top to bottom within each column. Metric values are centered at 0 and normalized by each column's/scenario's maximum value, which is displayed below the metric name at the border of each radar plot. Lower values indicate better performance for all metrics but SSDD. 1− PC = 1− Posterior Contraction. NRMSE = Normalized Root Mean Squared Error. SSDD (MMD) = Summary Space Domain Distance measured via MMD (not applicable for Analytic Posterior). PPD (resim) = Posterior Predictive Distance measured via the RMSE to resimulated data. ECE = Expected Calibration Error. NPE-UDA methods fail under prior misspecification but can be advantageous under contamination.

5.2 Experiment 2 - 2D Gaussian Means: Simple and Controllable Benchmark

Setup Inspired by Schmitt et al. (2023), our next experiment tests the performance of NPE-UDA methods against different *types* of misspecification. Here, the simple approximation task of inferring the means of a 2-dimensional Gaussian model enables a comparison to an analytic posterior, which represents the optimal solution under **Target 1**. The well-specified setting uses a multivariate standard normal prior and an identity likelihood covariance matrix. We evaluate performance under increasing misspecification in two prior misspecification scenarios – prior location $\boldsymbol{\mu_0}$ and prior scale $\boldsymbol{\Sigma_0} = \tau_0 \boldsymbol{I}_2$ – and two likelihood misspecification scenarios – likelihood scale $\boldsymbol{\Sigma} = \tau \boldsymbol{I}_2$ and data contamination ϵ (see Table B.2). For the contamination misspecification, a fraction ϵ of the observations is replaced by negative and positive vectors of the constant $c = 1.5$ to obtain atypical observations without affecting overall location or scale. Each simulated data set

contains $M = 100$ exchangeable observations. All methods train on $N = 48\,000$ well-specified data sets until convergence, with NPE-UDA methods additionally exposed to $N_{\text{obs}} = 48\,000$ unlabeled data sets. All methods are evaluated on $N_{\text{obs}} = 1\,000$ observed data sets (unseen by NPE-UDA methods).

Results Figure 5 displays the results for all misspecification scenarios. We invert the meaning of the posterior contraction (PC) metric, such that lower means better for all metrics. Thus, the performance of a method can mostly be inferred from its area. All methods perform well in the well-specified case (first row), whereas we observe distinct but consistent patterns for the different methods under increasing mismatch.

In the prior misspecification scenarios, NPE and NNPE perform well compared to the analytic posterior under a location shift, but fall off under a scale shift, with increasing error (NRMSE) and drastically increased miscalibration (ECE). The two NPE-UDA methods, on the other hand, perform poorly for both prior location and scale shift.

In the likelihood scale misspecification scenario, the NPE methods are generally less sensitive to the misspecification than the analytic posterior. NPE-MMD successfully aligns the summary space between domains, but the practical effects of this alignment are limited to slightly improved PC. NPE-DANN, on the other hand, shows the unstable training dynamics described in Section 3.3: It fails to align the summary space for both likelihood scale misspecification levels (as well as the $\mu_0 = 5$ prior location shift scenario), which translates to poor performance. Crucially, *this drastic failure in the observed domain is not detectable in the simulated domain*, where all methods, even NPE-DANN in the $\tau = 20$ scenario, perform well according to all metrics and only SSDD signals irregularities (see Figure B3).

In the data contamination scenario, deviating from the true posterior via **Target 2** enables NPE-MMD and NPE-DANN to excel, achieving much lower NRMSE and ECE than NPE, NNPE, *and even the analytic posterior*. Across all misspecification scenarios, NNPE performs similarly to NPE, with NNPE's noisier training process typically resulting in slightly lower contraction and calibration. This applies even to the contamination scenario, where we expected an advantage for NNPE due to its similarity to the corruption process.

With regard to differences between metrics, PPD reliably detects NPE-UDA failures under prior misspecification, but is less sensitive under likelihood scale shifts and insensitive under contamination. We suspect that this lessened informativeness compared to Experiment 1 is caused by the simple data structure of the Gaussian mean model. Further, the deformation of the inference network's latent space, measured via INLD, is again strongly associated with approximation quality, with rank correlations of $r = .79$ with NRMSE, $r = .95$ with ECE, and $r = .74$ with PPD (re-simulated).

Furthermore, in this low-dimensional example, we observed two sources of instability for NPE-UDA methods. First, despite the good performance across a wide range of λ settings in Experiment 1, we observed a high sensitivity to the λ regularization hyperparameter. For $\lambda = 0.1$, the domain alignment in the contamination scenario is too weak, eliminating the advantage of NPE-UDA methods (see Figure B5). Differently, setting $\lambda = 10$ leads to drastic failures due to overly aggressive domain alignment of NPE-MMD and increases the training instabilities of NPE-DANN (see Figure B6). Second, these extreme failures necessitated aggregation of the training run results via the median instead of the mean, since single extreme results (not for $\lambda = 0.1$, occasionally for $\lambda = 1$, frequently for $\lambda = 10$) rendered visualization of the results impossible.

5.3 Experiment 3 - Bayesian Denoising: High-Dimensional Benchmark

Setup This experiment tests the generalization of our results on a high-dimensional (in the context of SBI) benchmark simulating a noisy camera model (Ramesh et al., 2022). The parameter vector $\boldsymbol{\theta} \in \mathbb{R}^{256}$ represents a crisp image, whereas the observation $\boldsymbol{x} \in \mathbb{R}^{256}$ is a blurred version of the original image generated by the noisy camera. The training data set consists of $N = 50\,000$ images from the MNIST data set (Lecun et al., 1998), downscaled to 16×16 pixels for compatibility with the USPS data set (Hull, 1994).

We test four different misspecification scenarios (see Table 2 for examples). In the prior misspecification scenario, we keep the settings of the noisy camera model constant but use images from the USPS data set (Hull, 1994). While both data sets contain digits, the USPS data set features smaller margins, giving

Method	λ	Prior (MNIST → USPS) NRMSE ↓	PPD ↓	SSDD	Likelihood Scale NRMSE ↓	PPD ↓	SSDD	Contamination (Noise) NRMSE ↓	PPD ↓	SSDD	Contamination (Rows) NRMSE ↓	PPD ↓	SSDD
NPE	-	**0.259**	**0.131**	0.256	0.165	0.036	0.101	0.312	0.117	0.463	0.315	0.112	0.386
NNPE	-	0.274	0.137	0.206	0.173	0.041	0.088	**0.154**	**0.035**	0.019	0.214	0.064	0.071
NPE-DANN	0.01	0.329	0.193	0.027	**0.130**	0.031	0.027	0.217	0.065	0.017	0.180	0.056	0.016
NPE-DANN	0.10	0.326	0.193	0.029	0.134	0.031	0.016	0.211	0.057	0.015	**0.178**	**0.045**	0.014
NPE-DANN	1.00	0.352	0.205	0.038	0.147	0.032	0.014	0.240	0.068	0.015	0.201	0.051	0.014
NPE-MMD	0.01	0.303	0.184	0.029	0.145	**0.027**	0.022	0.268	0.085	0.016	0.262	0.085	0.016
NPE-MMD	0.10	0.312	0.189	0.018	0.189	0.059	0.009	0.245	0.071	0.013	0.181	0.047	0.012
NPE-MMD	1.00	0.374	0.225	0.004	0.322	0.129	0.000	0.321	0.129	0.000	0.322	0.129	0.000

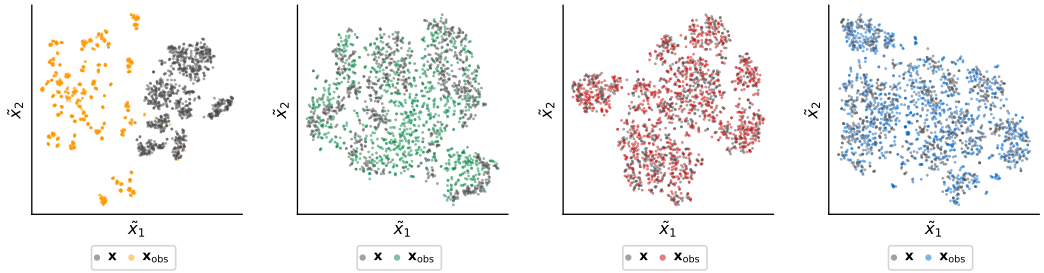
Table 1: **Experiment 3**. Metrics of the methods in all misspecification scenarios, averaged across 3 runs. NRMSE: Normalized Root Mean Squared Error (lower is better). PPD = Posterior Predictive Distance (RMSE) to resimulated data (lower is better). SSDD = Summary Space Domain Distance (MMD). Lower values indicate better summary space alignment, but too much alignment (i.e., vanishing SSDD) can lead to an uninformative summary space (e.g., NPE-MMD with $\lambda = 1.00$).

	Train	Prior (MNIST → USPS)			Likelihood Scale			Contamination (Noise)			Contamination (Rows)		
Parameters θ													
Observations $\mathbf{x}$	-												
NPE													
NNPE													
NPE-DANN													
NPE-MMD													

Table 2: **Experiment 3**. Parameters, observations, and samples from the run with the lowest NRMSE for each scenario and method. *Train* shows a sample from the parameters $\boldsymbol{\theta}$ of the training distribution and the corresponding observations $\boldsymbol{x}_{\text{obs}}$. The observations are identical for NPE, NPE-DANN, and NPE-MMD, whereas spike-and-slab noise is added for NNPE. The similarity to the observations in the *Contamination (Noise)* scenario explains the good performance of NNPE in that scenario.

the priors different support. In the likelihood scale scenario, we increase the amount of blur. In the noise contamination scenario, we replace 10% of the pixels with salt-and-pepper noise (i.e., set them to black or white). In the row contamination scenario, we randomly set two rows (12.5% of the pixels) of each observation to black. The NPE-UDA methods are trained with $N_{\text{obs}} = 1\,000$ observed training data sets. We evaluate the performance on further $N_{\text{obs}} = 1\,000$ observed test data sets (unseen by the NPE-UDA methods during training).

Results Table 1 displays an overview of the metrics in all scenarios. Table 2 shows samples from the best run (lowest NRMSE) for each scenario and method. We observe worse approximations for all robust methods compared to NPE in the prior misspecification scenario, even though the summary space domain distance (SSDD) is strongly diminished for NPE-DANN and NPE-MMD. This is somewhat expected, as performance improvements would also require an adaptation of the inference network, which cannot be induced by the methods tested here. NNPE is beneficial in the two contamination scenarios, whereas NPE-DANN and NPE-MMD improve performance in all three likelihood misspecification scenarios. The results highlight the



(a) NPE (b) NNPE (c) NPE-DANN (d) NPE-MMD

Figure 6: **Experiment 3** – Contamination (Rows): t-SNE representation of the summary spaces from the best run (lowest NRMSE) of each method. The t-SNE map is calculated jointly using summary statistics of data from both the simulated and the observed domain. For NPE, the two domains are clearly separated. With increasing alignment in the summary space, the overlap between the domains increases: The domain embeddings overlap partially for NNPE (medium SSDD in Table 1) and fully for NPE-DANN and NPE-MMD (low SSDD in Table 1). Since t-SNE can introduce artificial clustering, the overlap and not the specific shape is relevant.

differences between the robust methods: While NNPE mainly excels in the noise contamination scenario, where its misspecification model matches the domain shift, NPE-UDA methods effectively adapt to different likelihood shifts. Figure 6 shows a two-dimensional t-SNE (Van der Maaten & Hinton, 2008) representation of the summary spaces produced by the different methods. The observed overlap corresponds well to the SSDD values reported in Table 1.

