PHILOSOPHICAL TRANSACTIONS A

royalsocietypublishing.org/rsta

Opinion piece

Cite this article: Bürkner P-C, Schmitt M, Radev ST. 2026 Simulations in statistical workflows. *Phil. Trans. R. Soc. A* **384**: 20240616.
https://doi.org/10.1098/rsta.2024.0616

Received: 31 March 2025
Accepted: 14 September 2025

One contribution of 15 to a theme issue 'Statistical workflow'.

Subject Areas:
statistics

Keywords:
simulation-based inference, amortized inference, Bayesian statistics, frequentist statistics, statistics

Author for correspondence:
Paul-Christian Bürkner
e-mail: paul.buerkner@gmail.com

Simulations in statistical workflows

Paul-Christian Bürkner[1], Marvin Schmitt[2] and Stefan T. Radev[3]

[1]Department of Statistics, TU Dortmund, Dortmund, North Rhine-Westphalia, Germany
[2]Independent scientist, Helsinki, Finland
[3]Department of Cognitive Science, Rensselaer Polytechnic Institute, Troy, NY, USA

P-CB, 0000-0001-5765-8995

Simulations play important and diverse roles in statistical workflows, for example, in model specification, checking, validation and even directly in model inference. Over the past decades, the application areas and overall potential of simulations in statistical workflows have expanded significantly, driven by the development of new simulation-based algorithms and exponentially increasing computational resources. In this paper, we examine past and current trends in the field and offer perspectives on how simulations may shape the future of statistical practice.

This article is part of the theme issue 'Statistical workflow'.

1. Introduction

Computer simulations have granted modern scientists inclusive access to 'would-be' worlds created by devices of imaginary results [1]. These worlds can be deterministic or random-like [2]; they can carry new insights or span an epistemic void [3]; they can merely augment or altogether redefine scientific models [4]. Simulation can be modestly viewed as an aid to the golden path of experimentation [5] or as a science in its own right—*a science of simulation* [6]. Accounts of simulation as a 'new' scientific method seem to emerge at regular intervals throughout the history of ideas, for instance, as part of the *sciences of the artificial* [7], as part of a *digital revolution in science* [6] or as part of *simulation intelligence* encompassing an artificial intelligence (AI)-permeated computational toolkit [8].

THE ROYAL SOCIETY PUBLISHING

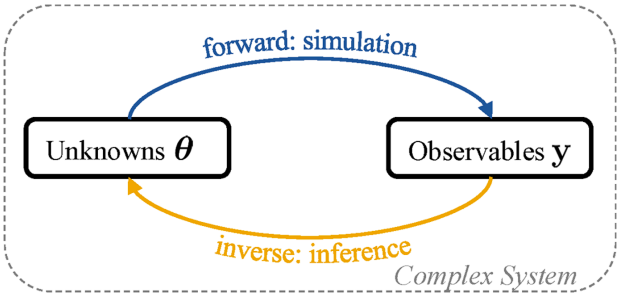

Figure 1. Forward simulation and inverse inference are two central components in a statistical model of unknowns θ and observables y.

In statistics, the emergence of simulation is typically associated with the birth of Monte Carlo (MC) methods during World War II [9], eventually leading to the publication of the Metropolis algorithm [10]. Hastings [11] and Peskun [12] later generalized the Metropolis algorithm as a family of simulation-based tools, namely, Markov chain Monte Carlo (MCMC), designed to mitigate the curse of dimensionality in direct MC estimation. Crucially, the inception of MC methods coincides with the end of what Efron & Hastie [13] characterize as *classical statistical inference* and the beginning of early computer-age methods, such as bootstrap and jackknife, that demand randomized resampling and mechanical repetition.

These early simulation methods greatly expanded the reach of inferential statistics, as they provided mechanical solutions to problems that were previously analytically intractable (e.g. high-dimensional expectations) or too cumbersome to execute manually (e.g. non-parametric estimation via bootstrap or permutation tests). Moreover, it was not until general-purpose computers had shrunk considerably in size that Bayesian inference became more than '…a macho activity enjoyed by those who were fluent in definite integration' [14, p. 319]. In a way, digital simulation fulfilled Pearson's dream of a universal tool for carrying out statistical experiments with synthetic coins and roulettes [15], ushering in the widespread application of both frequentist and Bayesian paradigms.

However, we argue that Bayesian methods have gained the most from the transition to computational statistics and the rise of simulation-based tools. Unlike frequentist approaches, which often rely on asymptotic approximations, Bayesian methods require the evaluation of high-dimensional posterior distributions that are rarely analytically tractable [16]. Put simply, Bayesian inference is hard because integration is hard. Thus, the same MCMC methods developed for approximating energy distributions in statistical physics have proven instrumental for sampling from *conditional distributions* arising in Bayesian analysis [9].

Early MCMC methods freed Bayesian analysis from the confines of conjugate models and closed-form posteriors, most notably through the introduction of the Gibbs sampler [17,18] and hybrid variants [19]. However, these methods still required *explicit models* with tractable likelihoods. By contrast, *implicit models*—defined through a stochastic mechanism (e.g. a randomized simulation) without a known distribution for computing the likelihood of their outputs [20]—remained out of reach for Bayesian methods until the advent of approximate Bayesian computation (ABC; [21,22]). In ABC, stochastic simulation explicitly takes on a dual role: it serves both as an epistemic tool for solving *forward problems* (from unknowns to knowns) and as a computational tool for solving *inverse problems* (from knowns to unknowns). Indeed, the coinage of the term simulation-based inference (SBI; [23]) is reflective of this role (see figure 1).

As the notion of statistical model loses its well-defined contours in an AI-dominated conceptual landscape, the role of simulation in statistics becomes increasingly difficult to delineate from its role in science in general. However, since this paper focuses on statistical workflows—iterative processes of model building and model criticism—our discussion will centre on the versatility

of simulating *from* statistical models. In particular, this means generating random draws from model-implied distributions, much like drawing random cards from a cleverly arranged deck. Given this focus, we will not detail sampling algorithms that primarily rely on evaluating the *density* of statistical models rather than their generative capabilities.

Among density-based approaches, the most prominent family of algorithms is undoubtedly MCMC [24]. MCMC has seen rapid developments in recent years, including locally adaptive methods that can handle distributions with complex geometries [25,26] and methods that leverage modern hardware for large-scale parallelization [27,28]. Other popular families of algorithms are sequential Monte Carlo (SMC; [29]) and importance sampling [30]. Even though these algorithms involve simulation steps, they typically do not require direct draws from the model's generative distribution, but instead rely on simulating proposals or particles guided by density evaluations.

The organization of the following sections elucidates use cases for statistical model simulations throughout the key stages of a minimalist statistical workflow, namely: (i) model specification, (ii) model verification, (iii) model inference, and (iv) model checking. We conclude with a brief outlook on simulation intelligence and the ascent of automated statistics.

2. Model specification

In a previous work, we argued that Bayesian modelling has evolved beyond traditional likelihood-prior formulations to encompass complex SBI techniques, posterior approximators and amortized learning strategies [31]. To structure this expanding landscape, we proposed the PAD taxonomy, which categorizes Bayesian models along three fundamental axes:

- *P (**P**robability Distribution)*. The probabilistic joint model (hereafter called P model) that defines the relationships between model parameters, latent variables and observed data.
- *A (Posterior **A**pproximator)*. The inference algorithm used to approximate the posterior distribution, ranging from MCMC methods to neural networks.
- *D (Training **D**ata)*. The observed data used to condition the model and update beliefs.

According to the PAD model taxonomy, *model specification* refers to defining the components of a P model. In a Bayesian setting, this entails formulating a prior $\pi(\theta)$ and an observation model $\pi(y \mid \theta)$ that together determine the generative joint model $\pi(\theta, y)$. In a frequentist setting, the P model reduces to the observation model $\pi(y \mid \theta)$, which is defined over a specific domain $\Theta \ni \theta$.

As part of a Bayesian workflow [32], the plausibility of P models can already be visualized and evaluated through prior predictive or prior pushforward checks [33]. The purpose of these checks is to assess whether the generative behaviour of a P model is consistent with the available domain expertise via simulations from the model-implied (i.e. prior predictive) distribution, $\pi(y) = \int \pi(y \mid \theta)\pi(\theta)\mathrm{d}\theta$. Consistency with domain expertise (i.e. known knowns and known unknowns) not only figures in domain-specific Bayesian workflows [34] but can also be considered as a crucial aspect of iterative theory building in general.

Here, once again, simulation serves as an antidote to complexity: consistency with domain expertise may not be immediately evident from the model formulation alone, even when a model's specification appears beguilingly simple (see [35] for a probabilistic extension of Conway's Game of Life). Moreover, if the data y is high-dimensional, the utility of consistency checks can be challenging to assess. This is where *prior pushforward checks* come into play [33], following the procedure suggested by Schad *et al*. [34]: (i) define a low-dimensional, interpretable summary statistic $T(y)$, (ii) determine plausible value regions for the summary statistic, and (iii) simulate model predictions to ensure the summary statistic falls within these regions[1].

Finally, the simulation-based procedure outlined above can be inverted for the purpose of *prior elicitation*, which seeks solutions to the hard problem of transforming non-probabilistic domain knowledge into well-defined prior distributions [36]. Accordingly, recent optimization-based

[1]In a frequentist setting, the prior can be replaced with a set of plausible values for the parameters, $\Theta_{\text{plausible}}$.

