SBC-SBI Thesis

Kai Samaroo

January 19, 2026

1 Introduction and Notation

Let $\boldsymbol{\theta} \in \Theta$ denote the model parameters, where Θ is the parameter space, and let $\boldsymbol{x} \in X$ denote the observed data, taking values in the data space X. Given X and Θ, we can specify a Bayesian model π by defining a prior distribution $\pi_{\text{prior}}(\boldsymbol{\theta})$ over parameters $\boldsymbol{\theta} \in \Theta$ and a likelihood $\pi_{\text{likelihood}}(\boldsymbol{x}|\boldsymbol{\theta})$ that specifies the probability distribution over data $\boldsymbol{x} \in X$ given a fixed parameter setting $\boldsymbol{\theta} \in \Theta$. We can think of each $\boldsymbol{\theta}$ as a "state of the universe" and any $\boldsymbol{x} \sim \pi_{\text{likelihood}}(\boldsymbol{x}|\boldsymbol{\theta})$ as a realization of some process given that the universe is in state $\boldsymbol{\theta}$. The prior encodes our beliefs about the parameter $\boldsymbol{\theta}$ (i.e. the "state of the universe") before observing any data. The prior and likelihood define a joint distribution over $(\boldsymbol{x}, \boldsymbol{\theta})$ given by

$$\pi_{\text{joint}}(\boldsymbol{x}, \boldsymbol{\theta}) = \pi_{\text{likelihood}}(\boldsymbol{x}|\boldsymbol{\theta})\pi_{\text{prior}}(\boldsymbol{\theta})$$

as well as a posterior distribution given by

$$\pi_{\text{posterior}}(\boldsymbol{\theta}|\boldsymbol{x}) = \frac{\pi_{\text{joint}}(\boldsymbol{x}, \boldsymbol{\theta})}{\pi_{\text{marginal}}(\boldsymbol{x})} = \frac{\pi_{\text{prior}}(\boldsymbol{\theta})\pi_{\text{likelihood}}(\boldsymbol{x}|\boldsymbol{\theta})}{\pi_{\text{marginal}}(\boldsymbol{x})}$$

where the marginal distribution over $\boldsymbol{x}$ is given by

$$\pi_{\text{marginal}}(\boldsymbol{x}) = \int \pi_{\text{likelihood}}(\boldsymbol{x}|\boldsymbol{\theta})\pi_{\text{prior}}(\boldsymbol{\theta})d\boldsymbol{\theta}.$$

The goal of Bayesian inference is to infer the parameters $\boldsymbol{\theta}$ based on observations of data $\boldsymbol{x}$. Returning to our analogy, we wish to infer which universe $\boldsymbol{\theta}$ our observations $\boldsymbol{x}$ came from. This inference is done by examining the posterior distribution $\pi_{\text{posterior}}(\boldsymbol{\theta}|\boldsymbol{x})$, which, in all but the simplest models, is analytically intractable and must be approximated using a computational inference method (e.g. Markov Chain Monte Carlo, Variational Inference, Neural Posterior Estimation, ...). Let $\tilde{\pi}_{\text{posterior}}(\boldsymbol{\theta}|\boldsymbol{x})$ denote the approximation to the posterior distribution $\pi_{\text{posterior}}(\boldsymbol{\theta}|\boldsymbol{x})$ produced by the computational inference method under consideration. Whenever it is clear to do so, we simplify notation by omitting subscripts and writing $\pi_{\text{prior}}(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta})$, $\pi_{\text{likelihood}}(\boldsymbol{x}|\boldsymbol{\theta}) = \pi(\boldsymbol{x}|\boldsymbol{\theta})$, $\pi_{\text{marginal}}(\boldsymbol{x}) = \pi(\boldsymbol{x})$, $\pi_{\text{joint}}(\boldsymbol{x}, \boldsymbol{\theta}) = \pi(\boldsymbol{x}, \boldsymbol{\theta})$, $\pi_{\text{posterior}}(\boldsymbol{\theta}|\boldsymbol{x}) = \pi(\boldsymbol{\theta}|\boldsymbol{x})$, and $\tilde{\pi}_{\text{posterior}}(\boldsymbol{\theta}|\boldsymbol{x}) = \tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$ since it is clear from looking at the arguments which density we are referring to.

2 Simulation-Based Calibration

2.1 Related Work:

Geweke [2] seems to be the first ever paper introducing a validation algorithm for Bayesian computational methods. More recently, Cook et al. [1] proposes a simulation-based method for validating Bayesian computational software. The latter is essentially the same as the well-known SBC method from Talts et al. [5], but slightly differs in the following ways:

- This paper normalizes the rank statistics to make them into empirical CDFs.
- This paper does not explicitly mention that we can calculate ranks for any test function $f : \Theta \times X \mapsto \mathbb{R}$, but they use both projections $f(\boldsymbol{\theta}) = \theta_i$ and some simple test functions (e.g. averages of batches of elements of $\boldsymbol{\theta}$).