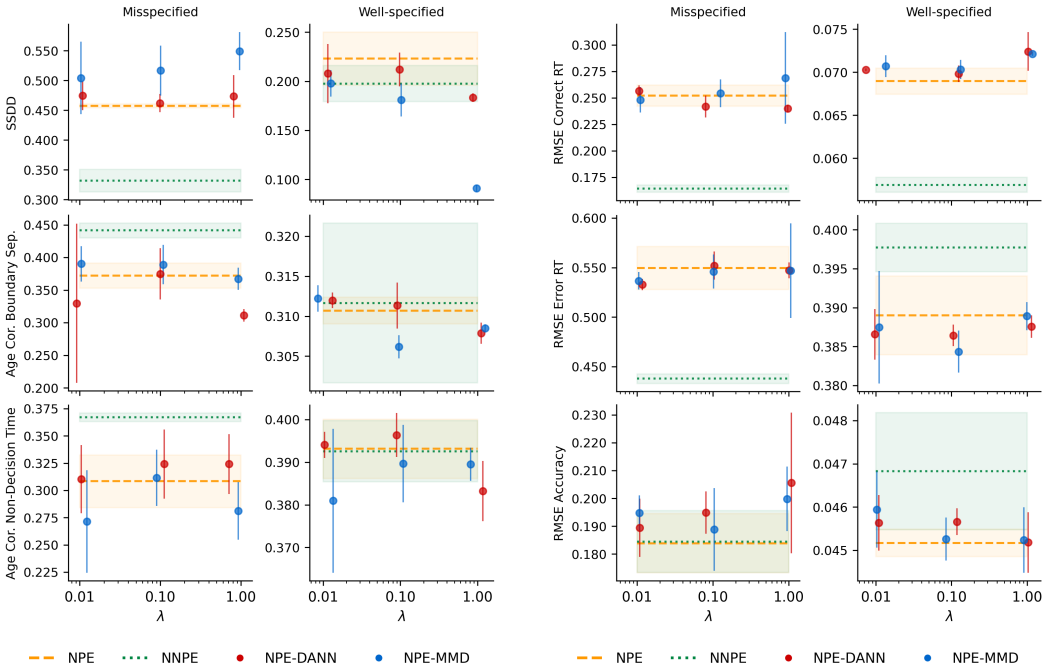
Overall, NPE-DANN achieves good performance over a wide range of λ values. In contrast, NPE-MMD is prone to overregularizing the summary space, leading to a complete loss of information in the summary space (see also Figure 2). This is indicated by a huge drop in performance and vanishing SSDD. We found NPE-MMD highly sensitive to the chosen batch size, which we had to increase from 32 to 128 to achieve acceptable results. Thus, increasing the batch size and reducing λ can counteract excessive regularization in higher-dimensional problems. Finally, the close correspondence between the NRMSE and PPD metrics confirms our hypothesis that the limited diagnostic power of PPD in the likelihood misspecification scenarios of Experiment 2 was caused by the limited informativeness of data simulated from a simple Gaussian model.

5.4 Experiment 4 - Decision Making: Large Human Behavior Data Set

Setup Finally, we evaluate the effectiveness of NPE-UDA methods using a real-world example from cognitive science, based on a large-scale empirical data set of a binary decision-making task. The data set comprises response time (RT) data from millions of participants who completed the Implicit Association Test (IAT), a widely used test in social psychology (Greenwald et al., 1998; 2003). The data are publicly available through Project Implicit (Xu et al., 2014).

To gain insights into the underlying cognitive processes, RT data are often analyzed using evidence accumulation models (Evans & Wagenmakers, 2020; Ratcliff et al., 2016). However, fitting these models within a Bayesian framework is computationally intensive, making them a prime use case for ABI. A further challenge with online RT data is the presence of substantial noise, stemming from limited experimental control (Gong & Huskey, 2023). For example, participants may guess, experience attentional lapses, or behave inconsistently. This issue can be framed as a form of likelihood misspecification, where the observed data are partially generated by processes not captured by the model.

Traditional approaches to address this include data cleaning procedures or extending the model with additional parameters (e.g., to capture guessing behavior, see Ratcliff & Tuerlinckx, 2002). Here, we explore whether NPE-UDA methods can offer an alternative by aligning the summary statistics of simulated (clean)



(a) Summary space domain distances (SSDDs; MMD) and external validity metrics for empirical data. The first row shows the SSDDs of simulated vs. empirical data. Rows two and three show (across-person) age correlations for posterior medians in two cognitive model parameters: boundary separation (in the congruent experimental condition) and non-decision times (for correct trials).

(b) Posterior predictive distances (PPDs; RMSE) to cleaned empirical response time data. The first row shows the PPD for response times (in seconds) on correct trials, averaged across posterior samples, participants, response time quantiles, and experimental conditions. The second row shows the PPDs for error response times, while the third row shows the PPDs for accuracy rates (in %).

Figure 7: **Experiment 4** – Metric results for all methods and different λ weights for the NPE-UDA methods. All runs display the averaged results across three runs per method, with across-run standard deviations shown as shaded areas (for NPE and NNPE) or error bars (for NPE-DANN and NPE-MMD). Please note that y-axis scales differ between the misspecified and the well-specified settings. λ = UDA regularization weight. The left column shows results for misspecified data sets, while the right column shows results for the (much larger) well-specified data set. While NNPE improves performance on misspecified data sets, the adaptation of NPE-UDA methods to the general observed domain does not cover the minority of misspecified data sets.

model data with those of the noisy empirical data. This alignment should allow for improved parameter estimation despite the noise. Thus, we aim for **Target 2**: approximating the posterior distributions of cognitive model parameters while implicitly filtering out the noise components in the empirical data. All methods are trained on $N = 32\,000$ simulated RT data sets to approximate the 6 parameters of an evidence accumulation model adapted to IAT task data. The NPE-UDA approaches additionally train on $N_{\text{obs}} = 32\,000$ unprocessed empirical data sets.

For evaluation, we construct several empirical test data sets using held-out observed data sets along two dimensions (see Appendix B.6 for details): (i) data are either unprocessed or cleaned using gold standard pre-processing procedures from the cognitive modeling literature, and (ii) data are categorized as either well-specified — where the cognitive model is expected to be valid — or misspecified, representing cases likely affected by processes not captured by the model (i.e., likelihood misspecification) based on their atypicality in

NPE summary space. The misspecified test set consists of data from $N_{\text{obs}} = 730$ participants, while the well-specified test set consists of $N_{\text{obs}} = 10\,000$ randomly selected participants for whose data the probabilistic model is classified as well-specified.

To evaluate the networks' performance, we first compare the methods by assessing the summary space domain distances between simulated and empirical data sets. Next, because certain cognitive parameters are known to covary with participant age (von Krause et al., 2020; Ratcliff et al., 2010; Theisen et al., 2021), we then examine how well each method captures this relationship by comparing age correlations for two key parameters across estimation approaches. Finally, we use the PPD to assess the methods based on their ability to predict unseen empirical data. Similar to before, we use a denoised version of the data as PPD reference, which we approximate for this real-world application via the intensive pre-processing procedures.

We train the NPE-UDA methods with λ weights of 0.01, 0.1, 1, and 10. As before, $\lambda = 10$ frequently caused training instabilities (particularly for the MMD approach) leading to extreme results. To retain readable visualizations, we leave out the $\lambda = 10$ failure setting in the following.

Results Figures 7a and 7b present the main results from our decision modeling experiment. Concerning Figure 7a, the NPE-UDA methods show the expected reduction in SSDD in the well-specified data sets, with domain alignment becoming more pronounced as the UDA weight parameter λ increases. However, this domain adaptation effect does not extend to the minority of misspecified test sets, where NNPE shows the smallest distance between domains in summary space. With respect to age correlations for the two cognitive parameters, NNPE leads to on average slightly higher correlations on misspecified data compared to the other approaches.

Figure 7b presents posterior retrodictive RMSE values between posterior resimulations and the cleaned test data; corresponding results for the uncleaned data are provided in the appendix (Figure B11). Consistent with the SSDD and correlation results, we do not observe clear advantages for any method on the well-specified data sets but an advantage of NNPE on the misspecified data sets for RT fits. All methods perform comparably concerning the accuracy fits. The NPE-UDA approaches perform similarly to NPE (cf. Figure 7b), with higher λ values leading to increased instability as indicated by RMSE variability across runs for misspecified data. Overall, while additive training noise via the NNPE method proved beneficial for misspecified data sets, the general domain adaptation induced by the NPE-UDA methods did not carry over from the majority of well-specified data sets to the minority of misspecified data sets.

6 Discussion

NPE-UDA Methods In this paper, we argued that introducing UDA to NPE methods shifts the inference target from the standard analytic posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x}_{\text{obs}})$ to a "denoised posterior" $p(\boldsymbol{\theta} \mid \widetilde{\boldsymbol{x}}_{\text{obs}})$ based on adjusted data $\widetilde{\boldsymbol{x}}_{\text{obs}}$. This shift implies potential robustness gains under likelihood misspecification, where implicitly ignoring unmodeled phenomena in the observed data can be desirable. However, it also reduces the amount of information available to counteract prior misspecification. We consistently found these patterns throughout our systematic evaluations for both the existing NPE-MMD and our new NPE-DANN method.

Our results suggest that while posterior accuracy is related to domain distance in the summary space, the relationship is not straightforward, pointing to more subtle effects of domain alignment. Compared to simpler methods, such as NNPE (Ward et al., 2022), the flexibility of NPE-UDA methods to automatically adapt to various types of likelihood misspecification comes at the cost of reduced transparency during inference. Thus, a closer examination of UDA mechanisms is necessary to determine the nature of domain adaptation and how exactly these adaptations affect the interpretation of the resulting posterior $p(\boldsymbol{\theta} \mid \widetilde{\boldsymbol{x}}_{\text{obs}})$. Employing interpretability methods or decoders to track summary space adaptations back to the data space seems a particularly promising avenue for future research.

Role of the Regularization Weight We confirmed the existence of an application-specific optimal amount of UDA regularization (Zellinger et al., 2021) in the NPE context. Both NPE-UDA methods exhibited substantial instabilities for higher λ values: While we observed the typical unstable training dynamics associated with adversarial training for NPE-DANN, NPE-MMD exhibited overly aggressive domain align-

ment, suppressing all information contained in the data. Notably, which λ values qualify as high varied substantially between the experiments: while a broad range of λ values was tolerable in Experiment 1, we found severe inference breakdowns already for $\lambda = 10$ in all other experiments.

Measuring Robustness in the Real World The critical influence of the λ hyperparameter directly leads to the next question: How can we measure robustness gains in the real world, where ground-truth parameter values are unavailable, and find an optimal value for λ? Posterior predictive measures that compare posterior resimulations to the observed data are usually the tool of choice. For measuring the success of robust methods, however, their standard application is flawed, since using the observed data directly as a reference implies that successfully ignoring noise is *punished* by increased posterior predictive distance to noisy outliers. Our evaluation metrics sought to account for this by constructing a "denoised" reference data set. Although this approach is useful for benchmarking robust methods on real-world data, creating application-specific reference data solely for hyperparameter tuning would negate the benefit of automatic domain adaptation. Thus, finding generally reliable measures for real-world settings remains an unsolved issue, embedded into the overarching open UDA problem of guiding hyperparameter optimization despite missing target labels (Zellinger et al., 2021; Musgrave et al., 2022).

Future Avenues Based on the results of our experiments, we believe two further pathways to be especially interesting for future research. First, we focused on UDA approaches that target the summary space to retain the modularity of summary/inference network NPE architectures. This directly enables extensions to different downstream tasks, such as amortized Bayesian model comparison (Radev et al., 2021; Elsemüller et al., 2023; Jeffrey & Wandelt, 2024). The latter reframes model comparison as a classification task and is thus situated in the most extensively studied UDA task setting. Second, in contrast to the non-amortized NPE-MMD approach by Huang et al. (2023), which specializes inference for a single observed data set, we evaluated amortized approximation performance on data sets *unseen* by the NPE-UDA methods. While we observed good performance with low amounts of observed training data, it would be valuable to systematically assess the minimum amount of data necessary for effective adaptation to the observed domain. Here, generalization gaps of NPE-UDA methods could be measured via the difference in performance on data sets seen and unseen during training.

Conclusion Our results reveal a rather complex story of NPE-UDA shaped by the interplay of factors such as problem dimensionality, misspecification type, and UDA model choice. They also emphasize the need for pairing stylized theoretical investigations with thorough empirical evaluation. In sum, UDA offers a straightforward way to incorporate real data into simulation-based training and shows promise in handling various types of likelihood misspecification. However, our systematic evaluation uncovered major obstacles and unanswered questions that hinder the direct application of NPE-UDA methods in critical settings.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

Lynton Ardizzone, Jakob Kruse, Sebastian Wirkert, Daniel Rahner, Eric W Pellegrini, Ralf S Klessen, Lena Maier-Hein, Carsten Rother, and Ullrich Köthe. Analyzing inverse problems with invertible neural networks. *arXiv preprint arXiv:1808.04730*, 2018.

Lynton Ardizzone, Jakob Kruse, Carsten Lüth, Niels Bracher, Carsten Rother, and Ullrich Köthe. Conditional invertible neural networks for diverse image-to-image translation. In *Pattern Recognition: 42nd DAGM German Conference, DAGM GCPR 2020, Tübingen, Germany, September 28–October 1, 2020, Proceedings 42*, pp. 373–387. Springer, 2021.

Grace Avecilla, Julie N Chuong, Fangfei Li, Gavin Sherlock, David Gresham, and Yoav Ram. Neural networks enable efficient and accurate simulation-based inference of evolutionary parameters from adaptation dynamics. *PLoS Biology*, 20(5):e3001633, 2022.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19, 2006.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010.

Yves Bernaerts, Michael Deistler, Pedro J Gonçalves, Jonas Beck, Marcel Stimberg, Federico Scala, Andreas S Tolias, Jakob Macke, Dmitry Kobak, and Philipp Berens. Combined statistical-mechanistic modeling links ion channel genes to physiology of cortical neuron types. *bioRxiv*, pp. 2023–03, 2023.

Michael Betancourt. Calibrating model-based inferences and decisions. *arXiv preprint arXiv:1803.08393*, 2018.