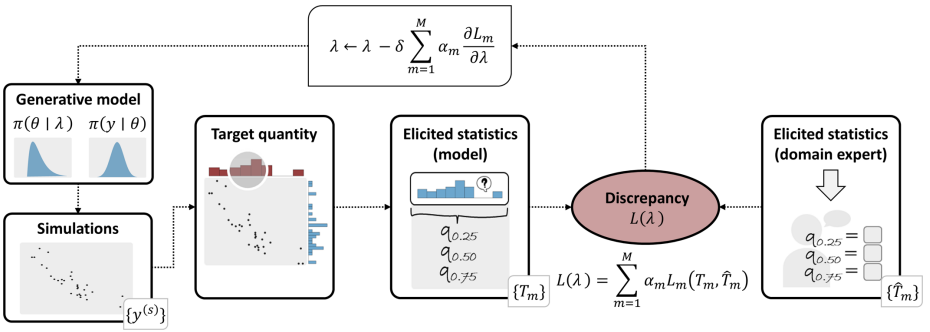

Figure 2. Graphical illustration of the simulation-based framework for prior elicitation proposed by Bockting *et al*. [37]. The process starts by identifying target quantities via the domain expert and eliciting expert statistics $\hat{T}$. Model predictions are then simulated by sampling from a parametric prior $\pi(\theta \mid \lambda)$ and computing model-implied target quantities T. A loss function L assesses the consistency between model and expert-elicited statistics and adjusts prior hyperparameters λ to minimize discrepancies. The process continues until prior predictions align closely with expert knowledge.

approaches attempt to *learn* the consistency between expert knowledge and model predictions by minimizing discrepancies between expert-elicited and model-generated summary statistics ([37,38], see also figure 2). Most recently, generative networks have been employed to infer non-parametric joint priors from sparse expert knowledge and synthetic data [39,40], opening new avenues for creative utilization of model simulations.

3. Model verification

Before deploying a statistical model on real data, we should first ensure that the interplay between the probabilistic joint model P and the approximator A functions as intended in the scenarios they were designed for, namely, when their underlying assumptions hold. In many cases, simulations offer the only feasible means of verifying such correctness, or at least gauging *in silico* whether inferences are sufficiently informative for their intended purposes.

(a) Verifying calibration

Proper uncertainty calibration is crucial for ensuring confidence in estimates and predictions from statistical models [41,42]. Informally, we can say that the uncertainty of an approximator is well calibrated if it captures the true amount of uncertainty in a system. Except for special cases, accessing this true uncertainty is only ever possible with simulations. Below, we formalize the verification process for uncertainty calibration for both Bayesian and frequentist perspectives.

(i) Bayesian calibration

In the following, we will denote an approximator of the true posterior $\pi(\theta \mid y)$ as $q(\theta \mid y)$. This $q(\theta \mid y)$ may be given implicitly, for example, in the form of an MCMC algorithm or explicitly, for example, in the form of a generative neural network. The accuracy of an approximator $q(\theta \mid y)$ is difficult to study analytically, since the true posterior $\pi(\theta \mid y)$ is rarely known in the first place. Fortunately, certain Bayesian self-consistency properties can be used even without knowledge about $\pi(\theta \mid y)$. Consider, for instance, a target quantity of interest $T = T(\theta)$, which may be derived from the parameters θ. This could include the parameters themselves or any pushforward quantity, such as posterior predictions. For all target quantities T and all uncertainty regions $U_\alpha(T(\theta) \mid y)$ obtained from $\pi(\theta \mid y)$ with nominal coverage probability α (e.g. credible intervals based on posterior

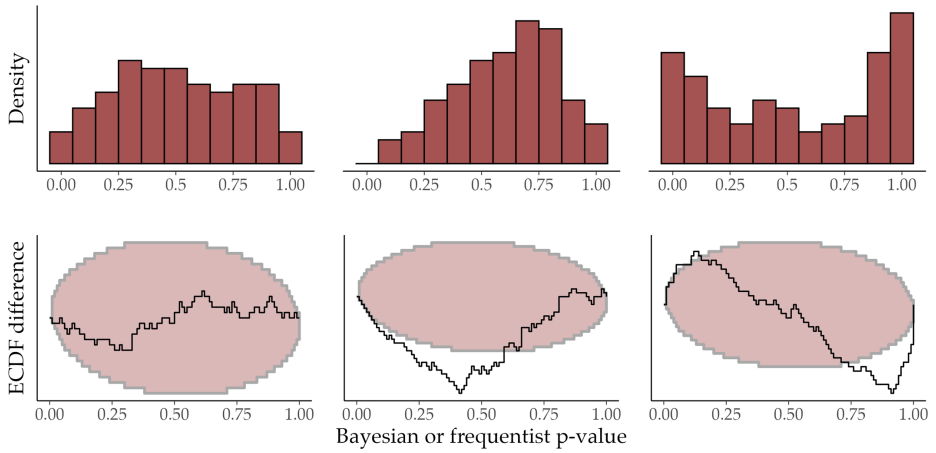

Figure 3. Simulation-based *p*-value histograms (top) and corresponding empirical cumulative distribution function (ECDF) difference plots [46] (bottom) for three hypothetical quantities of interest. The pink areas in the ECDF difference plots indicate 95% confidence intervals under the assumptions of uniformity and thus allow for a null-hypothesis significance test of Bayesian and frequentist model calibration. Left: a well-calibrated quantity. Centre: a miscalibrated quantity with too few small *p*-values. When testing for self-consistency, this indicates a positive bias in the quantity's estimates. Right: a miscalibrated quantity with too many extreme *p*-values. When testing for self-consistency, this indicates overconfident uncertainty estimates (i.e. under-dispersion).

quantiles), we know that

$$\alpha = \int \int \mathbb{I}(T^* \in U_\alpha(T \mid y))\, \pi(y \mid \theta^*)\, \pi(\theta^*)\, \mathrm{d}y\, \mathrm{d}\theta^*, \tag{3.1}$$

where $\theta^* \sim \pi(\theta^*)$ are prior draws that serve as the ground truth for the corresponding target quantity $T^* = T(\theta^*)$. In other words, the probability that an uncertainty region with coverage probability α contains the true value must be equal to α on average over the data-generating process. For a univariate quantity T, we can formulate this requirement as follows: under perfect Bayesian calibration, the posterior probability $p := \Pr_{\pi(\theta \mid y)}(T \leq T^*)$ (i.e. the 'Bayesian *p*-value') is uniformly distributed between 0 and 1 [42–44]. For a good approximation $q(\theta \mid y)$ of the true posterior $\pi(\theta \mid y)$, the above-mentioned property should hold within tolerable error bounds. Unfortunately, the associated integrals in equation (3.1) are not analytic, so none of their properties can be verified directly for $q(\theta \mid y)$. However, we can employ simulations from the underlying P model. For a dataset y, we define

$$p(T^*, T^{(1:M)} \mid y) := \frac{1}{M} \sum_{m=1}^{M} \mathbb{I}(T^{(m)} \leq T^*), \tag{3.2}$$

where $\theta^{(m)} \sim q(\theta \mid y)$ are M draws from the approximate posterior and $T^{(m)} = T(\theta^{(m)})$. If the posterior approximator is equal to the true posterior, the distribution of the posterior probabilities $p(T^*, T^{(1:M)} \mid y)$ will follow a discrete uniform distribution between 0 and 1 [45].

To test this property empirically, we first sample S parameter draws from the prior, $\theta^{*(s)} \sim \pi(\theta^*)$ and subsequently S datasets $y^{(s)} \sim \pi(y \mid \theta^{*(s)})$ from the likelihood. Then, we obtain M draws $\theta^{(m)} \sim q(\theta \mid y^{(s)})$ from the approximate posterior to compute $p^{(s)} := p(T^*, T^{(1:M)} \mid y^{(s)})$. Finally, we can test the set of posterior probabilities $\{p^{(s)}\}$ for uniformity (see figure 3 for an illustration). More powerful methods for diagnosing *joint calibration* are possible, including training a classifier to learn test statistics directly from simulations [47,48] or using the model likelihood if available [45].

These procedures for checking Bayesian calibration are collectively referred to as simulation-based calibration (SBC) [44,45]. Crucially, SBC requires *nested simulations*: in the outer loop, we

simulate parameters $\theta^{*(s)}$ and data $y^{(s)}$ from the joint model $\pi(\theta^*)\,\pi(y \mid \theta^*)$; in the inner loop, we sample $\theta^{(m)}$ from the approximate posterior $q(\theta \mid y^{(s)})$. As a result, SBC is typically computationally intensive, yet it remains a general method for validating approximate posteriors.

(ii) Frequentist calibration

Frequentist calibration works very similarly to Bayesian calibration, with two main differences. First, we do not sample true parameter values from a prior, $\theta^* \sim \pi(\theta^*)$, but rather treat θ^* as fixed at some value. Second, our target of inference is not the true posterior $\pi(\theta \mid y)$, but a point estimator $\hat{\theta} = \hat{\theta}(y)$ of the parameters, along with the true sampling distribution $\pi(\hat{\theta} \mid \theta^*)$ of the point estimator. This, in turn, implies a point estimator $\hat{T} = T(\hat{\theta})$ for any quantity of interest T, with a corresponding true sampling distribution $\pi(\hat{T} \mid T^*)$. The latter can be transformed into uncertainty regions $U_\alpha(\hat{T} \mid y)$ (i.e. confidence intervals), which, under perfect frequentist calibration, will contain the true value T^* for α per cent of the datasets generated from $\pi(y \mid \theta^*)$

$$\alpha = \int \mathbb{I}(T^* \in U_\alpha(\hat{T} \mid y))\,\pi(y \mid \theta^*)\,\mathrm{d}y. \tag{3.3}$$

If T is univariate, we again obtain an equivalent but simpler formulation: under perfect frequentist calibration, the sampling distribution probability $p := \mathbb{P}_{\pi(\hat{T} \mid T^*)}(\hat{T} \leq T^*)$ (i.e. the p-value) is uniformly distributed between 0 and 1 [42,49]. This enables testing the calibration of an approximator $q(\hat{T} \mid T^*)$ of the true sampling distribution $\pi(\hat{T} \mid T^*)$ via simulations: sample S draws $y^{(s)} \sim \pi(y \mid \theta^*)$, compute $p^{(s)} = \mathbb{P}_{q(\hat{T} \mid T^*)}(\hat{T} \leq T^*)$ using the approximate sampling distribution and then test the resulting set of p-values $\{p^{(s)}\}$ for uniformity.