Sebastian Bischoff, Alana Darcher, Michael Deistler, Richard Gao, Franziska Gerken, Manuel Gloeckler, Lisa Haxel, Jaivardhan Kapoor, Janne K Lappalainen, Jakob H Macke, et al. A practical guide to statistical distances for evaluating generative models in science. *arXiv preprint arXiv:2403.12636*, 2024.

Pier Giovanni Bissiri, Chris C Holmes, and Stephen G Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):1103–1130, 2016.

Paul-Christian Bürkner, Maximilian Scholz, and Stefan T Radev. Some models are useful, but how do we know which ones? towards a unified bayesian model taxonomy. *Statistic Surveys*, 17:216–310, 2023.

Paul-Christian Bürkner, Marvin Schmitt, and Stefan T Radev. Simulations in statistical workflows. *arXiv preprint arXiv:2503.24011*, 2025.

Patrick Cannon, Daniel Ward, and Sebastian M Schmon. Investigating the impact of model misspecification in neural simulation-based inference. *arXiv preprint arXiv:2209.01845*, 2022.

Bob Carpenter, Andrew Gelman, Matthew D Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of statistical software*, 76(1), 2017.

Jeffrey Chan, Valerio Perrone, Jeffrey Spence, Paul Jenkins, Sara Mathieson, and Yun Song. A likelihood-free inference framework for population genetic data using exchangeable neural networks. *Advances in neural information processing systems*, 31, 2018.

Yanzhi Chen, Dinghuai Zhang, Michael U. Gutmann, Aaron Courville, and Zhanxing Zhu. Neural approximate sufficient statistics for implicit models. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=SRDuJssQud`.

A Ćiprijanović, Diana Kafkes, S Jenkins, K Downey, Gabriel N Perdue, Sandeep Madireddy, T Johnston, and Brian Nord. Domain adaptation techniques for improved cross-domain study of galaxy mergers. *arXiv preprint arXiv:2011.03591*, 2020.

Corinna Cortes and Mehryar Mohri. Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science*, 519:103–126, January 2014. ISSN 0304-3975. doi: 10.1016/j.tcs.2013.09.027.

Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

Lars Dingeldein, Pilar Cossio, and Roberto Covino. Simulation-based inference of single-molecule experiments, 2024. URL `https://arxiv.org/abs/2410.15896`.

Lasse Elsemüller, Martin Schnuerch, Paul-Christian Bürkner, and Stefan T Radev. A deep learning method for comparing bayesian hierarchical models. *arXiv preprint arXiv:2301.11873*, 2023.

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 (TMLR)*, 2024.

Nathan J Evans and Eric-Jan Wagenmakers. Evidence accumulation models: current limitations and future directions. *Quantitative Methods for Psychology*, 16(2):73–90, 2020.

David T. Frazier, Ryan Kelly, Christopher Drovandi, and David J. Warne. The statistical accuracy of neural posterior and likelihood estimation, 2024. URL https://arxiv.org/abs/2411.12068.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Richard Gao, Michael Deistler, and Jakob H Macke. Generalized bayesian inference for scientific simulators via amortized cost estimation. *Advances in Neural Information Processing Systems*, 36:80191–80219, 2023.

Samuel Gershman and Noah Goodman. Amortized inference in probabilistic reasoning. In *Proceedings of the annual meeting of the cognitive science society*, volume 36, 2014.

Manuel Glöckler, Michael Deistler, and Jakob H Macke. Variational methods for simulation-based inference. *arXiv preprint arXiv:2203.04176*, 2022.

Manuel Glöckler, Michael Deistler, and Jakob H Macke. Adversarial robustness of amortized bayesian inference. *arXiv preprint arXiv:2305.14984*, 2023.

Manuel Gloeckler, Michael Deistler, Christian Weilbach, Frank Wood, and Jakob H Macke. All-in-one simulation-based inference. *arXiv preprint arXiv:2404.09636*, 2024.

Xuanjun Gong and Richard Huskey. Moving behavioral experimentation online: A tutorial and some recommendations for drift diffusion modeling. *American Behavioral Scientist*, pp. 00027642231207073, 2023. ISSN 0002-7642. doi: 10.1177/00027642231207073. URL https://doi.org/10.1177/00027642231207073. Publisher: SAGE Publications Inc.

Anthony G. Greenwald, Debbie E. McGhee, and Jordan L. K. Schwartz. Measuring individual differences in implicit cognition: The implicit association test. *Journal of Personality and Social Psychology*, 74(6): 1464–1480, 1998. ISSN 1939-1315, 0022-3514. doi: 10.1037/0022-3514.74.6.1464. URL http://doi.apa.org/getdoi.cfm?doi=10.1037/0022-3514.74.6.1464.

Anthony G. Greenwald, Brian A. Nosek, and Mahzarin R. Banaji. Understanding and using the implicit association test: I. an improved scoring algorithm. *Journal of Personality and Social Psychology*, 85(2): 197–216, 2003. ISSN 1939-1315, 0022-3514. doi: 10.1037/0022-3514.85.2.197. URL http://doi.apa.org/getdoi.cfm?doi=10.1037/0022-3514.85.2.197.

A Gretton, K. Borgwardt, Malte Rasch, Bernhard Schölkopf, and AJ Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13:723–773, 03 2012.

Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free mcmc with amortized approximate ratio estimators. In *International conference on machine learning*, pp. 4239–4248. PMLR, 2020.

Daolang Huang, Ayush Bharti, Amauri Souza, Luigi Acerbi, and Samuel Kaski. Learning robust statistics for simulation-based inference under model misspecification. *Advances in Neural Information Processing Systems*, 36, 2023.

Peter J Huber. *Robust statistics*. John Wiley & Sons, 1981.

J.J. Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994. doi: 10.1109/34.291440.

Niall Jeffrey and Benjamin D Wandelt. Evidence networks: simple losses for fast, amortized, neural bayesian model comparison. *Machine Learning: Science and Technology*, 5(1):015008, 2024.

Fredrik D Johansson, David Sontag, and Rajesh Ranganath. Support and invertibility in domain-invariant representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 527–536. PMLR, 2019.

Ryan P Kelly, David J Nott, David T Frazier, David J Warne, and Chris Drovandi. Misspecification-robust sequential neural likelihood. *arXiv preprint arXiv:2301.13368*, 2023.

Ryan P Kelly, David J Warne, David T Frazier, David J Nott, Michael U Gutmann, and Christopher Drovandi. Simulation-based bayesian inference under model misspecification. *arXiv preprint arXiv:2503.12315*, 2025.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Karl Christoph Klauer, Andreas Voss, Florian Schmitz, and Sarah Teige-Mocigemba. Process components of the implicit association test: a diffusion-model analysis. *Journal of Personality and Social Psychology*, 93(3):353, 2007.

Xianghao Kong, Wentao Jiang, Jinrang Jia, Yifeng Shi, Runsheng Xu, and Si Liu. Dusa: Decoupled unsupervised sim2real adaptation for vehicle-to-everything collaborative perception. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 1943–1954, 2023.

Peter D Kvam, Louis H Irving, Konstantina Sokratous, and Colin Tucker Smith. Improving the reliability and validity of the iat with a dynamic model driven by similarity. *Behavior Research Methods*, 56(3): 2158–2193, 2024.

Tuan Anh Le, Atilim Gunes Baydin, and Frank Wood. Inference compilation and universal probabilistic programming. In *Artificial Intelligence and Statistics*, pp. 1338–1348. PMLR, 2017.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=PqvMRDCJT9t`.

Xiaofeng Liu, Chaehwa Yoo, Fangxu Xing, Hyejin Oh, Georges El Fakhri, Je-Won Kang, Jonghye Woo, et al. Deep unsupervised domain adaptation: A review of recent advances and perspectives. *APSIPA Transactions on Signal and Information Processing*, 11(1), 2022.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pp. 97–105. PMLR, 2015.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*, 2009.

Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, Bernhard Schölkopf, et al. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10 (1-2):1–141, 2017.

Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. Unsupervised domain adaptation: A reality check. *arXiv preprint arXiv:2111.15672*, 2021.

Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. Three new validators and a large-scale benchmark ranking for unsupervised domain adaptation. *arXiv preprint arXiv:2208.07360*, 2022.

Costantino Pacilio, Swetha Bhagwat, and Roberto Cotesta. Simulation-based inference of black hole ringdowns in the time domain. *Physical Review D*, 110(8):083010, 2024.

Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE transactions on neural networks*, 22(2):199–210, 2010.

George Papamakarios and Iain Murray. Fast ε-free inference of simulation models with bayesian conditional density estimation. *Advances in neural information processing systems*, 29, 2016.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *The Journal of Machine Learning Research*, 22(1):2617–2680, 2021.

Christian S Perone, Pedro Ballester, Rodrigo C Barros, and Julien Cohen-Adad. Unsupervised domain adaptation for medical imaging segmentation with self-ensembling. *NeuroImage*, 194:1–11, 2019.

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*, 2020.

Stefan T Radev, Marco D'Alessandro, Ulf K Mertens, Andreas Voss, Ullrich Köthe, and Paul-Christian Bürkner. Amortized bayesian model comparison with evidential deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

Stefan T Radev, Marvin Schmitt, Lukas Schumacher, Lasse Elsemüller, Valentin Pratz, Yannik Schälte, Ullrich Köthe, and Paul-Christian Bürkner. Bayesflow: Amortized bayesian workflows with neural networks. *arXiv preprint arXiv:2306.16015*, 2023.

Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024.

Poornima Ramesh, Jan-Matthis Lueckmann, Jan Boelts, Álvaro Tejero-Cantero, David S. Greenberg, Pedro J. Goncalves, and Jakob H. Macke. GATSBI: Generative adversarial training for simulation-based inference. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=kR1hC6j48Tp`.

Roger Ratcliff and Francis Tuerlinckx. Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, 9(3):438–481, 2002. ISSN 1531-5320. doi: 10.3758/BF03196302. URL `https://doi.org/10.3758/BF03196302`.

Roger Ratcliff, Anjali Thapar, and Gail McKoon. Individual differences, aging, and IQ in two-choice tasks. *Cognitive psychology*, 60(3):127–157, 2010. ISSN 1095-5623. doi: 10.1016/j.cogpsych.2009.09.001.

Roger Ratcliff, Philip L. Smith, Scott D. Brown, and Gail McKoon. Diffusion decision model: Current issues and history. *Trends in cognitive sciences*, 20(4):260–281, 2016. doi: 10.1016/j.tics.2016.01.007.

Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. A survey on domain adaptation theory: Learning bounds and theoretical guarantees, July 2022.

William Edwin Ricker. Stock and recruitment. *Journal of the Fisheries Board of Canada*, 11(5):559–623, 1954.

Daniel Ritchie, Paul Horsfall, and Noah D Goodman. Deep amortized inference for probabilistic programs. *arXiv preprint arXiv:1610.05735*, 2016.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pp. 234–241. Springer, 2015.

Neil Savage. Synthetic data could be better than real data. *Nature*, 2023.

Marvin Schmitt, Paul-Christian Bürkner, and Köthe. Detecting model misspecification in amortized bayesian inference with neural networks. *Proceedings of the German Conference on Pattern Recognition (GCPR)*, 2023.

Ozan Sener, Hyun Oh Song, Ashutosh Saxena, and Silvio Savarese. Learning transferrable representations for unsupervised domain adaptation. *Advances in neural information processing systems*, 29, 2016.

Ali Siahkoohi, Gabrio Rizzuti, Rafael Orozco, and Felix J Herrmann. Reliable amortized variational inference with physics-based latent distribution correction. *Geophysics*, 88(3):R297–R322, 2023.

Yu Sun, Eric Tzeng, Trevor Darrell, and Alexei A Efros. Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*, 2019.

Paxson Swierc, Megan Zhao, Aleksandra Ćiprijanović, and Brian Nord. Domain adaptation for measurements of strong gravitational lenses. *arXiv preprint arXiv:2311.17238*, 2023.

Paxson Swierc, Marcos Tamargo-Arizmendi, Aleksandra Ćiprijanović, and Brian D Nord. Domain-adaptive neural posterior estimation for strong gravitational lens analysis. *arXiv preprint arXiv:2410.16347*, 2024.

Maximilian Theisen, Veronika Lerche, Mischa von Krause, and Andreas Voss. Age differences in diffusion model parameters: A meta-analysis. *Psychological Research*, 85:2012–2021, 2021. Publisher: Springer.

Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

Rheeya Uppaal, Yixuan Li, and Junjie Hu. How useful is continued pre-training for generative unsupervised domain adaptation? In *Proceedings of the 9th Workshop on Representation Learning for NLP (RepL4NLP-2024)*, pp. 99–117, 2024.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, Tony Yu, and the scikit-image contributors. scikit-image: image processing in Python. *PeerJ*, 2:e453, 6 2014. ISSN 2167-8359. doi: 10.7717/peerj.453. URL `https://doi.org/10.7717/peerj.453`.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Mischa von Krause and Stefan Radev. A big data analysis of the associations between cognitive parameters and socioeconomic outcomes. *OSF preprint https://doi.org/10.31219/osf.io/ge83u*, 2025.