(iii) Power analysis

The frequentist self-consistency of p-values holds only if the true parameter value θ^* used for simulating data from the likelihood $\pi(y \mid \theta^*)$ is the same parameter assumed in the sampling distribution $\pi(\hat{\theta} \mid \theta^*)$. In a null-hypothesis significance framework (see also §4a), we consider a sampling distribution $\pi(\hat{\theta} \mid \theta_0)$ given the parameter value θ_0 characterizing the null-hypothesis. If the θ^* assumed in the sampling process $\pi(y \mid \theta^*)$ is different from θ_0, uniformity of the p-value distribution is no longer expected. Instead, when θ^* represents an explicit alternative hypothesis, the p-value distribution measures statistical power. Simulations are the prime tool for performing power analysis since analytical expressions of sampling distribution are often only available for few special parameter values, usually representing the null hypothesis [50]. Simulation-based power analysis can also be performed in a Bayesian framework by choosing a prior $\pi(\theta_0)$ within the estimated model that differs from the prior $\pi(\theta^*)$ used in the data-generating process.

(b) Other simulation studies

Good uncertainty calibration is undoubtedly an important property that should ideally be verified before entrusting the pair of probabilistic model and approximator with performing inference on real data. However, calibration alone is often insufficient. For example, when only testing parameters for calibration in a Bayesian model, $T(\theta) = \theta$, then not only the posterior $\pi(\theta \mid y)$ but also the prior $\pi(\theta)$ would pass the above-described uniformity tests [45]. Accordingly, we have to additionally study the *sharpness* of uncertainty estimators [42], which essentially indicates how small the uncertainty regions are for any fixed α (see also [31]). Among all well-calibrated uncertainty estimators, we would then choose the one that is sharpest. As for calibration, sharpness is almost always non-analytic, so simulations are the only way to study it.

Even when not considering uncertainty at all, but just studying the accuracy of point estimators, simulations are required. We can measure the accuracy of a point estimator $\hat{T}$ as the distance

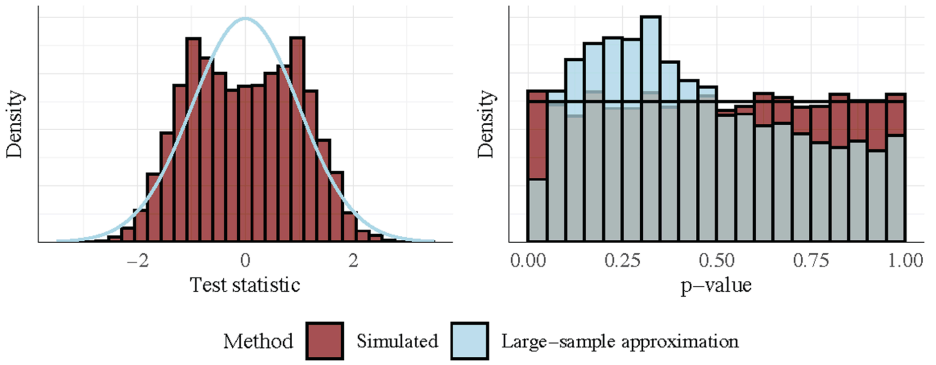

Figure 4. Illustration of a simulation-based test. In the chosen scenario, the means of the two groups are compared using a two-sample *t*-test for equal variances. Two independent datasets, each consisting of $N = 40$ observations, were simulated from LogNormal($\mu = 2, \sigma = 2$). The null hypothesis asserts equal group means and equal variances. However, the data deviate substantially from normality. Accordingly, even for $N = 40$ per group, the left plot shows that the true distribution of the test statistics (approximated via simulations; red histogram) is clearly different from the large sample approximation assumed by the *t*-test (blue density). The right plot shows that the corresponding *p*-value distribution under the null hypothesis is highly non-uniform for the large sample approximation (blue histogram). By contrast, *p*-values obtained via a simulation-based test ($S = 10\,000$) are almost perfectly uniform, with only small random error (red histogram).

D to its target T^* averaged across the data-generating process

$$\bar{D}(\hat{T}, T^*) := \int D(\hat{T}(y), T^*)\, \pi(y \mid \theta^*)\, \mathrm{d}y. \tag{3.4}$$

For example, when D is the squared difference, this leads to the mean squared error (m.s.e.)

$$\text{m.s.e.}\,(\hat{T}, T^*) := \int (\hat{T}(y) - T^*)^2\, \pi(y \mid \theta^*)\, \mathrm{d}y \approx \frac{1}{S} \sum_{s=1}^{S} (\hat{T}(y^{(s)}) - T^*)^2, \tag{3.5}$$

approximated via draws $y^{(s)} \sim \pi(y \mid \theta^*)$. The same can be done for a Bayesian point estimator (e.g. the posterior mean or median) by first sampling $\theta^{*(s)} \sim \pi(\theta^*)$ and then $y^{(s)} \sim \pi(y \mid \theta^{*(s)})$.

4. Model inference

Simulations can also be used directly for model inference [23], that is, to estimate parameters or obtain decisions from the triple of statistical model, approximator and observed data.

(a) Hypothesis testing

Every statistical test is based on a test statistic $T = T(y)$ that extracts the specific property of interest from the data y. For example, if the goal is to compare the location of two groups, then $T(y)$ could be the difference between the two group means. In addition to the test statistic itself, its distribution $\pi(T(y) \mid \theta)$ implied by the likelihood $\pi(y \mid \theta)$ needs to be analytically tractable, at least for some reference parameter value θ_0 that captures the null hypothesis (e.g. no difference).

In some cases, the sampling distribution under the null hypothesis is indeed analytic, for instance, as in *t*-tests for normally distributed data, but often enough it is not. We may then appeal to asymptotic approximations, under which the (sampling) distribution of the test statistic is a well-characterized distribution in the limit of an infinite number of simulations [51]. However, the accuracy of these approximations for smaller sample sizes can be questionable (figure 4).

Simulation-based tests provide a powerful alternative to analytic solutions or large-sample approximations [52,53]. They generally proceed as follows: sample S draws $y_0^{(s)} \sim \pi(y \mid \theta_0)$ from the

likelihood given θ_0. Then, compute the test statistic $T_0^{(s)} := T(y_0^{(s)})$ for each draw, which implies $T_0^{(s)} \sim \pi(T(y) \mid \theta_0)$. Thereby, we gain access to samples from the target distribution $\pi(T(y) \mid \theta_0)$ simply via predefined transformations of individual samples (see figure 4, left). Then, we compare the test statistic value $\tilde{T} := T(y_{\text{obs}})$ from the observed data y_{obs} with the distribution of simulated test statistic values $\{T_0^{(s)}\}$.

Suppose we are interested in a one-sided test for lower values of T. We would first compute $\tilde{r} = \sum_{s=1}^{S} \mathbb{I}(T_0^{(s)} \tilde{T})$, where $\mathbb{I}$ is the indicator function. Then, we would compute the p-value $\tilde{p}$ corresponding to y_{obs} as the normalized rank $\tilde{p} := \tilde{r}/S$ (see figure 4, right) and the significance threshold T_α as the empirical α-percentile of a set $\{T_0^{(s)}\}$ of simulated test statistics.

This approach is very general and can even be applied on top of Bayesian methods [54]. However, it relies on two important prerequisites. First, we need to be able to sample efficiently from the likelihood, so that we can choose S large enough to obtain sufficiently accurate approximations of the p-value. Second—and often more prohibitive—we require a suitable test statistic $T(y)$ that captures the effect of interest independently of potential nuisance parameters and is straightforward to compute directly from the data. If T is a function of model parameter estimates, that is, $T = T(\hat{\theta}(y))$, then performing simulation-based testing becomes computationally intensive: obtaining model estimates for a single dataset is often already slow, and doing so S times may render the entire procedure infeasible. Taken together, these challenges have thus far limited the widespread adoption of simulation-based testing. Fortunately, efficiency issues can be addressed by *amortized methods*, which we discuss in §4b as a special case of SBI.

(b) Parameter estimation

In parameter estimation, we construct estimators that approximate the unknown parameters of a P model using the data in an attempt to solve the underlying inverse problem (i.e. estimating the unknowns from the knowns). There are myriad ways to construct useful estimators. Two popular approaches rooted in the frequentist and Bayesian frameworks, respectively, are (i) constructing well-calibrated *confidence sets*, and (ii) recovering the *posterior distribution* by updating the prior $\pi(\theta)$ with data. For the sake of clarity, we briefly reiterate the goals of these approaches.