Mischa von Krause, Veronika Lerche, Anna-Lena Schubert, and Andreas Voss. Do non-decision times mediate the association between age and intelligence across different content and process domains? *Journal of Intelligence*, 8(3):33, 2020. ISSN 2079-3200. doi: 10.3390/jintelligence8030033. URL `https://www.mdpi.com/2079-3200/8/3/33`.

Mischa von Krause, Stefan T. Radev, and Andreas Voss. Mental speed is high until age 60 as revealed by analysis of over a million participants. *Nature human behaviour*, 6(5):700–708, 2022. Publisher: Nature Publishing Group UK London.

Xiaoyu Wang, Ryan P. Kelly, David J Warne, and Christopher Drovandi. Preconditioned neural posterior estimation for likelihood-free inference. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=vgIBAOkIhY.

Daniel Ward, Patrick Cannon, Mark Beaumont, Matteo Fasiolo, and Sebastian Schmon. Robust neural posterior estimation and statistical model criticism. *Advances in Neural Information Processing Systems*, 35:33845–33859, 2022.

Antoine Wehenkel, Juan L Gamella, Ozan Sener, Jens Behrmann, Guillermo Sapiro, Marco Cuturi, and Jörn-Henrik Jacobsen. Addressing misspecification in simulation-based inference through data-driven calibration. *arXiv preprint arXiv:2405.08719*, 2024.

Jonas Bernhard Wildberger, Maximilian Dax, Simon Buchholz, Stephen R Green, Jakob H. Macke, and Bernhard Schölkopf. Flow matching for scalable simulation-based inference. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=D2cS6SoYlP.

Simon N Wood. Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466(7310): 1102–1104, 2010.

Kaiyuan Xu, Brian Nosek, and Anthony Greenwald. Psychology data from the race implicit association test on the project implicit demo website. *Journal of Open Psychology Data*, 2(1):e3, 2014. ISSN 2050-9863. doi: 10.5334/jopd.ac. URL http://openpsychologydata.metajnl.com/articles/10.5334/jopd.ac/.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. *Advances in neural information processing systems*, 30, 2017.

Andrew Zammit-Mangion, Matthew Sainsbury-Dale, and Raphaël Huser. Neural methods for amortized inference. *Annual Review of Statistics and Its Application*, 12, 2024.

Werner Zellinger, Natalia Shepeleva, Marius-Constantin Dinu, Hamid Eghbal-zadeh, Hoan Duc Nguyen, Bernhard Nessler, Sergei Pereverzyev, and Bernhard A Moser. The balancing principle for parameter choice in distance-regularized domain adaptation. *Advances in Neural Information Processing Systems*, 34:20798–20811, 2021.

Yi Zhang and Lars Mikelsons. Solving stochastic inverse problems with stochastic bayesflow. In *2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 966–972. IEEE, 2023.

YiFan Zhang, Xue Wang, Jian Liang, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. Free lunch for domain adversarial training: Environment label smoothing. *arXiv preprint arXiv:2302.00194*, 2023.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415, 2022.

APPENDIX

A Theoretical Details

Defining Amortized Bayesian Inference The term "amortized" has been used inconsistently throughout the literature, often denoting different generalization scopes. To clarify this concept for the discussion within this work, we offer the following definition:

Definition 1. *Let $\mathcal{A}$ denote a learner, $\boldsymbol{y}$ denote target variables, $\boldsymbol{x}$ represent input data, and $\boldsymbol{c}$ denote context variables. A learner $\boldsymbol{y} \sim \mathcal{A}(\boldsymbol{x}, \boldsymbol{c})$ is an* amortized Bayesian approximator *of a target quantity $\boldsymbol{y}$ with respect to a joint distribution $p(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$ if it can directly approximate $p(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{c})$ for any $(\boldsymbol{x}, \boldsymbol{c}) \sim p(\boldsymbol{x}, \boldsymbol{c})$ without requiring further training or additional approximation algorithms.*

By this definition, sequential methods that necessitate further training for new data (Papamakarios & Murray, 2016; Glöckler et al., 2022) are not considered amortized. Similarly, neural likelihood estimation (NLE; Papamakarios & Murray, 2016) and neural ratio estimation (NRE) (Hermans et al., 2020) which depend on MCMC algorithms do not qualify as amortized. In contrast, recent transformer-based (Gloeckler et al., 2024) or context-aware methods (Elsemüller et al., 2024) clearly fall within the scope of amortized neural posterior estimation (NPE).

B Experimental Details

Since the analytic posterior is only obtainable in Experiment 2, we measure performance relative to the data-generating parameters $\boldsymbol{\theta}^*$ to enable a direct comparison between the experiments. For likelihood misspecification settings, $\boldsymbol{\theta}^*$ is closely related to the posterior $p(\boldsymbol{\theta} \mid \widetilde{\boldsymbol{x}}_{\text{obs}})$ based on adjusted (e.g., decontaminated) data $\widetilde{\boldsymbol{x}}_{\text{obs}}$ (**Target 2**). Thus, the NPE-UDA posterior approximations being closer to $\boldsymbol{\theta}^*$ than the analytic posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ in the contamination scenario of Experiment 2 indicates that NPE-UDA methods indeed focus **Target 2**.

In all experiments, we build upon the `BayesFlow` Python library for amortized Bayesian workflows using generative neural networks (Radev et al., 2023).

B.1 Method Details

NNPE We implemented NNPE following the original implementation of Ward et al. (2022) at `https://github.com/danielward27/rnpe`, who used a spike scale of $\sigma = 0.01$ and a slab scale of $\tau = 0.25$ for all experiments. To remain consistent with the original implementation of Ward et al. (2022), we applied NNPE to standardized data in all experiments (in Experiment 3, we applied equivalent scaling instead, see Appendix B.5). Whether spike (standard normal) or slab (standard Cauchy) noise is applied to a simulated data point is determined by sampling from a Bernoulli distribution with $p = 0.5$.

Sensitivities of NPE-UDA In both experiments, we found the typical UDA phenomenon of sensitivity to higher learning rates (Perone et al., 2019) in the form of unstable learning dynamics such as exploding gradients. We also found sensitivity to short training times, suggesting that finding a stable optimum for the two-component NPE-UDA loss in Eq. 3 requires more gradient updates than usual.

Computational Cost of NPE-UDA Since the NPE-UDA methods operate in the compressed summary space, the runtime increase during training is minimal compared to NPE. For example, despite the relatively large (32-dimensional) summary space in Experiment 3, NPE and NPE-MMD took 12s/epoch and NPE-DANN 13s/epoch during GPU training on a cluster.

B.2 Metrics

We compute multiple metrics that measure the performance based on the approximation performance of J data-generating parameters $\{\theta_j^*\}_{j=1}^J$ via S posterior samples (we forego the obs notation where possible for brevity here). Depending on the metric, results are averaged across the J parameters and/or N observed data sets.

B.2.1 Parameter Space Performance Metrics

Negative log likelihood (NLL):

$$\text{NLL} = -\frac{1}{N}\sum_{n=1}^{N} \log q(\boldsymbol{\theta}_n^* \mid \phi(\boldsymbol{x}_n)), \tag{8}$$

where we utilize the fact that normalizing flows allow us to easily compute approximate (log) densities.

Normalized root mean squared error (NRMSE):

$$\text{NRMSE} = \frac{1}{N}\sum_{n=1}^{N}\left[\frac{1}{J}\sum_{j=1}^{J}\frac{\sqrt{\frac{1}{S}\sum_{s=1}^{S}(\theta_{j,n}^* - \hat{\theta}_{j,n}^{(s)})^2}}{\max(\theta_j^*) - \min(\theta_j^*)}\right]. \tag{9}$$

Expected calibration error (ECE) via the fraction of ground-truth inliers for R linearly spaced α-confidence intervals in $[0.005, 0.995]$ (Ardizzone et al., 2018; Radev et al., 2020):

$$\text{ECE} = \frac{1}{J}\sum_{j=1}^{J}\text{median}_{r=1}^{R}\left(\left|\frac{1}{N}\sum_{n=1}^{N}\mathbb{I}\left\{Q_{\frac{1-\alpha_r}{2}}(\hat{\theta}_j^{(n)}) \leq \theta_j^* \leq Q_{1-\frac{1-\alpha_r}{2}}(\hat{\theta}_j^{(n)})\right\} - \alpha_r\right|\right), \tag{10}$$

where $\text{median}_{r=1}^{R}$ represents the median fraction of inliers across the $R = 20$ credible intervals and $Q_k(\hat{\theta}_j^{(n)})$ represents the k-th quantile of the posterior samples for the n-th data set. We estimate the ECE on all test data sets via the median calibration error of $R = 20$ linearly spaced credible intervals, averaged across J model parameters.

Posterior contraction (PC) relative to the prior distribution (Betancourt, 2018):

$$\text{PC} = \frac{1}{N}\sum_{n=1}^{N}\left[\frac{1}{J}\sum_{j=1}^{J}\left(1 - \frac{\text{Var}(\hat{\theta}_{j,n}^{(s)})}{\text{Var}(\theta_{j,n}^*)}\right)\right]. \tag{11}$$

B.2.2 Data Space Performance Metrics

Posterior predictive distance (PPD):

$$\text{PPD} = \frac{1}{N}\sum_{n=1}^{N}\left[\frac{1}{S}\sum_{s=1}^{S} d\left(\boldsymbol{x}_n, \hat{\boldsymbol{x}}_n^{(s)}\right)\right]. \tag{12}$$

where $\hat{\boldsymbol{x}}^{(s)}$ represents a re-simulation based on a posterior sample of all estimated parameters, $\hat{\boldsymbol{\theta}}^{(s)}$, and we use RMSE for $d(\cdot, \cdot)$. To keep computation times reasonable, we limit the number of re-simulations $\hat{\boldsymbol{x}}^{(s)}$ by using a random subset of all S posterior samples in the experiments with a large number of posterior samples (Experiment 1: 100 samples; Experiment 2: 100 samples). We investigate two different PPD variants: (i) The default calculation defined in Equation 12 with $\boldsymbol{x}_n$ as the reference data set, (ii) and a calculation where $\boldsymbol{x}_n$ is replaced with a "denoised" reference data set $\widetilde{\boldsymbol{x}}_n$ that is obtained by re-simulating from the ground-truth parameter in the synthetic experiments and intensive data pre-processing in the real-world experiment. One limitation of the PPD metric is that any parameter values that would break the simulator have to be excluded before re-simulating, which can reduce the sensitivity of the metric for detecting approximation failures.

B.2.3 Network Space Metrics

Summary space domain distance (SSDD): SSDD, which does not quantify approximation performance but the degree of summary space alignment, is measured with two variants. The first variant is based on the biased sample-based $\widehat{\text{MMD}}^2$ estimator (Gretton et al., 2012):

$$\text{SSDD}_{\text{MMD}} = \frac{1}{N} \sum_{n=1}^{N} \widehat{\text{MMD}}^2 \big[\{\phi(\boldsymbol{x}_n)\} \,||\, \{\phi(\boldsymbol{x}_n^{\text{obs}})\} \big], \tag{13}$$

where $\{\phi(\boldsymbol{x}_n)\}$ and $\{\phi(\boldsymbol{x}_n^{\text{obs}})\}$ are sets of summary statistics over which the expectations are approximated. The second variant $\text{SSDD}_{\text{C2ST}}$ is based on the classifier two-sample test (C2ST) and represents the accuracy of an MLP classifier trained to distinguish the sets of summary statistics $\{\phi(\boldsymbol{x}_n)\}$ and $\{\phi(\boldsymbol{x}_n^{\text{obs}})\}$ (Bischoff et al., 2024).

Inference network latent distance (INLD): Following (Siahkoohi et al., 2023), we consider distortions in the inference network's latent space a proxy for approximation quality, which is a direct consequence of maximum likelihood training. To measure general distortions (beyond location and scale), we again consider the biased sample-based $\widehat{\text{MMD}}^2$ estimator (Gretton et al., 2012):

$$\text{INLD}_{\text{MMD}} = \frac{1}{N} \sum_{n=1}^{N} \widehat{\text{MMD}}^2 \big[\{\boldsymbol{z}_n\} \,||\, \{f(\boldsymbol{\theta}_n; \boldsymbol{x}_n)\} \big], \tag{14}$$

where $f(\boldsymbol{\theta}_n; \boldsymbol{x}_n)$ denotes the forward direction of the conditional invertible network realizing the normalizing flow.

B.3 Experiment 1 - Ricker

Probabilistic Model We follow the model specification of Radev et al. (2020), including their prior specifications for the growth rate parameter r and the scaling parameter ρ (see Radev et al. (2020) for details). The only deviation from Radev et al. (2020) is the specification of the parameter σ governing the standard deviation of Gaussian noise, where we follow Huang et al. (2023) and fix $\sigma = 0.3$ to allow for an easy visual inspection of the resulting 2D posterior landscape during method development.

Network Architecture We use a LSTNet architecture as described in Zhang & Mikelsons (2023) for the summary network ϕ, compressing the input to 6-dimensional summary statistics. For the generative inference network q, we use an affine coupling flow architecture (Ardizzone et al., 2021; Kingma & Dhariwal, 2018) with 3 coupling layers. See Table B.1 for the optimized hyperparameters (regarding architectural as well as training choices) per method and their respective search range.

Table B.1: Hyperparameter search ranges for all methods.

Hyperparameter	**Range**	**Method(s)**
Initial learning rate (α)	1×10^{-4}–5×10^{-3}	NPE, NNPE, NPE-MMD, NPE-DANN
NPE-UDA alignment weight λ	0.01–150.0	NPE-MMD, NPE-DANN
Discriminator depth	2–4	NPE-DANN
Discriminator width	128–1024	NPE-DANN
Gradient reversal weight λ_{grl}	0.5–15.0	NPE-DANN
Label smoothing	0.0–0.3	NPE-DANN

Training and Evaluation Details We use an AdamW optimizer with an initial learning rate of set by the hyperparameter search algorithm and cosine decay. We further use a batch size of 32 and train for 20 epochs. For the evaluation, we generate $S = 5\,000$ posterior samples per method and test data set.

Additional Results Figure B1 contains all performance metrics for the additional NPE-DANN hyperparameters, showing an overall little effect of the additional hyperparameters. We therefore favor simple settings of these hyperparameters in the following experiments.

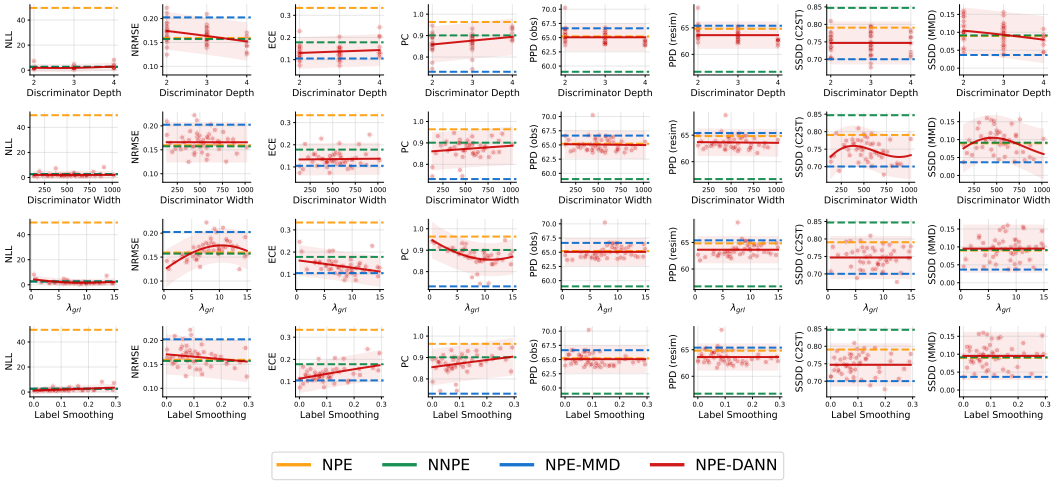


Figure B1: **Experiment 1**: Performance metrics for the additional NPE-DANN hyperparameters resulting from 50 separate Bayesian hyperparameter optimization runs per method. The solid trend lines represent the predictive mean of a Gaussian process regression fitted to the individual run results, with the shaded areas representing 95% confidence intervals of the predictive distribution. If a parameter was not optimized, the methods average performance is depicted by a dashed horizontal line. Lower values indicate better performance for all metrics but PC. NLL = Negative Log Likelihood. NRMSE = Normalized Root Mean Squared Error. ECE = Expected Calibration Error. PC = Posterior Contraction.

B.4 Experiment 2 - 2D Gaussian Means

Misspecification Setting	Prior	Likelihood
Well-specified	$\boldsymbol{\mu} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_2)$	$\boldsymbol{x}_k \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{I}_2)$
Prior location misspecification	$\boldsymbol{\mu} \sim \mathcal{N}(\boldsymbol{\mu_0}, \boldsymbol{I}_2)$	$\boldsymbol{x}_k \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{I}_2)$
Prior scale misspecification	$\boldsymbol{\mu} \sim \mathcal{N}(\mathbf{0}, \tau_0 \boldsymbol{I}_2)$	$\boldsymbol{x}_k \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{I}_2)$
Likelihood scale misspecification	$\boldsymbol{\mu} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_2)$	$\boldsymbol{x}_k \sim \mathcal{N}(\boldsymbol{\mu}, \tau \boldsymbol{I}_2)$
Contamination misspecification	$\boldsymbol{\mu} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_2)$	$\boldsymbol{x}_k \sim \frac{\epsilon}{2} \cdot \delta(\boldsymbol{x} - \boldsymbol{c}) + \frac{\epsilon}{2} \cdot \delta(\boldsymbol{x} + \boldsymbol{c}) + (1 - \epsilon) \cdot \mathcal{N}(\mu, \boldsymbol{I}_2)$

Table B.2: **Experiment 2**: Overview of the model specifications in the different misspecification settings.

Table B.2 provides an overview of the well-specified setting and the different misspecification scenarios inspired by Schmitt et al. (2023).

Network Architecture We use a deep set architecture (Zaheer et al., 2017) for the summary network ϕ, compressing the input to 4-dimensional summary statistics. For the generative inference network q, we use an affine coupling flow architecture (Ardizzone et al., 2021; Kingma & Dhariwal, 2018) with 3 coupling layers.

For the domain classifier ψ in NPE-DANN, we use a standard feedforward network with 2 hidden layers of width 256. We do not use label smoothing or weight the gradient reversal balance.

Training and Evaluation Details To rule out any overfitting effects on the results, we use an online training approach where new data from the simulated and the observed domain is simulated at each training step, resulting in overall simulation budgets of $N = 48\,000$ and $N_{\text{obs}} = 49\,000$. Since we use a batch size of 32, also for the observed data in NPE-UDA methods, online training amounts to 1 500 mini-batches and thus gradient updates. We use an Adam optimizer with an initial learning rate of $5 \cdot 10^{-4}$ and cosine decay. We generate $S = 5\,000$ posterior samples per method and test data set.

We provide additional results iterating over (i) performance in the simulated vs. the observed domain and (ii) $\lambda = [0.1, 1, 10]$.

Additional Results Figure B2, Figure B3, and Figure B4 show the performance in the simulated domain. Despite notable performance differences in the observed domain, all methods perform well in the simulated domain for the vast majority of settings, with the only exception being the failures of NPE-DANN for high regularization weights in Figure B4. NNPE performs worse in the simulated (noiseless) domain since it was optimized based on noisy training data. Besides the NPE-DANN failures, we mostly do not observe a trade-off of the summary space alignment of the NPE-UDA methods. Only in the high regularization setting $\lambda = 10$, the ECE is systematically higher compared to NPE.

Figure B5 and Figure B6 show the performance in the observed domain for varying λ settings. The results confirm our finding of an application- and also method-specific λ optimum: Whereas the difference of the NPE-UDA methods to NPE is often small for $\lambda = 0.1$, $\lambda = 10$ still leads to performance improvements of NPE-MMD in likelihood misspecification scenarios but renders NPE-DANN highly unstable when large domain shifts are present.

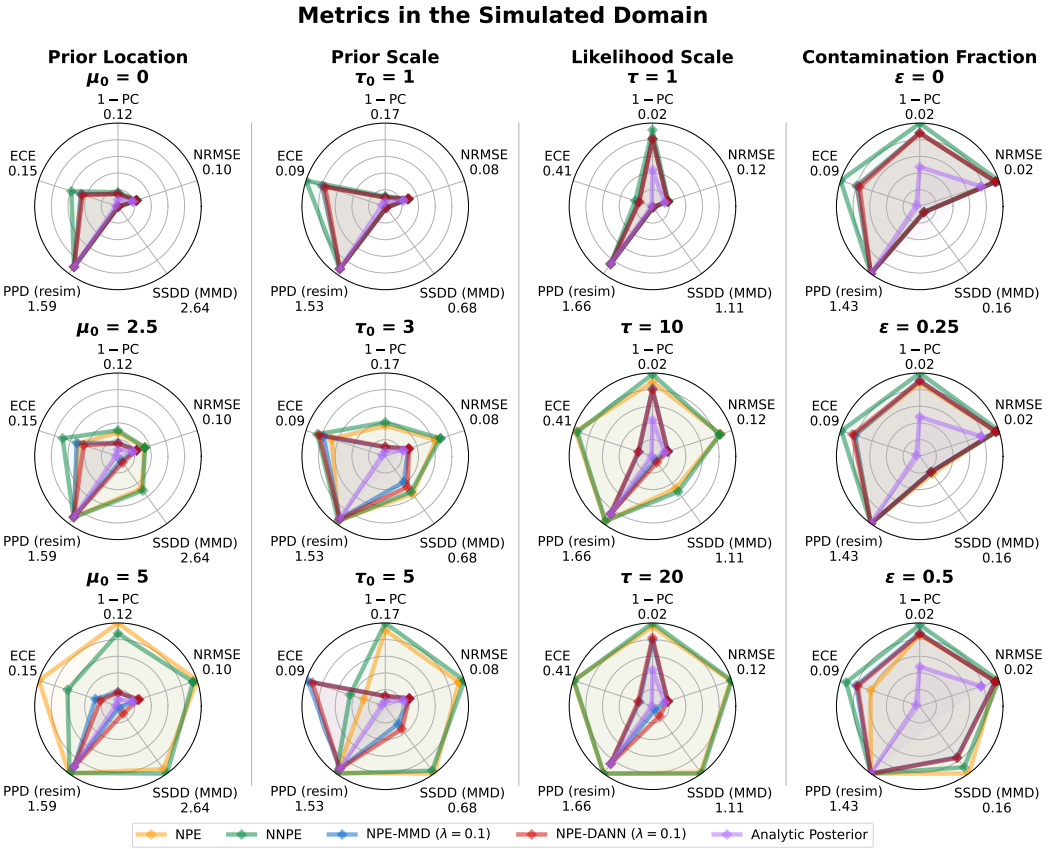


Figure B2: **Experiment 2**: Performance metrics and summary space domain distance (SSDD) of the methods in all misspecification scenarios (columns) **on simulated (i.e., well-specified) data for $\lambda = 0.1$ in NPE-MMD and NPE-DANN**, aggregated via the median of 10 runs. Lower values indicate better performance for all metrics but SSDD. 1− PC = 1− Posterior Contraction. NRMSE = Normalized Root Mean Squared Error. SSDD (MMD) = Summary Space Domain Distance measured via MMD (not applicable for Analytic Posterior). PPD (resim) = Posterior Predictive Distance measured via the RMSE to resimulated data. ECE = Expected Calibration Error.

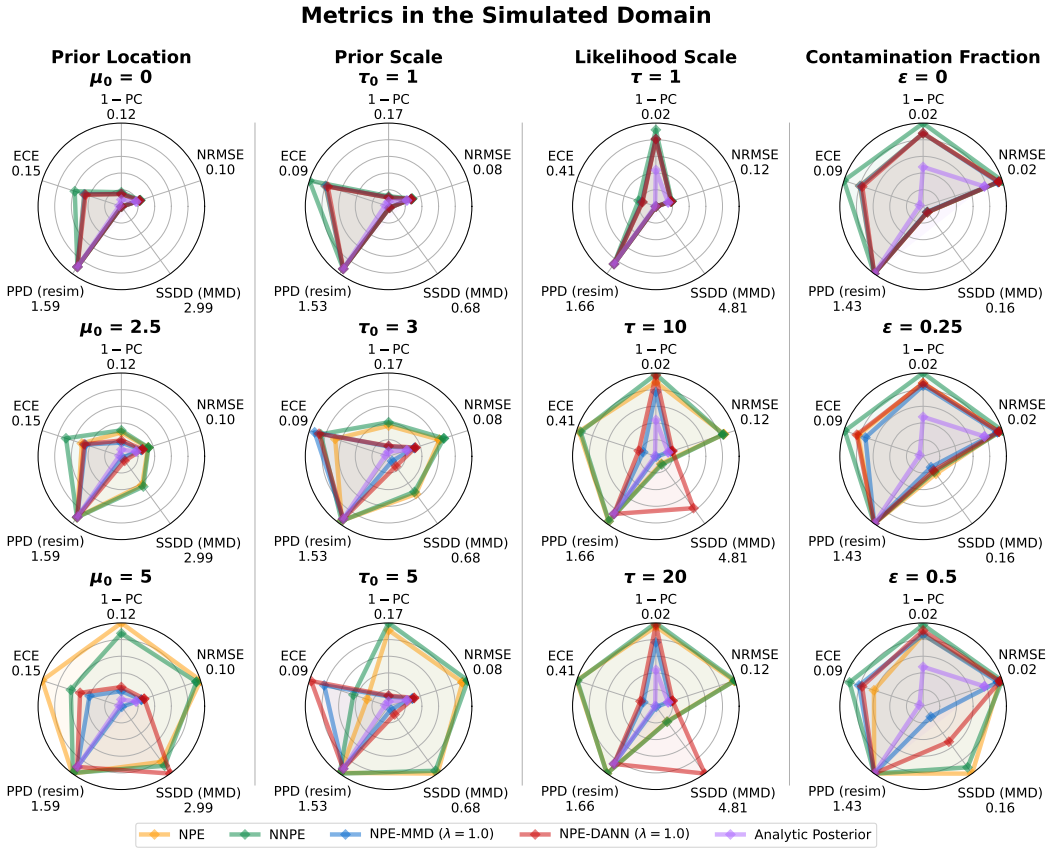


Figure B3: **Experiment 2**: Performance metrics and summary space domain distance (SSDD) of the methods in all misspecification scenarios (columns) **on simulated (i.e., well-specified) data for $\lambda = 1$ in NPE-MMD and NPE-DANN**, aggregated via the median of 10 runs. Lower values indicate better performance for all metrics but SSDD. 1− PC = 1− Posterior Contraction. NRMSE = Normalized Root Mean Squared Error. SSDD (MMD) = Summary Space Domain Distance measured via MMD (not applicable for Analytic Posterior). PPD (resim) = Posterior Predictive Distance measured via the RMSE to resimulated data. ECE = Expected Calibration Error.