Confidence sets. For any parameter vector θ and likelihood function $\pi(y \mid \theta)$, we are interested in constructing a random confidence set $\mathcal{R}(y)$ with nominal $1 - \alpha$ coverage, such that

$$\mathbb{P}_{\pi(y|\theta)}(\theta \in \mathcal{R}(y)) \geq 1 - \alpha \quad \forall \theta \in \Theta. \tag{4.1}$$

Posterior distributions. For a given prior $\pi(\theta)$, we are interested in updating the prior to the posterior $\pi(\theta \mid y)$, which incorporates all information about θ carried by the data y,

$$\pi(\theta \mid y) = \pi(y \mid \theta)\, \pi(\theta)\, \pi(y)^{-1}. \tag{4.2}$$

Needless to say, both problems are computationally challenging for non-analytic models (e.g. no closed-form for the Bayesian posterior or unknown frequentist sampling distribution). The situation is further aggravated by P models defined *solely* through a (randomized) algorithm for generating data y from parameter configurations θ instead of explicitly assuming a parametric data model $\pi(y \mid \theta)$ [20]. For such *implicit models*, the construction of confidence sets is challenging not only because we cannot evaluate the likelihood but also because we need to test null hypotheses across the entire parameter space [55]. By the same token, posterior approximation is difficult because not only does the marginal likelihood $\pi(y)$ involve an intractable integral, but also the implicit likelihood [23], rendering posterior approximation doubly intractable.

(i) Approximate Bayesian computation

Model simulations can be used to connect even intractable models to real data while remaining true to the likelihood principle [20,23]. Arguably, the most popular approach to SBI is ABC [56].

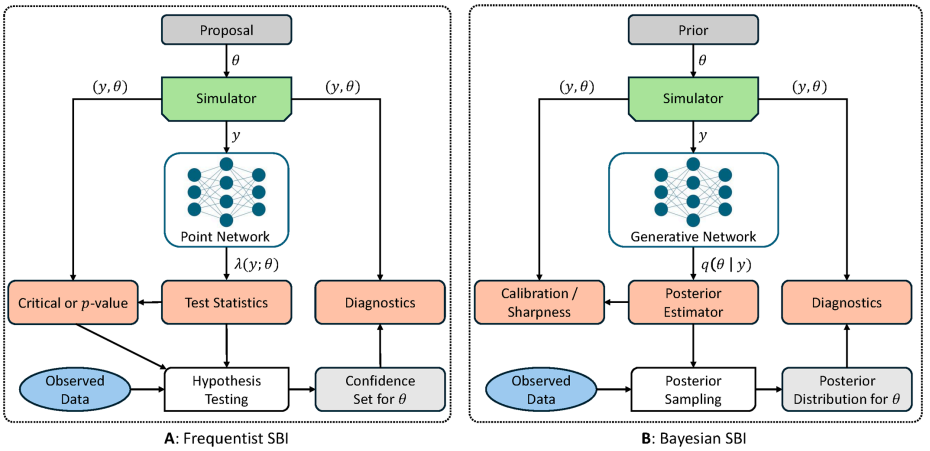

Figure 5. Graphical illustration of frequentist versus Bayesian approaches to SBI. (A) In frequentist SBI, a classifier is trained on simulated data and parameters (y, θ) to efficiently estimate a test statistic $\lambda(y; \theta)$ (e.g. the odds function). This statistic is used to compute critical and p-values via further simulations. An approximate confidence set for θ is constructed by inverting a series of hypothesis tests. The coverage of this set can be diagnosed through additional simulations (adapter after [55]). (B) In Bayesian SBI, a generative neural network is trained to estimate the posterior $\pi(\theta \mid y)$ from simulated (y, θ) pairs. The calibration and sharpness of the resulting estimator can be evaluated via further simulations. Given observed data, the estimator can rapidly sample from the approximate posterior. Further simulations can be used to evaluate the fidelity of these samples via predictive checks or OOD detection.

The original ABC rejection sampler approximates the posterior by repeatedly proposing parameters from the prior $\pi(\theta)$ and then simulating a dataset from the data model $\pi(y \mid \theta)$. If the resulting dataset is sufficiently similar to the actually observed dataset y_{obs}, the corresponding parameter vector is retained as a sample from the posterior; otherwise, it is rejected. More sophisticated versions include likelihood-free MCMC (ABC-MCMC; [57,58]), SMC (ABC-SMC; [59,60]) and various hybrid methods [61].

Despite their theoretical elegance [62], most ABC methods share the fundamental limitation of all non-amortized methods: computations must be repeated from scratch each time a model is fit to new data. Yet, there are many scenarios in which multiple model refits are necessary—for example, when modelling multiple datasets [63], performing *in silico* model verification (§3) or aiming for real-time inference [64]. To address these challenges, a different way of utilizing simulations was needed—a way to pool or 'compile' computations into global estimators that can produce near-instant results for arbitrary queries [65–67]. In hindsight, neural networks as modular and flexibly composable universal function approximators, proved to be the ideal choice.

(ii) Amortized inference via neural networks

Amortized inference asks how to flexibly reuse inferences or estimators in order to answer numerous queries without recomputation overhead. It has been proposed as a model of human probabilistic reasoning [65], a method for fast inversion of graphical models [68] and a means of learning variational posteriors with neural networks [69,70]. For statistical inference with 'white-box' models, the latter idea extends to using simulations as training data for neural networks that approximate quantities of interest, such as point estimates, test statistics or full posterior distributions [31,71] (see also figure 5). Neural networks are particularly favourable for amortized inference if the statistical model is sufficiently high dimensional (in data and/or parameter space), no good handcrafted summary statistics are available and model simulations can be obtained in reasonable time (to provide sufficient training budget).

In a frequentist setting, simulation-based training can be employed to amortize the computation of arbitrary test statistics $\lambda(y;\theta)$ [55] (see also left panel of figure 5). For instance, one viable approach is to reframe odds ratio estimation as a classification problem, in which case we can train a neural classifier to approximate the likelihood ratio [72] or odds function [55] based on simulations. Subsequently, we can learn a quantile regression of $\lambda(y;\theta)$ on θ that can approximate the critical value C_α for every α-level, allowing us to construct approximate confidence sets of the form

$$\widehat{\mathcal{R}}(y_{\text{obs}}) := \{\theta \in \Theta \mid \lambda(y_{\text{obs}};\theta) \geq \widehat{C}_\alpha\}. \tag{4.3}$$

Finally, the empirical coverage of frequentist confidence sets for every y_{obs} can be assessed on a further held-out set of simulations [55].

In amortized Bayesian inference (ABI), a generative network seeks to learn a global posterior functional $q(\theta \mid y)$ for any observation y. Typically, the network would minimize a strictly proper scoring rule $\mathcal{S}$ in expectation over the joint distribution $\pi(\theta, y)$ of the P model

$$\mathcal{L}(q) := \mathbb{E}_{\pi(\theta,y)}\Big[\mathcal{S}(q(\cdot \mid y), \theta)\Big] \approx \frac{1}{M}\sum_{m=1}^{M} \mathcal{S}(q(\cdot \mid y^{(m)}), \theta^{(m)}), \tag{4.4}$$

which would guarantee $q(\theta \mid y) = \pi(\theta \mid y)$ under perfect convergence for large simulation budgets $M \to \infty$ [42] and a universal density approximator q (e.g. coupling-based normalizing flows [73]). For instance, using the log-score for S, we retrieve the popular maximum likelihood objective,

$$\mathcal{L}^{\text{MLE}}(q) := \mathbb{E}_{\pi(\theta,y)}\big[-\log q(\theta \mid y)\big]. \tag{4.5}$$

Evaluating equation (4.5) empirically requires generative networks $q(\theta \mid y)$ with tractable densities, such as normalizing flows. More recently, score-based diffusion models have entered ABI [74–76], aimed at estimating the *score* of the (log) posterior density $\nabla_\theta \log \pi(\theta \mid y)$ by minimizing a tractable re-formulation of the weighted Fisher divergence

$$\mathcal{L}^{\text{SM}}(s) := \frac{1}{2}\int_0^T w(t)\mathbb{E}_{\pi_t(\theta_t,y)}\Big[\|s(\theta^{(t)}; y, t) - \nabla_{\theta^{(t)}} \log \pi_t(\theta^{(t)} \mid y)\|^2\Big]\, \mathrm{d}t, \tag{4.6}$$

where $w(t)$ is a positive weight function, s is a neural network and $\pi_t(\theta^{(t)}, y)$ is the model distribution defined in diffusion time t. The most popular tractable version of equation (4.6) is conditional denoising [77], but the framework includes various other 'free-form' model families capable of learning from simulations, such as flow matching [78,79].

Notably, it was not initially evident that neural networks could serve as standalone global posterior approximators. Early approaches, for example, combined neural networks with ABC [80] or SMC [66]. Papamakarios & Murra [81, p. 3] noted that generative networks could be trained over the entire prior predictive distribution $\pi(y)$, but dismissed the approach as 'grossly inefficient'.

However, subsequent work [82,83] showed that amortization over the prior (and even across different priors [84]), combined with learnable data embeddings on the fly, is in fact a viable route to fully Bayesian inference. These developments were fuelled by increasingly powerful neural density estimators (e.g. coupling-based normalizing flows [85,86]) and greater representational capacity through algorithmic alignment [87,88]. Thus, it is now clear that amortization is not wasteful but rather essential for efficiently generating model-based insights across fields as varied as cognitive science [63], evolutionary dynamics [89] and astrophysics [90], among others.

Importantly, the probabilistic calibration of all amortized methods can be efficiently checked via SBC (see §§3a and 5b), since the cost of independent posterior sampling over multiple model simulations is amortized. Additionally, the closeness of observed data to the typical set of simulations can be assessed via out-of-distribution (OOD) detection in (embedded) data space [91,92]. This is particularly important since amortized posteriors estimated from OOD data (e.g. resulting from model misspecification) tend to be biased [91], which constitutes one of the major limitations of these methods.

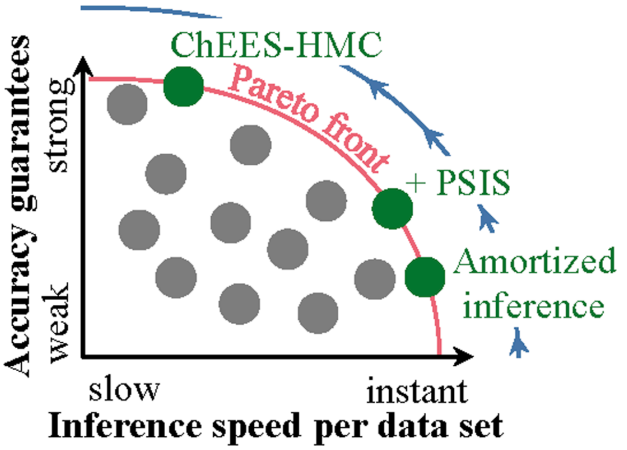

Figure 6. Estimation methods along the Pareto front balance a trade-off between inference speed and accuracy guarantees [95].

Amortized methods highlight a fruitful merger between scientific simulation and deep learning. Despite their advantages in big data and likelihood-free settings, they lack the guarantees of gold-standard MCMC as part of the standard Bayesian workflow [32], and their accuracy can break down when asked to extrapolate beyond simulated data [91,93]. Recent work seeks to address these limitations through amortized workflows that combine ABI with MCMC [94] (figure 6) and trustworthy approaches that train on both simulated and real data [92,96–98].

5. Model checking

Model checking examines the compatibility of a statistical model with observed data after inference. This *data-conditional* perspective contrasts with model verification (§3), which typically relies on simulated data (e.g. from the prior predictive distribution). By contrast, model checking uses the observed data y_{obs} to assess inference validity for the case of interest. Simulations are particularly important for scaling model checking algorithms to complex, high-dimensional settings.

(a) Predictive checking

If the data-generating process is formalized as a simulation program, we can generate synthetic *data replications* based on the inferred parameter values. Predictive checking [99] combines parameter estimation (inverse problem) with simulation (forward problem), thereby casting the inference results back into the data space.

(i) Frequentist predictive checking

In the frequentist framework, predictive checking is implemented by inserting the estimated parameter values $\hat{\theta}$ into the data model $\pi(y \mid \theta)$, leading to data simulations

$$y' \sim \pi(y \mid \hat{\theta}). \tag{5.1}$$

This process is not limited to estimates of central tendency, but it can be mirrored for other quantities, such as quantiles or confidence interval bounds of practical interest [100]. Likewise, modellers have great flexibility in the choice of metrics to evaluate the fit between the observed data y_{obs} and the simulated data replications y': any discrepancy measure $T(y_{\text{obs}}, y')$ can be used to judge whether the data replications are satisfactory (e.g. comparing averages, variances or quantiles).

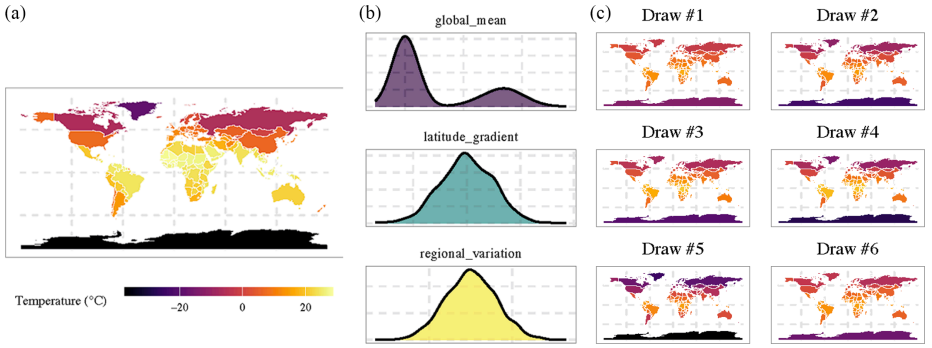

Figure 7. Statistical modelling of global temperature data with three parameters. Based on the observed global temperature y_{obs} (a), Bayesian inference yields draws from the posterior distribution of the parameters θ (b). These posterior draws can be reinserted into the simulation program (i.e. the global temperature model $f : \theta \mapsto y$) to obtain draws from the posterior predictive distribution (c). These synthetic *data replications* contain uncertainty propagated from the posterior.

(ii) Bayesian (posterior) predictive checking

In the Bayesian framework, we can propagate all uncertainty from the posterior through the data model. The result is the *posterior predictive distribution*, which expresses the distribution of new data y' while accounting for all uncertainty in the system,

$$\pi(y' \mid y_{\text{obs}}) = \int \pi(y' \mid \theta, y_{\text{obs}})\, \pi(\theta \mid y_{\text{obs}})\, \mathrm{d}\theta. \tag{5.2}$$

Concretely, we can easily draw S samples (i.e. data replications) from the posterior predictive distribution with an ancestral sampling scheme,

$$\theta^{(s)} \sim \pi(\theta \mid y_{\text{obs}}), \quad y_1'^{(s)}, \ldots, y_N'^{(s)} \sim \pi(y \mid \theta^{(s)}, y_{\text{obs}}) \quad \text{for } s = 1, \ldots, S, \tag{5.3}$$

where each simulated dataset $y'^{(s)} = \{y_1'^{(s)}, \ldots, y_N'^{(s)}\}$ contains N observations. The draws from the posterior predictive distribution represent uncertainty-aware replications of the observed data (see figure 7 for an illustrative example), where all associated uncertainty is propagated through the forward simulation process. Akin to frequentist predictive checking, any discrepancy statistic can then be computed from the observed data and the distribution of replicated data. For example, one might check whether the observed maximum value or a certain autocorrelation is extreme relative to the distribution of those metrics across the simulated replicates. If the observed data exhibit features that would be very unlikely under the posterior predictive distribution, it indicates model misfit.

(b) Posterior simulation-based calibration

As detailed in §3a(i), SBC checks if the inference is well-calibrated for data simulated from the prior predictive distribution. However, after observing real data y_{obs}, it is often more relevant to assess the inference conditional on that specific data rather than the entire prior predictive space. To address this, posterior SBC [101] uses the principles of SBC to validate the model's implementation and inference algorithm *conditioned on the observed data* by means of simulations. This approach first draws S samples $\theta'^{(s)} \sim \pi(\theta \mid y_{\text{obs}})$ from the (approximate) posterior conditional on the observed data y_{obs}. For each posterior draw, we then simulate data replications $y'^{(s)}$ from the posterior predictive distribution according to equation (5.3). Finally, for each simulated data replication $y'^{(s)}$, we draw D augmented posterior samples $\theta''^{(s,1)}, \ldots, \theta''^{(s,D)}$ conditional on the observed data y_{obs} and the respective data replication $y'^{(s)}$,

$$\theta''^{(s,1)}, \ldots, \theta''^{(s,D)} \sim \pi(\theta \mid y_{\text{obs}}, y'^{(s)}), \tag{5.4}$$

and finally check whether the augmented posterior $\pi(\theta \mid y_{\text{obs}}, y')$ is well-calibrated with the 'old' posterior $\pi(\theta \mid y_{\text{obs}})$ serving as prior. Pursuing a similar data-driven perspective, Fazio *et al*. [102] propose adding small portions of real or simulated data to improve the robustness of Bayesian simulations, which in turn leads to increased stability for SBC and model simulations more broadly (figure 1).

(c) Model comparison

Model comparison is an umbrella term for methods that seek to evaluate multiple competing models $M_1, \ldots, M_L \in \mathcal{M}$ in the context of observed data. Model comparison can be framed as an extension of model checking, where instead of validating a single model in isolation with respect to some metric, we assess *multiple candidate models simultaneously* using suitable metrics that enable direct comparisons.

Formalizing the process of comparing models helps researchers make principled and reproducible decisions about model selection, and this process can be supported with model simulations. In the context of model comparison, simulations serve a dual purpose: first, simulations render otherwise intractable algorithms computationally feasible (e.g. marginal likelihood estimation with neural SBI). Second, simulations constitute the core principle of model comparison methods, such as predictive checking of multiple candidate models or meta-uncertainty (see below).

Within the scope of model simulations for Bayesian inference, model comparison extends the data-generating simulation process by explicitly encoding the statistical data-generating model M_l,

$$y_{\text{obs}} \sim \pi(y \mid \theta, M_l) \text{ with } \theta \sim \pi(\theta \mid M_l) \text{ and } M_l \sim \pi(M), \tag{5.5}$$

where $\pi(M)$ is a discrete prior distribution over the candidate models. In *prior-based model comparison*, the central quantity is the marginal likelihood $\pi(y_{\text{obs}} \mid M_l) = \int \pi(y_{\text{obs}} \mid \theta, M_l)\, \pi(\theta \mid M_l) \mathrm{d}\theta$, which quantifies the evidence for model M_l by integrating over all parameter values. This enables the computation of Bayes factors [103] and posterior model probabilities. Since computing the marginal likelihood requires solving a potentially high-dimensional integral, this family of model comparison methods is typically computationally infeasible for complex models. Based on synthetic simulations from the joint model $\pi(\theta, y)$, amortized inference tackles this problem by directly learning the evidence (aka. evidential learning [104]) or by simultaneously learning an amortized posterior and likelihood approximator [104].

By design, prior-based model comparison is sensitive to the choice of prior [105,106], which often leads to issues in practical scenarios where prior specification is a hard challenge [107–111]. Differently, *posterior-based methods*, such as cross-validation (CV), estimate the *expected* predictive distribution over new data, $\mathbb{E}\left[\pi(y' \mid y_{\text{obs}}, M_l)\right]$ [112,113]. Taken to the extreme, leave-one-out CV (LOO-CV) evaluates the predictive distribution of single held-out observations [112]. Importantly, amortized methods can perform overhead-free LOO-CV (or other CV schemes) in cases where established approximate methods (e.g. Pareto-smoothed importance sampling; [114]) are infeasible [115].