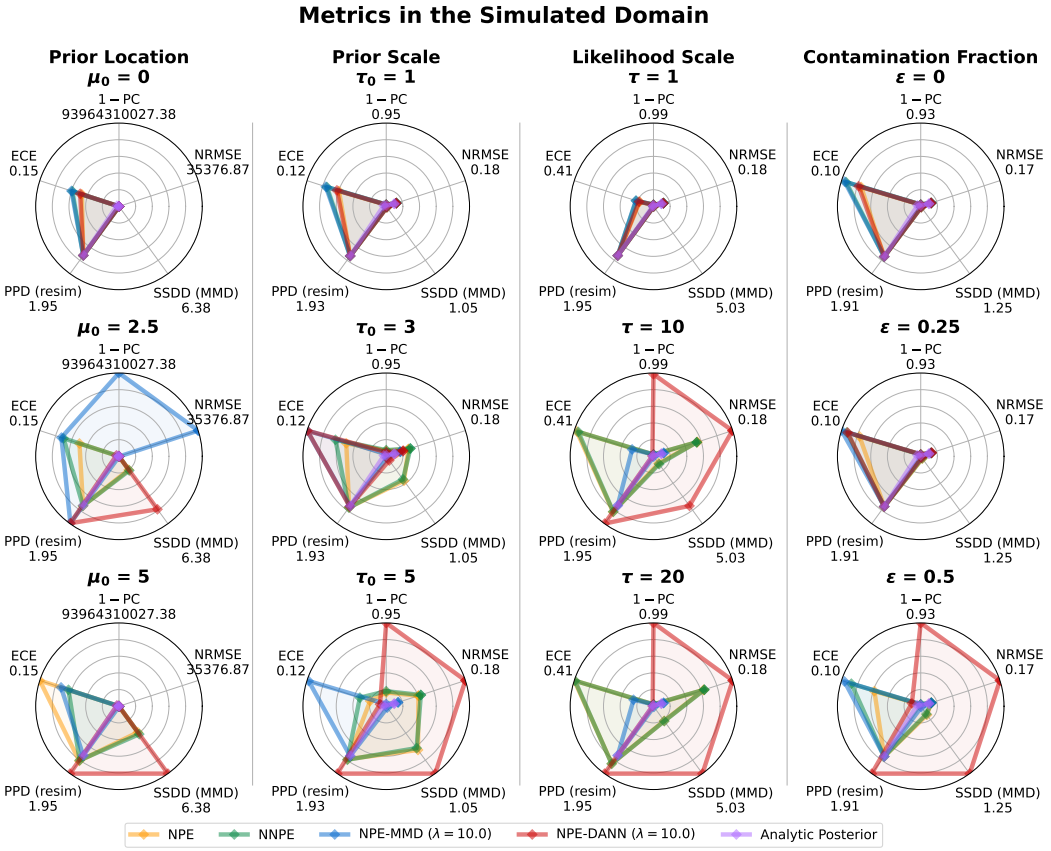


Figure B4: **Experiment 2**: Performance metrics and summary space domain distance (SSDD) of the methods in all misspecification scenarios (columns) **on simulated (i.e., well-specified) data for $\lambda = 10$ in NPE-MMD and NPE-DANN**, aggregated via the median of 10 runs. Lower values indicate better performance for all metrics but SSDD. 1− PC = 1− Posterior Contraction. NRMSE = Normalized Root Mean Squared Error. SSDD (MMD) = Summary Space Domain Distance measured via MMD (not applicable for Analytic Posterior). PPD (resim) = Posterior Predictive Distance measured via the RMSE to resimulated data. ECE = Expected Calibration Error.

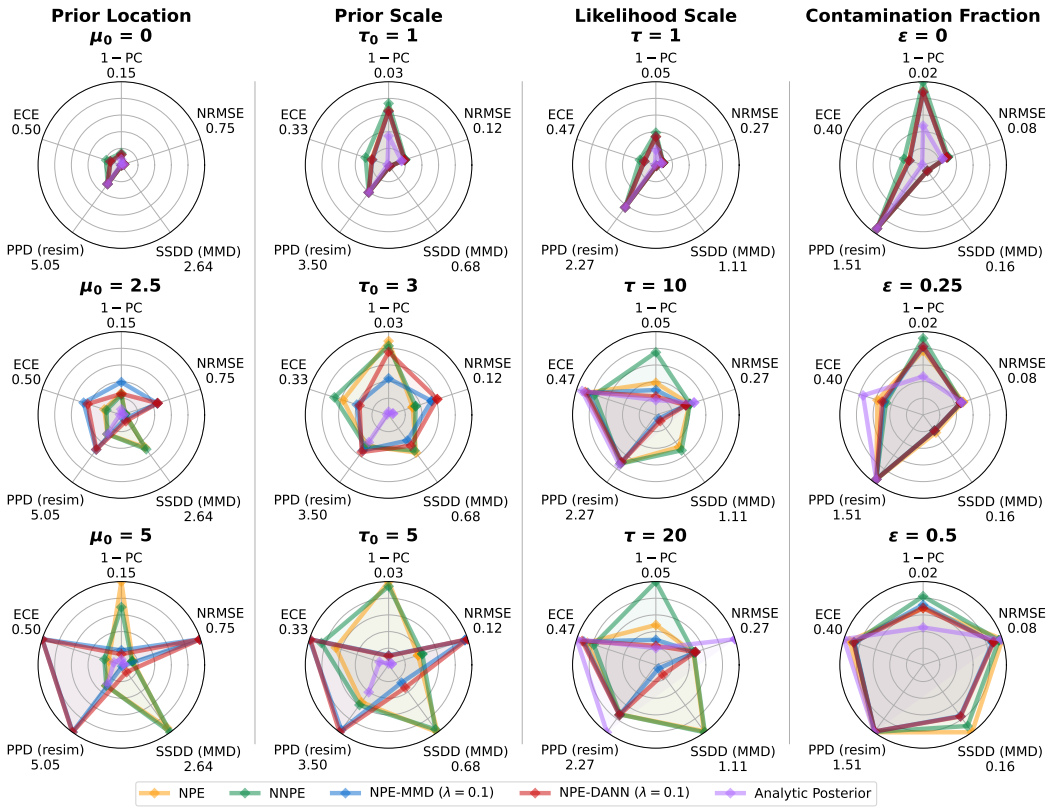


Figure B5: **Experiment 2**: Performance metrics and summary space domain distance (SSDD) of the methods in all misspecification scenarios (columns) **for $\lambda = 0.1$ in NPE-MMD and NPE-DANN**, aggregated via the median of 10 runs. Lower values indicate better performance for all metrics but SSDD. 1− PC = 1− Posterior Contraction. NRMSE = Normalized Root Mean Squared Error. SSDD (MMD) = Summary Space Domain Distance measured via MMD (not applicable for Analytic Posterior). PPD (resim) = Posterior Predictive Distance measured via the RMSE to resimulated data. ECE = Expected Calibration Error.

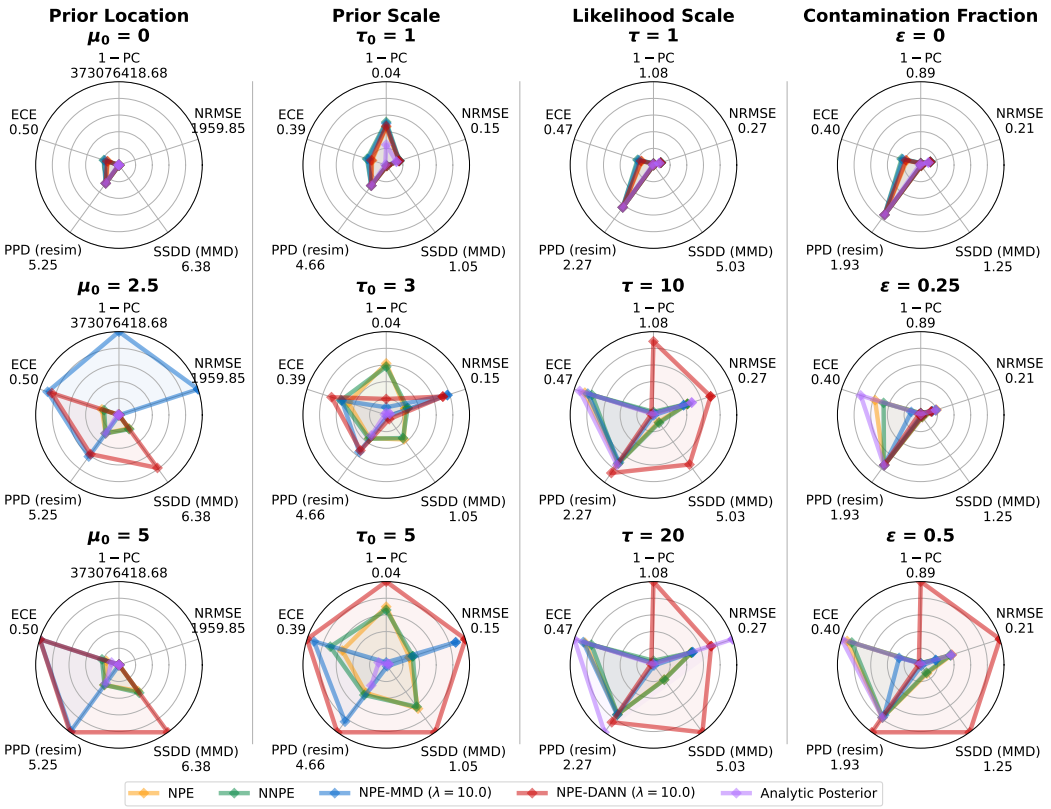


Figure B6: **Experiment 2**: Performance metrics and summary space domain distance (SSDD) of the methods in all misspecification scenarios (columns) **for $\lambda = 10$ in NPE-MMD and NPE-DANN**, aggregated via the median of 10 runs. Lower values indicate better performance for all metrics but SSDD. 1− PC = 1− Posterior Contraction. NRMSE = Normalized Root Mean Squared Error. SSDD (MMD) = Summary Space Domain Distance measured via MMD (not applicable for Analytic Posterior). PPD (resim) = Posterior Predictive Distance measured via the RMSE to resimulated data. ECE = Expected Calibration Error.

B.5 Experiment 3

Probabilistic Model We adopt a noisy camera model similar to the one presented in Ramesh et al. (2022). First, the input image is clipped to the range $[-1, 1]$. Next, we use scikit-image (van der Walt et al., 2014) to add Poisson noise to the image, then filter it using a Gaussian filter from SciPy (Virtanen et al., 2020) with a standard deviation σ for the Gaussian kernel. The result is a blurred image with identical size as the input image.

Data Preparation For each data set, we normalize the images to the range $[-1, 1]$. The MNIST (Lecun et al., 1998) images are rescaled from 28×28 to 16×16 with anti-aliasing enabled. To produce the training data $\boldsymbol{x}$, the images are processed by the simulator, with $\sigma_0 = 1.4$. We adapt NNPE, which is originally defined for standardized data, by scaling the noise scales by the standard deviation $\sigma_{\boldsymbol{x}}$ of the training data.

While the training data remains constant across scenarios, the observed data is generated in different ways. For the prior misspecification scenario, we use the USPS data set (Hull, 1994) instead of MNIST, but the parameters of the simulator remain identical (i.e., $\sigma = \sigma_0$). For the likelihood scale scenario, we use $\widetilde{\sigma} = 1.25 \cdot \sigma_0$, leading to an increased blur. For the noise contamination scenario, we randomly set 10% of the pixels of each observation to black or white. For the row contamination scenario, we randomly set 2 rows of each observation (i.e., 12.5% of the pixels) to black. Refer to Table 2 for samples from each scenario.

Network Architecture For the summary network, we use a 4-layer convolutional neural network, which outputs 32 learned summary variables.