While less sensitive to prior specification, posterior-based model comparison methods lack the consistency guarantees of marginal likelihood approaches; that is, they are not bound to recover the true data-generating model in the asymptotic limit of infinite data. Here again, model simulations have the potential to improve what we can learn from a limited amount of observed data: for example, Schmitt *et al*. [54] developed a meta-uncertainty framework to judge the *replicability* of Bayesian model comparison by combining (i) prior-based model comparison, (ii) frequentist sampling distributions based on model simulations, and (iii) posterior predictive distributions based on observed data.

(d) Sensitivity analysis

Sensitivity analysis (aka. robustness analysis [116]) constitutes a critical step in statistical modelling that focuses on understanding how the output of a model changes in response to variations in its inputs. These inputs can include fixed (hyper-)parameters of the model, the specification of prior distributions and observation models in a Bayesian setting or the choices made while pre-processing the data. As such, each analysis *hides an iceberg of uncertainty* [117], and sensitivity analysis is a principled approach to render this uncertainty tangible. In the PAD taxonomy (see §2), sensitivity analyses can target each of the three axes of Bayesian models: different statistical models, varying posterior approximators and different variations of the training data.

Since the assumed data-generating process is essentially a simulation program, simulations also lie at the heart of sensitivity analysis. By systematically varying the inputs of an analysis within a plausible space, then executing the simulation program and ultimately observing the resulting changes in the outputs, researchers can assess the robustness of their conclusions to these initial variations [116]. Simulation represents the bridging element between input variations on the one hand and inference results on the other hand. At the same time, hyperparameters in the simulation process itself can be subject to sensitivity analyses as well, which allows for powerful conclusions about the data-generating process but also comes with a yet increased degree of complexity. While sensitivity analyses might seem fairly straightforward for trivial models with low complexity, the computational demands for principled sensitivity analyses quickly become unbearable for complex statistical analyses with high-dimensional parameter spaces and involved data pre-processing routines.

Akin to tracing all sources of uncertainty in Bayesian analysis, keeping track of all possible choices in sensitivity analysis resembles a *Garden of Forking Paths* [118,119]. Taming this combinatorial explosion is the key to enabling large-scale sensitivity analyses with finite computational resources. To this end, Kallioinen *et al*. [94] developed a computationally efficient approach that uses importance sampling to estimate the effect of power-scaling the prior and likelihood in a Bayesian analysis. Elsemüller *et al*. [84] proposed *sensitivity-aware amortized Bayesian inference*, which extends the scope of amortization by encoding context information about the statistical model. By additionally conditioning on this context information during the simulation-based training stage, the trained neural networks can subsequently perform near-instant sensitivity analyses without worrying about the associated cost of modifying model assumptions during inference.

(e) Increasing the efficiency of simulation-based estimators

Many use cases of simulations we highlighted above apply the standard MC estimator $S^{-1}\sum_{s=1} f(\theta^{(s)})$ for approximating a target expectation $\mathbb{E}[f(\theta)]$. However, simulation-based estimators can become more sample-efficient than the standard MC estimator if additional information is available and correctly utilized. For example, in latent variable models, analytically integrating out the latent variables (if possible) before performing predictive simulations on new data can increase the stability of the resulting estimators [120]. As another example, 'control variates' can improve the efficiency of MC estimators by leveraging the score (i.e. the gradient of the log target density) when available [121]. More generally, if conditional expectations of the form $\mathbb{E}[f(\theta) \mid T(y)]$ for an informative statistic $T(y)$ are available, approximating these conditional expectations via simulations can yield a more efficient estimator of $\mathbb{E}[f(\theta)]$, a process also known as Rao–Blackwellization (see [122] for an overview). These examples illustrate that simulation-based methods are not an orthogonal alternative to analytical or density-based methods but can also benefit from the latter provided the right type of additional information.

6. Conclusion and outlook

In the preceding pages, we have seen that simulations are a powerful and versatile tool throughout all major steps of statistical workflows. In the foreseeable future, we expect the utility and use of simulations to increase even further. In particular, amortized inference, which relies heavily on both model simulations and continuous progress in deep learning, is likely to become a viable, widespread alternative to established inference approaches.

Going beyond the techniques available to date, one could imagine simulation-based training of entire world models that are able to rapidly and semi-automatically execute full statistical workflows on real data, essentially playing the role of a statistics expert assisting the user in their analysis. One of the key challenges for such machine-assisted statistics is how to combine the general expertise of the machine with the subject matter knowledge of the user. No matter how many simulations the machine has been trained on, it would still likely miss key subject matter knowledge about the specific real data being analysed, just as a statistics expert would not know all the intricate details of the data.

This also highlights a more general question pertinent to all inference machines trained on simulated data: how to bridge the statistical gap between simulated worlds and the real world. Put differently, we need to learn how to formally integrate information from simulated data, real data and subject matter knowledge into inferences that are both fast and trustworthy. Much remains to be simulated.

Data accessibility. This article has no additional data.

Declaration of AI use. We have not used AI-assisted technologies in creating this article.

Authors' contributions. P.-C.B.: conceptualization, funding acquisition, project administration, supervision, visualization, writing—original draft, writing—review and editing; M.S.: conceptualization, visualization, writing—original draft, writing—review and editing; S.T.R.: conceptualization, funding acquisition, supervision, visualization, writing—original draft, writing—review and editing.

All authors gave final approval for publication and agreed to be held accountable for the work performed therein.

Conflict of interest declaration. We declare we have no competing interests.

Funding. P.C.B. acknowledges support of the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) via Projects 508399956 and 528702768 as well as the Collaborative Research Center 391 (Spatio-Temporal Statistics for the Transition of Energy and Transport)—520388526. S.T.R. is supported by the National Science Foundation under grant no. 2448380.

References

1. Good IJ. 1950 *Probability and the weighing of evidence*. New York, NY: Hafners.
2. Shannon RE. 1998 Introduction to the art and science of simulation. In *Winter Simulation Conf. Proc.*, pp. 7–14, vol. 1. IEEE. (doi:10.1109/wsc.1998.744892)
3. Grim P, Singer D. 2024 Computational philosophy. In *The Stanford encyclopedia of philosophy* (eds EN Zalta, U Nodelman). Metaphysics Research Lab, Stanford University.
4. Durán JM. 2020 What is a simulation model? *Minds Mach.* **30**, 301–323. (doi:10.1007/s11023-020-09520-z)
5. Guala F. 2002 Models, Simulations, and Experiments. In *Model-based reasoning: science, technology, values*, pp. 59–74. New York, NY: Springer. (doi:10.1007/978-1-4615-0605-8_4)
6. Casti JL. 1996 *Would-be worlds: how simulation is changing the frontiers of science*. John Wiley & Sons, Inc.
7. Simon HA. 1974 *The sciences of the artificial*. Cambridge, MA: MIT Press.
8. Lavin A *et al*. 2021 Simulation intelligence: towards a new generation of scientific methods. *arXiv Preprint*
9. Robert C, Casella G. 2011 A short history of Markov chain Monte Carlo: subjective recollections from incomplete data. *Stat. Sci* **26**, 102–115. (doi:10.1214/10-sts351)
10. Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. 1953 Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087–1092. (doi:10.2172/4390578)

11. Hastings WK. 1970 Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109. (doi:10.1093/oso/9780198509936.003.0015)
12. Peskun PH. 1973 Optimum Monte-Carlo sampling using Markov chains. *Biometrika* **60**, 607–612. (doi:10.2307/2335011)
13. Efron B, Hastie T. 2021 *Computer age statistical inference, student edition: algorithms,evidence, and data science*. vol. 6. Cambridge, UK: Cambridge University Press.
14. MacKay DJ. 2003 *Information theory, inference and learning algorithms*. Cambridge, UK: Cambridge University Press.
15. Kucharski A. 2016 *The perfect bet: how science and math are taking the luck out of gambling*. London, UK: Hachette UK.
16. Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, Rubin DB. 2013 *Bayesian data analysis*, 3rd edn. London, UK; Boca Raton, FL: Chapman and Hall/CRC.
17. Geman S, Geman D. 1984 Stochastic relaxation, gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* **6**, 721–741. (doi:10.1109/tpami.1984.4767596)
18. Gelfand AE, Smith AF. 1989 Sampling based approaches to calculating marginal densities. *J. Am. Stat. Assoc.* **85**, 398–409. (doi:10.21236/ada208388)
19. Tierney L. 1994 Markov chains for exploring posterior distributions. *Ann. Statist.* **22**, 1701–1728. (doi:10.1214/aos/1176325750)
20. Diggle PJ, Gratton RJ. 1984 Monte Carlo methods of inference for implicit statistical models. *J. R. Stat. Soc. Ser. B: Stat. Methodol.* **46**, 193–212. (doi:10.1111/j.2517-6161.1984.tb01290.x)
21. Rubin DB. 1984 Bayesianly justifiable and relevant frequency calculations for the applied statistician. *Ann. Statist.* **12**, 1151–1172. (doi:10.1214/aos/1176346785)
22. Tavaré S, Balding DJ, Griffiths RC, Donnelly P. 1997 Inferring coalescence times from DNA sequence data. *Genetics* **145**, 505–518. (doi:10.1093/genetics/145.2.505)
23. Cranmer K, Brehmer J, Louppe G. 2020 The frontier of simulation-based inference. *Proc. Natl Acad. Sci. USA* **117**, 30055–30062. (doi:10.1073/pnas.1912789117)
24. Brooks S, Gelman A, Jones G, Meng XL. 2011 *Handbook of Markov Chain Monte Carlo*. Boca Raton, FL: Chapman and Hall/CRC. (doi:10.1201/b10905)
25. Biron-Lattes M, Surjanovic N, Syed S, Campbell T, Bouchard-Côté A. 2024 automala: locally adaptive metropolis-adjusted Langevin algorithm. In *International Conference on Artificial Intelligence and Statistics*, pp. 4600–4608. PMLR.
26. Modi C, Barnett A, Carpenter B. 2024 Delayed rejection Hamiltonian Monte Carlo for sampling multiscale distributions. *Bayesian Anal.* **19**, 815–842. (doi:10.1214/23-BA1360)
27. Margossian CC, Hoffman MD, Sountsov P, Riou-Durand L, Vehtari A, Gelman A. 2024 Nested ˆr: assessing the convergence of Markov Chain Monte Carlo when running many short chains. *Bayesian Anal.* **20**, 1–28. (doi:10.1214/24-BA1453)
28. Sountsov P, Carroll C, Hoffman MD. 2024 Running Markov Chain Monte Carlo on modern hardware and software. *arXiv* (doi:10.1201/9781003453420-21)
29. Del Moral P, Doucet A, Jasra A. 2006 Sequential Monte Carlo samplers. *J. R. Stat. Soc. B: Stat. Methodol.* **68**, 411–436. (doi:10.1111/j.1467-9868.2006.00553.x)
30. Tokdar ST, Kass RE. 2010 Importance sampling: a review. *Wiley Interdiscip. Rev* **2**, 54–60. (doi:10.1002/wics.56)
31. Bürkner PC, Scholz M, Radev ST. 2023 Some models are useful, but how do we know which ones? Towards a unified Bayesian model taxonomy. *Stat. Surv.* **17**, 216–310. (doi:10.1214/23-SS145)
32. Gelman A *et al*. 2020 Bayesian workflow. *arXiv*
33. Gabry J, Simpson D, Vehtari A, Betancourt M, Gelman A. 2019 Visualization in Bayesian workflow. *J. R. Stat. Soc. Ser. A: Stat. Soc.* **182**, 389–402. (doi:10.1111/rssa.12378)
34. Schad DJ, Betancourt M, Vasishth S. 2021 Toward a principled Bayesian workflow in cognitive science. *Psychol. Methods* **26**, 103–126. (doi:10.1037/met0000275)
35. Aguilera-Venegas G, Galán-García JL, Egea-Guerrero R, Galán-García MÁ, Rodríguez-Cielos P, Padilla-Domínguez Y, Galán-Luque M. 2019 A probabilistic extension to Conway's game of life. *Adv. Comput. Math.* **45**, 2111–2121. (doi:10.1007/s10444-019-09696-8)
36. Mikkola P *et al*. 2024 Prior knowledge elicitation: the past, present, and future. *Bayesian Anal.* **19**, 1129–1161. (doi:10.1214/23-BA1381)

37. Bockting F, Radev ST, Bürkner PC. 2024 Simulation-based prior knowledge elicitation for parametric Bayesian models. *Sci. Rep*. **14**, 17330. (doi:10.1038/s41598-024-68090-7)
38. Hartmann M, Agiashvili G, Bürkner P, Klami A. 2020 Flexible prior elicitation via the prior predictive distribution. In *Conference on Uncertainty in Artificial Intelligence*, pp. 1129–1138. PMLR.
39. Mikkola P, Acerbi L, Klami A. 2024 Preferential normalizing flows. *arXiv Preprint* (doi:10.52202/079017-1769)
40. Bockting F, Radev ST, Bürkner PC. 2024 Expert-elicitation method for non-parametric joint priors using normalizing flows. *arXiv*
41. Little RJ. 2006 Calibrated Bayes. *Am. Stat*. **60**, 213–223. (doi:10.1198/000313006x117837)
42. Gneiting T, Balabdaoui F, Raftery AE. 2007 Probabilistic forecasts, calibration and sharpness. J. R. Stat. Soc. Ser. B Stat. Methodol. **69**, 243–268. (doi:10.21236/ada454827)
43. Cook SR, Gelman A, Rubin DB. 2006 Validation of software for Bayesian models using posterior quantiles. *J. Comput. Graph. Stat.* **15**, 675–692. (doi:10.1198/106186006X136976)
44. Talts S, Betancourt M, Simpson D, Vehtari A, Gelman A. 2018 Validating Bayesian inference algorithms with simulation-based calibration. *arXiv*
45. Modrák M, Moon AH, 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 Anal.* **20**, 461–488. (doi:10.1214/23-ba1404)
46. Säilynoja T, Bürkner PC, Vehtari A. 2022 Graphical test for discrete uniformity and its applications in goodness-of-fit evaluation and multiple sample comparison. *Stat. Comput.* **32**, 1–21. (doi:10.1007/s11222-022-10090-6)
47. Yao Y, Domke J. 2023 Discriminative calibration: check Bayesian computation from simulations and flexible classifier. *Adv. Neural Inf. Process. Syst.* **36**, 36106–36131.
48. Bansal V, Chen T, Scott JG. 2025 The surprising strength of weak classifiers for validating neural posterior estimates. *arXiv*
49. Rosenblatt M. 1952 Remarks on a multivariate transformation. *Ann. Math. Stat.* **23**, 470–472.
50. Cohen J. 2013 *Statistical power analysis for the behavioral sciences*, 2nd edn. Oxford, UK: Routledge.
51. Ibragimov IA, Has'minskii RZ. 2013 *Statistical estimation: asymptotic theory*. Berlin, Germany: Springer Science & Business Media.
52. Racine JS, Mackinnon JG. 2007 Simulation-based tests that can use any number of simulations. *Commun. Stat. Simul. Comput*. **36**, 357–365. (doi:10.1080/03610910601161256)
53. Vasishth S, Broe M. 2010 *The foundations of statistics: a simulation-based approach*. Berlin, Heidelberg: Springer. (doi:10.1007/978-3-642-16313-5)
54. Schmitt M, Radev ST, Bürkner PC. 2023 Meta-uncertainty in Bayesian model comparison. In *AISTATS Conference Proceedings*.
55. Dalmasso N, Masserano L, Zhao D, Izbicki R, Lee AB. 2024 Likelihood-free frequentist inference: bridging classical statistics and machine learning for reliable simulator-based inference. *Electron. J. Stat.* **18**, 5045–5090. (doi:10.1214/24-ejs2307)
56. Marin JM, Pudlo P, Robert CP, Ryder RJ. 2012 Approximate Bayesian computational methods. *Stat. Comput*. **22**, 1167–1180. (doi:10.1007/s11222-011-9288-2)
57. Marjoram P, Molitor J, Plagnol V, Tavare S. 2003 Markov Chain Monte Carlo without likelihoods. *Proc. Natl Acad. Sci. USA* **100**, 15324–15328. (doi:10.1073/pnas.0306899100)
58. Picchini U. 2014 Inference for SDE models via approximate Bayesian computation. *J. Comput. Graph. Stat.-Simul. Comput*. **23**, 1080–1100. (doi:10.1080/10618600.2013.866048)
59. Beaumont MA, Cornuet JM, Marin JM, Robert CP. 2009 Adaptive approximate Bayesian computation. *Biometrika* **96**, 983–990. (doi:10.1093/biomet/asp052)
60. Del Moral P, Doucet A, Jasra A. 2012 An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Stat. Comput*. **22**, 1009–1020. (doi:10.1007/s11222-011-9271-y)
61. Picchini U, Tamborrino M. 2024 Guided sequential ABC schemes for intractable Bayesian models. *Bayesian Anal.* **20**, 1–32. (doi:10.1214/24-BA1451)
62. Frazier DT, Martin GM, Robert CP, Rousseau J. 2018 Asymptotic properties of approximate Bayesian computation. *Biometrika* **105**, 593–607. (doi:10.1093/biomet/asy027)