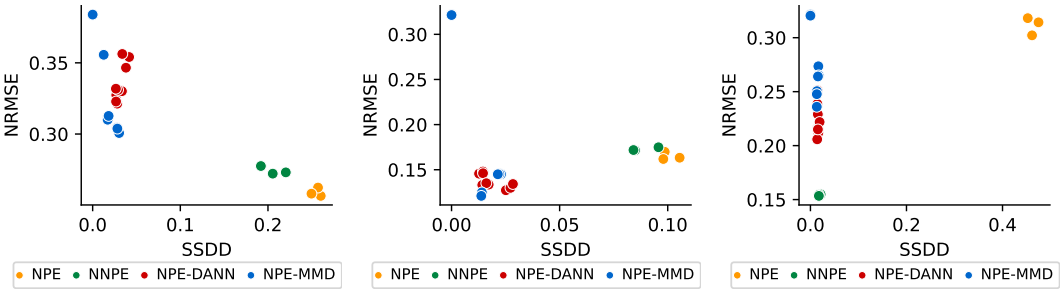
For the inference network, we use flow matching (Lipman et al., 2023; Wildberger et al., 2023) to convert a multivariate Gaussian distribution to the approximate posterior distribution. We use a U-Net architecture (Ronneberger et al., 2015) to learn the flow field conditional on the summary variables.

For NPE-DANN, we use a domain classifier ψ consisting of a standard feedforward network with 3 hidden layers of width 256, a gradient reversal layer (GRL) weight of 1, and no label smoothing.

Training and Evaluation Details We use an AdamW optimizer with an initial learning rate of $5 \cdot 10^{-4}$ and cosine decay. We use a batch size of 32 and train for 20 epochs, except for NPE-MMD, which required increasing the batch size to 128. To keep the number of gradient updates constant, we also increased the number of epochs to 80 for NPE-MMD. The training budget is 50 000 training images, and 1 000 observed images. Training one neural network takes approximately 10 minutes on a GPU.

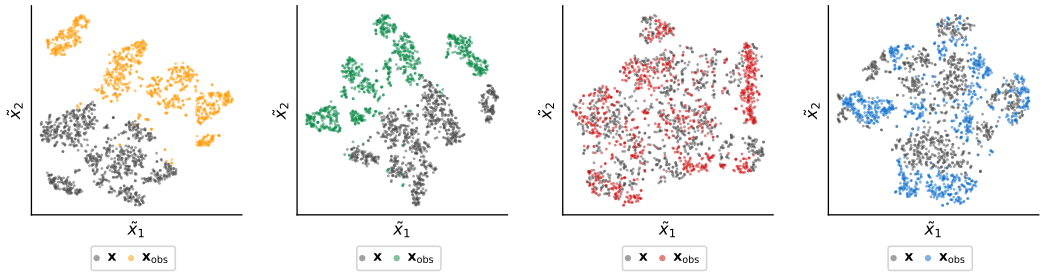
We use a moderate number of posterior samples of $S = 100$ per method and test data set to limit the computational cost of the experiment, allowing for a broader exploration of hyperparameters and the variance between multiple runs.

Additional Results Table B.3 displays the performance on a held-out in-distribution data set, to assess the influence on the loss on the in-domain observations. We consistently observe similar performance for NPE and the NPE-UDA methods for low λ and the expected performance loss of the noisy NNPE training on in-distribution data. Interestingly, NPE-MMD with low λ shows a slight but consistent improvement over NPE in this setting. The reason for this is unclear – the additional observed data might serve as an extra training signal that improves the learned summary statistics, even for in-distribution data. Additionally, we can verify that for NPE-MMD with $\lambda = 1.0$ with vanishing SSDD, the summary space does not contain information, as also the in-distribution performance drops to a value we expect of unconditional generation. Table B.4 displays the same data as Table 1, but with uncertainty indicators (standard deviation). Figure B7 shows the plots corresponding to Figure 2 for the remaining three scenarios.



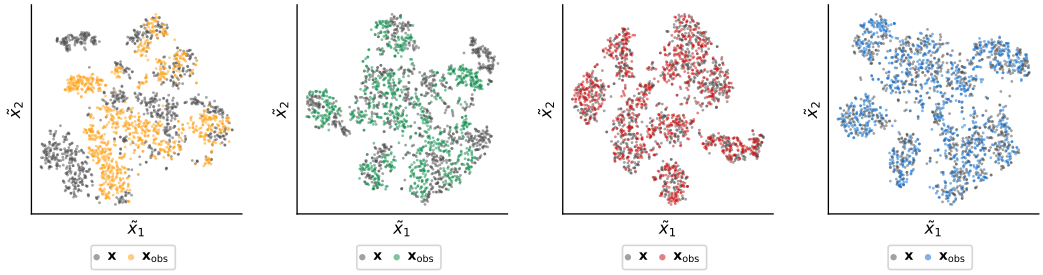
(a) Prior (MNIST → USPS) (b) Likelihood Scale (c) Contamination (Noise)

Figure B7: **Experiment 3**: Relationship of summary space domain distance (SSDD; MMD) and normalized root mean squared error (NRMSE, lower is better). For a) we see that despite the reduced SSDD, there is no gain in performance. For b) and c), we observe a sweet spot at a low SSDD value, before performance drops again when approaching zero. Refer to Table 1 for numerical values.



(a) NPE (b) NNPE (c) NPE-DANN (d) NPE-MMD

Figure B8: **Experiment 3** – Prior: t-SNE representation of the summary spaces from the best run (lowest NRMSE) of each method. Please refer to Figure 6 for a detailed caption.



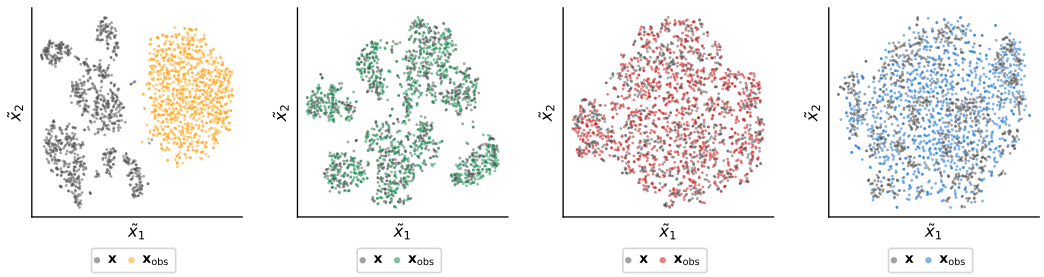
(a) NPE (b) NNPE (c) NPE-DANN (d) NPE-MMD

Figure B9: **Experiment 3** – Likelihood (Scale): t-SNE representation of the summary spaces from the best run (lowest NRMSE) of each method. Please refer to Figure 6 for a detailed caption.

Method	λ	Prior (MNIST → USPS) NRMSE ↓	Prior (MNIST → USPS) PPD ↓	Likelihood Scale NRMSE ↓	Likelihood Scale PPD ↓
NPE	-	0.104 (1.6e-03)	0.020 (3.0e-04)	0.103 (9.9e-04)	0.020 (2.0e-04)
NNPE	-	0.140 (8.9e-04)	0.030 (2.1e-04)	0.141 (1.7e-04)	0.031 (1.6e-04)
NPE-DANN	0.01	0.116 (2.0e-03)	0.024 (4.9e-04)	0.110 (8.2e-04)	0.023 (2.6e-04)
NPE-DANN	0.10	0.163 (2.0e-02)	0.037 (5.5e-03)	0.122 (1.3e-03)	0.026 (2.9e-04)
NPE-DANN	1.00	0.267 (7.6e-03)	0.080 (4.6e-03)	0.140 (9.3e-04)	0.030 (2.9e-04)
NPE-MMD	0.01	**0.092 (9.0e-04)**	0.019 (2.7e-04)	**0.091 (3.2e-04)**	**0.018 (2.4e-04)**
NPE-MMD	0.10	0.092 (1.8e-03)	**0.018 (2.7e-04)**	0.169 (1.1e-01)	0.055 (5.1e-02)
NPE-MMD	1.00	0.298 (3.2e-02)	0.111 (2.3e-02)	0.320 (4.2e-04)	0.127 (1.8e-04)

Method	λ	Contamination (Noise) NRMSE ↓	Contamination (Noise) PPD ↓	Contamination (Rows) NRMSE ↓	Contamination (Rows) PPD ↓
NPE	-	0.104 (1.8e-03)	0.021 (2.7e-04)	0.104 (1.3e-03)	0.020 (2.7e-04)
NNPE	-	0.141 (1.1e-03)	0.031 (1.7e-04)	0.139 (6.5e-04)	0.030 (1.6e-04)
NPE-DANN	0.01	0.114 (3.1e-04)	0.023 (7.6e-05)	0.115 (4.4e-03)	0.023 (7.2e-04)
NPE-DANN	0.10	0.139 (1.2e-03)	0.029 (4.8e-04)	0.136 (5.8e-03)	0.028 (2.0e-03)
NPE-DANN	1.00	0.192 (1.0e-02)	0.045 (3.6e-03)	0.179 (1.8e-02)	0.040 (6.2e-03)
NPE-MMD	0.01	**0.091 (1.2e-03)**	0.018 (2.3e-04)	**0.091 (5.9e-04)**	0.018 (1.1e-04)
NPE-MMD	0.10	0.092 (1.7e-03)	**0.018 (4.1e-04)**	0.093 (6.8e-04)	**0.018 (9.4e-05)**
NPE-MMD	1.00	0.319 (2.6e-04)	0.128 (2.5e-04)	0.320 (5.6e-04)	0.128 (2.2e-04)

Table B.3: **Experiment 3**: Overview of the metrics on a held-out validation data set from the training distribution (mean and standard deviation of three runs). NRMSE: Normalized Root Mean Squared Error (lower is better). PPD = Posterior Predictive Distance (RMSE) to resimulated data (lower is better) For NNPE and NPE-DANN we see reduced performance on the training distribution. For NPE-MMD, we see that for successful runs, the performance on the training distribution improves. For settings with vanishing SSDD (compare Table 1) the performance drops massively, for both training distribution and observed distribution. This supports the notion that no meaningful information is learned in the summary space.



(a) NPE (b) NNPE (c) NPE-DANN (d) NPE-MMD

Figure B10: **Experiment 3** – Contamination (Noise): t-SNE representation of the summary spaces from the best run (lowest NRMSE) of each method. Please refer to Figure 6 for a detailed caption.

Method	λ	Prior (MNIST → USPS) NRMSE ↓	PPD ↓	SSDD	Likelihood Scale NRMSE ↓	PPD ↓	SSDD
NPE	-	**0.259 (2.5e-03)**	**0.131 (1.7e-03)**	0.256 (4.4e-03)	0.165 (3.5e-03)	0.036 (1.0e-03)	0.101 (3.4e-03)
NNPE	-	0.274 (2.3e-03)	0.137 (1.6e-03)	0.206 (1.2e-02)	0.173 (1.5e-03)	0.041 (1.3e-03)	0.088 (5.3e-03)
NPE-DANN	0.01	0.329 (4.1e-03)	0.193 (2.6e-03)	0.027 (1.1e-03)	**0.130 (2.8e-03)**	0.031 (1.8e-03)	0.027 (1.4e-03)
NPE-DANN	0.10	0.326 (3.7e-03)	0.193 (1.4e-03)	0.029 (2.8e-03)	0.134 (8.5e-04)	0.031 (1.0e-03)	0.016 (1.2e-03)
NPE-DANN	1.00	0.352 (4.1e-03)	0.205 (4.3e-03)	0.038 (3.2e-03)	0.147 (1.0e-03)	0.032 (2.7e-04)	0.014 (8.6e-04)
NPE-MMD	0.01	0.303 (1.6e-03)	0.184 (7.4e-04)	0.029 (9.9e-04)	0.145 (8.3e-05)	**0.027 (2.9e-04)**	0.022 (6.1e-04)
NPE-MMD	0.10	0.312 (1.1e-03)	0.189 (5.7e-04)	0.018 (4.6e-04)	0.189 (9.4e-02)	0.059 (4.9e-02)	0.009 (6.5e-03)
NPE-MMD	1.00	0.374 (1.3e-02)	0.225 (1.1e-02)	0.004 (5.9e-03)	0.322 (2.1e-04)	0.129 (2.7e-04)	0.000 (2.4e-06)

Method	λ	Contamination (Noise) NRMSE ↓	PPD ↓	SSDD	Contamination (Rows) NRMSE ↓	PPD ↓	SSDD
NPE	-	0.312 (6.8e-03)	0.117 (4.2e-03)	0.463 (9.2e-03)	0.315 (7.3e-03)	0.112 (4.0e-03)	0.386 (4.0e-02)
NNPE	-	**0.154 (6.1e-04)**	**0.035 (2.4e-04)**	0.019 (1.7e-03)	0.214 (7.4e-03)	0.064 (4.2e-03)	0.071 (1.5e-02)
NPE-DANN	0.01	0.217 (3.2e-03)	0.065 (8.7e-04)	0.017 (1.8e-03)	0.180 (7.9e-04)	0.056 (1.6e-03)	0.016 (1.2e-03)
NPE-DANN	0.10	0.211 (4.4e-03)	0.057 (2.2e-03)	0.015 (9.6e-04)	**0.178 (1.7e-02)**	**0.045 (5.1e-03)**	0.014 (6.1e-04)
NPE-DANN	1.00	0.240 (9.0e-03)	0.068 (3.6e-03)	0.015 (4.3e-04)	0.201 (2.0e-02)	0.051 (8.0e-03)	0.014 (2.4e-04)
NPE-MMD	0.01	0.268 (4.2e-03)	0.085 (3.1e-03)	0.016 (6.4e-04)	0.262 (1.1e-03)	0.085 (6.4e-04)	0.016 (3.9e-04)
NPE-MMD	0.10	0.245 (6.3e-03)	0.071 (3.6e-03)	0.013 (1.5e-04)	0.181 (9.7e-03)	0.047 (3.8e-03)	0.012 (3.0e-04)
NPE-MMD	1.00	0.321 (4.2e-04)	0.129 (3.4e-04)	0.000 (3.1e-06)	0.322 (5.3e-04)	0.129 (2.9e-04)	-0.000 (2.7e-06)

Table B.4: **Experiment 3**: Overview of the metrics in the different misspecification scenarios (mean and standard deviation of three runs). Please refer to Table 1 for a detailed description. Note that each standard deviation is given for a constant set of hyperparameters, so it only covers the computational uncertainty for a given setting. As shown by the performance changes when changing λ, hyperparameters have a large influence on the results, and different hyperparameter choices might lead to qualitative changes in the results.

B.6 Experiment 4 - Decision Making

Real-World Data We utilize empirical response time data from Implicit Association Test (IAT) experiments conducted in online settings and provided by Project Implicit (Xu et al., 2014). In the IAT, participants categorize words and images into two target categories as quickly and accurately as possible, aiming to respond swiftly while minimizing errors. We use a subsample from the standard Race IAT data set, collected between late 2014 and early 2015, with an initial sample size of $N = 166,283$. For each person, we use 120 experimental trials, 60 from each of the main experimental conditions in the IAT.

Probabilistic Model As our cognitive model, we employ the diffusion decision model (DDM; Ratcliff et al., 2016). The DDM models the observed choices and reaction times as outcomes of a noisy evidence accumulation process and has been successfully applied to this and similar data sets (Klauer et al., 2007; Kvam et al., 2024). We base our modeling approach on previous work with the same data set (von Krause et al., 2022; von Krause & Radev, 2025, see these papers for the exact model formulation). To capture differences between experimental conditions – typically labeled congruent and incongruent – we specify separate drift rates and boundary separation parameters for each condition. Additionally, we introduce distinct non-decision time parameters for correct and error responses, to account for how error response times are recorded in the Project Implicit data set (i.e., the error RT is not saved directly, but only after the initially erroneous answer has been corrected). We adopt broad yet informative priors based on previous modeling studies utilizing the same data (von Krause & Radev, 2025), to maximize the closeness of our approach to real-world applications.

Network Architecture For the summary network ϕ, we utilize a deep set architecture (Zaheer et al., 2017) that reduces the input (response time, accuracy, and experimental condition of 120 trials per person) to a 12-dimensional representation. We use a dropout rate of 0.05 to avoid overfitting due to offline training. The generative inference network q is based on an affine coupling flow model (Ardizzone et al., 2021; Kingma & Dhariwal, 2018), consisting of six invertible layers. As before, the domain classifier ψ used in NPE-DANN is a conventional feedforward neural network with three hidden layers, each comprising 256 units. We omit both label smoothing and any reweighting of the gradient reversal component. We use an AdamW optimizer with an initial learning rate of $1 \cdot 10^{-4}$ and cosine decay.

Training and Evaluation Details All neural networks are trained offline using a fixed simulation budget of $32,000$ data sets. Each model is trained for 100 epochs with a batch size of 32, resulting in $1,000$ iterations per epoch. For validation during training, we generate an additional $1,000$ simulated data sets. In the case of NPE-UDA methods, we also use $32,000$ empirical data sets for training and another $1,000$ for validation. We use $S = 100$ posterior samples per method and test data set to keep computation times reasonable.

To evaluate model performance, we construct four distinct test sets from the empirical data. We begin by categorizing the remaining $133,283$ examples – after removing training and validation data – into well-specified and misspecified subsets. Well-specified data sets closely resemble the simulated data, whereas misspecified ones are more likely to reflect model mismatch. To define this distinction, we use an out-of-distribution detection approach in summary space inspired by Schmitt et al. (2023): We train three standard NPE networks with equal settings and extract 12-dimensional summary embeddings for $10,000$ simulated test sets. We then compute the 90th percentile of Mahalanobis distances within the embedding space for each network. Averaging the resulting quantiles across the three runs yields a threshold, which we use to classify empirical data sets: Those with embeddings within the threshold are labeled well-specified, while those exceeding it are considered misspecified. We calculate the mean Mahalanobis distance for the embeddings of each empirical test data set – using the covariance matrix derived from simulated data – averaged across the three networks. Based on these distances, we categorize each data set accordingly. For subsequent analyses, we use as test data all 730 data sets labeled as misspecified, along with a separate random subsample of $10,000$ well-specified examples.

Both well-specified and misspecified test sets undergo a data cleaning procedure, resulting in two versions of each: a raw (uncleaned) and a cleaned data set. Cleaning involves removing response times below 200 ms or above 10 seconds, as well as intra-individual outlier trials. These outliers are identified as trials with

log-transformed response times falling outside 1.5 interquartile ranges (IQRs) from the 25% or 75% quantiles, calculated separately for each participant, condition, and trial correctness.

For the NNPE method, we again apply the spike-and-slab procedure to introduce noise into the input data—in this case, the response times. To prevent any sign reversal due to contamination, we enforce a lower bound by setting any negative RT values to 0.01 seconds.

Metrics As our first evaluation metric, we compute the MMD between the embedding spaces of simulated test data and uncleaned empirical test data. We hypothesized that UDA methods would yield better alignment—i.e., lower MMD values—compared to standard NPE, reflecting improved generalization to empirical data.

Our second metric dimensions are concerned with external validity. We examine correlations between individual participants' posterior median cognitive parameters and their age—available in the Project Implicit data set. Prior research has shown robust age-related effects, particularly on the boundary separation parameter and non-decision time (for correct trials). We therefore compute these correlations across all networks and test sets to evaluate the extent to which these known effects are recovered.

Our third metric focuses on posterior predictive distances. For each participant in the empirical test set, we use our trained networks to approximate 100 samples from the joint posterior distribution over cognitive model parameters. Using these samples, we re-simulate data and compare them to the original empirical data using root mean square error (RMSE) on a set of summary statistics: mean accuracy and response time quantiles (10%, 30%, 50%, 70%, and 90%), split by condition and further separated into correct and error trials. RMSEs are averaged across posterior samples, participants, quantiles, and experimental conditions. This yields one RMSE value each for accuracy, correct response times, and error response times per estimation method and per test data set (i.e., well-specified/clean, misspecified/clean, well-specified/unclean, and misspecified/unclean).

Additional Results Our last metric again uses MMD, but this time to quantify the discrepancy between the prior distributions of cognitive parameters and their corresponding posterior distributions after inference on empirical data. We utilize this divergence between prior and posterior distributions of cognitive parameters to test the hypothesis of Huang et al. (2023) that increasing λ values lead to a convergence of the posterior to the prior. Figure B12 shows the results, with little evidence for a convergence to the prior up to $\lambda = 1.0$. However, we observe a sharp decrease in the distance between prior and posterior distributions for $\lambda = 10$ (not shown), suggesting that the model learns less from the data at this level of UDA influence. Similarly, while the lowest summary space distances are observed at $\lambda = 10$, the PPD reveals that, at this setting, the estimated parameters fail to adequately reproduce the empirical data.

Together, these findings highlight the importance of carefully balancing the UDA component in training: while stronger alignment can improve summary-level similarity, excessive emphasis may drastically impair parameter recovery and reduce the informativeness of the posterior. As parameter estimation effectively fails for the $\lambda = 10$ networks, we exclude these from further analyses.

Finally, we also compared results for shorter training time (50 instead of 100 epochs) and uniform instead of informative priors for the DDM parameters. In both cases, the alternative settings lead to worse performance across metrics and methods, so we omit these results for reasons of brevity.

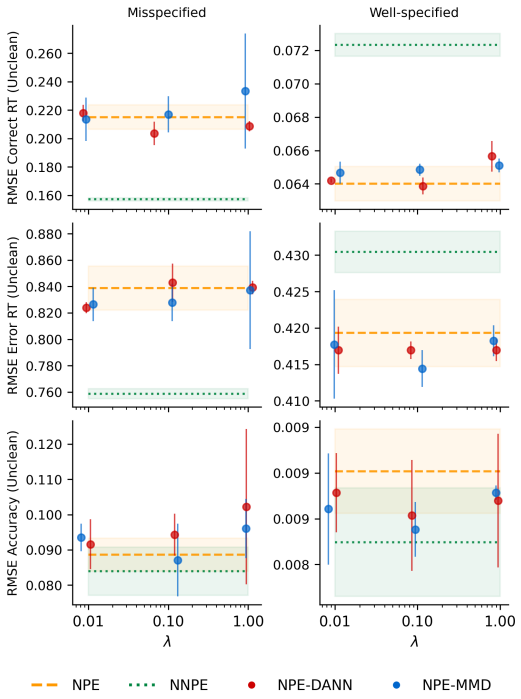


Figure B11: **Experiment 4** – Posterior predictive distance (RMSE) to **uncleaned** empirical response time data, averaged across three runs per method. Across run standard deviations shown as shaded areas (for NPE and NNPE) or error bars (for NPE-DANN and NPE-MMD). The left column shows results for misspecified data sets, while the right column shows results for the (much larger) well-specified data set. The first row shows the network's prediction error for response times (in seconds) on correct trials, averaged across posterior samples, participants, response time quantiles, and experimental conditions. The second row shows the same metric for error response times, while the third row shows results for accuracy rates (in %). Please note that y-axis scales differ across subplots. λ = UDA weight.

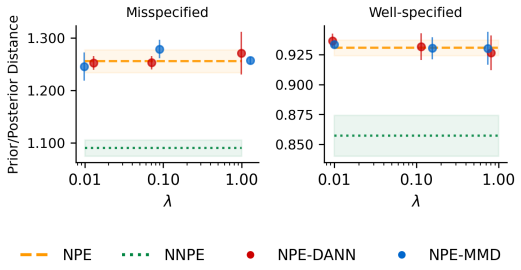


Figure B12: **Experiment 4** – MMD of prior vs. posterior distributions in the parameter space, averaged across three runs per method. Across run standard deviations shown as shaded areas (for NPE and NNPE) or error bars (for NPE-DANN and NPE-MMD). The left column shows results for misspecified data sets, while the right column shows results for the (much larger) well-specified data set. Please note that y-axis scales differ across subplots. λ = UDA weight.

Declaration in accordance to § 8 (1) c) and d) of the doctoral degree regulation of the Faculty

D

FAKULTÄT FÜR
VERHALTENS- UND EMPIRISCHE
KULTURWISSENSCHAFTEN

UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

Promotionsausschuss der Fakultät für Verhaltens- und Empirische Kulturwissenschaften der Ruprecht-Karls-Universität Heidelberg / Doctoral Committee of the Faculty of Behavioural and Cultural Studies of Heidelberg University

Erklärung gemäß § 8 (1) c) der Promotionsordnung der Universität Heidelberg für die Fakultät für Verhaltens- und Empirische Kulturwissenschaften / Declaration in accordance to § 8 (1) c) of the doctoral degree regulation of Heidelberg University, Faculty of Behavioural and Cultural Studies

Ich erkläre, dass ich die vorgelegte Dissertation selbstständig angefertigt, nur die angegebenen Hilfsmittel benutzt und die Zitate gekennzeichnet habe. / I declare that I have made the submitted dissertation independently, using only the specified tools and have correctly marked all quotations.

Erklärung gemäß § 8 (1) d) der Promotionsordnung der Universität Heidelberg für die Fakultät für Verhaltens- und Empirische Kulturwissenschaften / Declaration in accordance to § 8 (1) d) of the doctoral degree regulation of Heidelberg University, Faculty of Behavioural and Cultural Studies

Ich erkläre, dass ich die vorgelegte Dissertation in dieser oder einer anderen Form nicht anderweitig als Prüfungsarbeit verwendet oder einer anderen Fakultät als Dissertation vorgelegt habe. / I declare that I did not use the submitted dissertation in this or any other form as an examination paper until now and that I did not submit it in another faculty.

Vorname Nachname / First name Family name	Lasse Elsemüller
Datum / Date	01.08.2025
Unterschrift / Signature	Dem Dekanat der Fakultät für Verhaltens- und Empirische Kulturwissenschaften liegt eine unterschriebene Version dieser Erklärung vom 01.08.2025 vor.