63. von Krause M, Radev ST, Voss A. 2022 Mental speed is high until age 60 as revealed by analysis of over a million participants. *Nat. Hum. Behav.* **6**, 700–708. (doi:10.1038/s41562-021-01282-7)
64. Zeng J, Xue K, Chen H. 2025 Real-time probabilistic model updating and damage detection using machine learning-based likelihood-free inference. *Mech. Syst. Signal Process* **230**, 112612. (doi:10.1016/j.ymssp.2025.112612)
65. Gershman S, Goodman N. 2014 Amortized inference in probabilistic reasoning. In *Proceedings of the annual meeting of the cognitive science society*. vol. 36.
66. Paige B, Wood F. 2016 Inference networks for sequential Monte Carlo in graphical models. In *International Conference on Machine Learning*, pp. 3040–3049. PMLR.
67. Le TA, Baydin AG, Wood F. 2017 Inference compilation and universal probabilistic programming. In *Artificial intelligence and statistics*, pp. 1338–1348. PMLR.
68. Stuhlmüller A, Taylor J, Goodman N. 2013 Learning stochastic inverses. *Adv. Neural Inf. Process. Syst.* **26**.
69. Kingma DP, Welling M. 2013 Auto-encoding variational Bayes. *arXiv*
70. Rezende DJ, Mohamed S, Wierstra D. 2024 Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pp. 1278–1286. PMLR.
71. Zammit-Mangion A, Sainsbury-Dale M, Huser R. 2025 Neural methods for amortized inference. *Annu. Rev. Stat. Its Appl.* **12**, 311–335. (doi:10.1146/annurev-statistics-112723-034123)
72. Hermans J, Begy V, Louppe G. 2020 Likelihood-free MCMC with amortized approximate ratio estimators. In *Int. Conf. on Machine Learning*, pp. 4239–4248. PMLR.
73. Draxler F, Wahl S, Schnörr C, Köthe U. 2024 On the universality of coupling-based normalizing flows. *arXiv*
74. Sharrock L, Simons J, Liu S, Beaumont M. 2022 Sequential neural score estimation: likelihood-free inference with conditional score based diffusion models. *arXiv*
75. Geffner T, Papamakarios G, Mnih A. 2023 Compositional score modeling for simulation-based inference. In *International Conference on Machine Learning*, pp. 11098–11116. PMLR.
76. Gloeckler M, Deistler M, Weilbach C, Wood F, Macke JH. 2024 All-in-one simulation-based inference. *arXiv*
77. Song J, Meng C, Ermon S. 2021 Denoising diffusion implicit models. In *International Conference on Learning Representations*.
78. Albergo MS, Boffi NM, Vanden-Eijnden E. 2023 Stochastic interpolants: a unifying framework for flows and diffusions. *arXiv*
79. Lipman Y, Chen RT, Ben-Hamu H, Nickel M, Le M. 2022 Flow matching for generative modeling. *arXiv*
80. Blum MGB, François O. 2010 Non-linear regression models for approximate Bayesian computation. *Stat. Comput.* **20**, 63–73. (doi:10.1007/s11222-009-9116-0)
81. Papamakarios G, Murray I. 2016 Fast ε-free inference of simulation models with Bayesian conditional density estimation. *Adv. Neural Inform. Process. Syst.* **29**.
82. Gonçalves PJ *et al*. 2020 Training deep neural density estimators to identify mechanistic models of neural dynamics. *eLife* **9**, e56261. (doi:10.7554/eLife.56261)
83. Radev ST, Mertens UK, Voss A, Ardizzone L, Kothe U. 2020 Bayesflow: learning complex stochastic models with invertible neural networks. *IEEE Trans. Neural Networks Learn. Syst.* **33**, 2020. (doi:10.1109/tnnls.2020.3042395)
84. Elsemüller L, Olischläger H, Schmitt M, Bürkner PC, Koethe U, Radev ST. 2024 Sensitivity-aware amortized Bayesian inference. *Trans. Mach. Learn. Res.*
85. Dinh L, Sohl-Dickstein J, Bengio S. 2017 Density estimation using real NVP. In *Int. Conf. on Learning Representations*.
86. Ardizzone L, Kruse J, Rother C, Köthe U. 2018 Analyzing inverse problems with invertible neural networks. In *International Conference on Learning Representations*.
87. Bloem-Reddy B, Teh YW. 2020 Probabilistic symmetries and invariant neural networks. *J. Mach. Learn. Res.* **21**, 1–61.
88. Xu K, Li J, Zhang M, Du SS, Kawarabayashi K i, Jegelka S. 2020 What can neural networks reason about? In *International Conference on Learning Representations*.

89. Avecilla G, Chuong JN, Li F, Sherlock G, Gresham D, Ram Y. 2022 Neural networks enable efficient and accurate simulation-based inference of evolutionary parameters from adaptation dynamics. *PLoS Biol*. **20**, e3001633. (doi:10.1371/journal.pbio.3001633)
90. Dax M *et al*. 2025 Real-time inference for binary neutron star mergers using machine learning. *Nature* **639**, 49–53. (doi:10.1038/s41586-025-08593-z)
91. Schmitt M, Bürkner PC, Köthe U, Radev ST. 2024 Detecting model misspecification in amortized Bayesian inference with neural networks. In *DAGM German Conference on Pattern Recognition*, pp. 541–557. Cham, Switzerland: Springer Nature. (doi:10.1007/978-3-031-54605-1_35)
92. Huang D, Bharti A, Souza A, Acerbi L, Kaski S. 2023 Learning robust statistics for simulation-based inference under model misspecification. *Adv. Neural Inf. Process. Syst*. **36**, 7289–7310.
93. Frazier DT, Kelly R, Drovandi C, Warne DJ. 2024 The statistical accuracy of neural posterior and likelihood estimation. *arXiv*
94. Kallioinen N, Paananen T, Bürkner PC, Vehtari A. 2023 Detecting and diagnosing prior and likelihood sensitivity with power-scaling. *Stat. Comput* **34**, 1–27. (doi:10.1007/s11222-023-10366-5)
95. Li C, Vehtari A, Bürkner PC, Radev ST, Acerbi L, Schmitt M. 2024 Amortized Bayesian workflow. *arXiv*
96. Elsemüller L, Pratz V, Krause M v, Voss A, Bürkner PC, Radev ST. 2025 Does unsupervised domain adaptation improve the robustness of amortized Bayesian inference? A systematic evaluation. *arXiv Preprint*
97. Mishra A, Habermann D, Schmitt M, Radev ST, Bürkner PC. 2025 Robust amortized Bayesian inference with self-consistency losses on unlabeled data. *arXiv Preprint*
98. Swierc P, Tamargo-Arizmendi M, Ćiprijanović A, Nord BD. 2024 Domain-adaptive neural posterior estimation for strong gravitational lens analysis. *arXiv*
99. Gelman A, Hill J. 2006 *Data analysis using regression and multilevel/hierarchical models*. Cambridge, UK: Cambridge University Press. (doi:10.1017/cbo9780511790942)
100. Lawless JF, Fredette M. 2005 Frequentist prediction intervals and predictive distributions. *Biometrika* **92**, 529–542. (doi:10.1093/biomet/92.3.529)
101. Säilynoja T, Schmitt M, Bürkner PC, Vehtari A. 2025 Posterior SBC: simulation-based calibration checking conditional on data. *arXiv*
102. Fazio L, Scholz M, Bürkner PC. 2024 Primed priors for simulation-based validation of Bayesian models. *arXiv*
103. Kass RE, Raftery AE. 1995 Bayes factors. *J. Am. Stat. Assoc*. **90**, 773–795. (doi:10.1080/01621459.1995.10476572)
104. Radev ST, D'Alessandro M, Mertens UK, Voss A, Kothe U, Burkner PC. 2023 Amortized Bayesian model comparison with evidential deep learning. *IEEE Trans. Neural Networks Learn. Syst*. **34**, 4903–4917. (doi:10.1109/TNNLS.2021.3124052)
105. Schad DJ, Nicenboim B, Bürkner PC, Betancourt M, Vasishth S. 2023 Workflow techniques for the robust use of Bayes factors. *Psychol. Methods* **28**, 1404–1426. (doi:10.1037/met0000472)
106. Oelrich O, Ding S, Magnusson M, Vehtari A, Villani M. 2020 When are Bayesian model probabilities overconfident? *arXiv*
107. Aguilar JE, Bürkner PC. 2023 Intuitive joint priors for Bayesian linear multilevel models: the R2D2M2 prior. *Electron. J. Stat* **17**, 1711–1767. (doi:10.1214/23-EJS2136)
108. Dellaportas P, Forster JJ, Ntzoufras I. 2012 Joint specification of model space and parameter space prior distributions. *Stat. Sci* **27**, 232–246. (doi:10.1214/11-sts369)
109. Van Dongen S. 2006 Prior specification in Bayesian statistics: three cautionary tales. *J. Theor. Biol*. **242**, 90–100. (doi:10.1016/j.jtbi.2006.02.002)
110. Zitzmann S, Helm C, Hecht M. 2021 Prior specification for more stable Bayesian estimation of multilevel latent variable models in small samples: a comparative investigation of two different approaches. *Front. Psychol*. **11**, 611267. (doi:10.3389/fpsyg.2020.611267)
111. Lindley DV. 1957 A statistical paradox. *Biometrika* **44**, 187–192.
112. Vehtari A, Gelman A, Gabry J. 2017 Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Stat. Comput*. **27**, 1413–1432. (doi:10.1007/s11222-016-9696-4)
113. Vehtari A, Simpson DP, Yao Y, Gelman A. 2019 Limitations of 'limitations of Bayesian leave-one-out cross-validation for model selection'. *Comput. Brain Behav*. **2**, 22–27. (doi:10.1007/s42113-018-0020-6)

114. Vehtari A, Simpson D, Gelman A, Yao Y, Gabry J. 2024 Pareto smoothed importance sampling. *J. Mach. Learn. Res.* **25**, 1–58.
115. Radev ST, Schmitt M, Pratz V, Picchini U, Köthe U, Bürkner PC. 2023 JANA: Jointly amortized neural approximation of complex Bayesian models (eds RJ Evans, I Shpitser). In *Proceedings of the thirty-ninth conference on uncertainty in artificial intelligence, Proceedings of machine learning research*, vol. 216, pp. 1695–1706, PMLR.
116. Berger JO *et al*. 1994 An overview of robust Bayesian analysis. *Test.* **3**, 5–124.
117. Wagenmakers EJ, Sarafoglou A, Aczel B. 2022 One statistical analysis must not rule them all. *Nature* **605**, 423–425. (doi:10.1530/ey.19.15.12)
118. Borges JL. 1941 *El jardín de senderos que se bifurcan* (Garden of forking paths). Editorial Sur.
119. McElreath R. 2020 *Statistical rethinking: a Bayesian course with examples in R and Stan*, 2nd edn. London, UK: CRC Press, Taylor & Francis Group.
120. Merkle EC, Furr D, Rabe-Hesketh S. 2019 Bayesian comparison of latent variable models: conditional versus marginal likelihoods. *Psychometrika* **84**, 802–829. (doi:10.1007/s11336-019-09679-0)
121. South LF, Oates CJ, Mira A, Drovandi C. 2023 Regularized zero-variance control variates. *Bayesian Anal* **18**, 865–888. (doi:10.1214/22-ba1328)
122. Robert CP, Roberts GO. 2021 Rao-Blackwellization in the MCMC era. *arXiv*