The main motivation behind this paper seems to be checking for typos in the approximate posterior sampling algorithm, as opposed to testing whether or not the approximate posterior is actually a good approximation to the true posterior. However, of course, these validation methods can be used for both, and in modern-day Bayesian inference the focus seems to have shifted to the latter.

2.2 Simulation-Based Calibration In Theory

Much of the theory developed in this section was taken from Modrák et al. [3]. We begin by introducing the notions of approximate posterior correctness in Definition 2.1, and ties in Definition 2.2.

Definition 2.1 (Correctness of approximate posterior)**.** *Assume a data space X, a parameter space Θ, a Bayesian model π on (X, Θ), and an approximate posterior $\tilde{\pi}$. We say that the approximate posterior $\tilde{\pi}$ is correct, denoted $\tilde{\pi} \equiv \pi$, if it equals the true posterior except for a set of measure zero:*

$$\int_X \int_\Theta \mathbb{I}\left[\pi(\boldsymbol{\theta}|\boldsymbol{x}) \neq \tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})\right] \pi(\boldsymbol{x}, \boldsymbol{\theta}) d\boldsymbol{x} d\boldsymbol{\theta} = 0$$

Intuitively, an approximate posterior is considered correct if it yields the same inferences as the true posterior. This notion aligns with the practical goals of Bayesian inference, where discrepancies on sets of measure zero are not a concern.

Definition 2.2 (Ties)**.** *Assume a data space X, a parameter space Θ, a Bayesian model π on (X, Θ), and a test function $f : X \times \Theta \mapsto \mathbb{R}$. We say that the model π has ties with respect to f if there exists some $(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \in X \times \Theta$ such that*

$$\mathbb{P}_{\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}|\boldsymbol{x})} \Big(f(\boldsymbol{x}, \boldsymbol{\theta}) = f(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \Big) > 0$$

Furthermore, we say that an approximate posterior $\tilde{\pi}$ has ties with respect to f if there exists some $(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \in X \times \Theta$ such that

$$\mathbb{P}_{\boldsymbol{\theta} \sim \tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})} \Big(f(\boldsymbol{x}, \boldsymbol{\theta}) = f(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \Big) > 0$$

We restrict our attention to the case where nether π nor $\tilde{\pi}$ have ties with respect to f, which is often the case in practice (e.g. in fully continuous models). Moreoever, Lemma 1 from Appendix A in Modrák et al. [3] shows that any setting in which π and/or $\tilde{\pi}$ do have ties can be mapped to an equivalent setting with no ties. Thus, the theory is invariant to the presence of ties.
The SBC algorithm involves sampling from the prior, so $\pi(\boldsymbol{\theta})$ must be a proper prior distribution. Furthermore, the samples from the approximate posterior $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$ must be (at least approximately) independent. In the case of correlated MCMC samplers, this can be achieved by sufficient thinning or by using sampling schemes that yield effectively uncorrelated draws.

Remark 2.1. *The following sampling methods are equivalent (i.e. they generate equidistributed $\boldsymbol{x}, \tilde{\boldsymbol{\theta}}$):*

(A) $\tilde{\boldsymbol{\theta}} \sim \pi(\boldsymbol{\theta})$ *followed by* $\boldsymbol{x} \sim \pi(\boldsymbol{x}|\tilde{\boldsymbol{\theta}})$

(B) $(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \sim \pi(\boldsymbol{x}, \boldsymbol{\theta})$

(C) $\boldsymbol{x} \sim \pi(\boldsymbol{x})$ *followed by* $\tilde{\boldsymbol{\theta}} \sim \pi(\boldsymbol{\theta}|\boldsymbol{x})$

The equivalence (A) $\iff$ (C) will help build intuition as to why Simulation-Based Calibration (Algorithm 1) is useful for testing approximate posteriors.

The Simulation-Based Calibration (SBC) algorithm proceeds as follows:

Algorithm 1: SBC Algorithm

Input: Prior $\pi(\boldsymbol{\theta})$, likelihood sampler $\pi(\boldsymbol{x}|\tilde{\boldsymbol{\theta}})$, posterior approximation $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$, test function $f : X \times \Theta \mapsto \mathbb{R}$, number of SBC iterations N_{iter}, number of samples per SBC iteration N_{samp}.
Output: i.i.d rank statistics $\{\hat{R}_1, ..., \hat{R}_{N_{\text{iter}}}\}$.

for $n = 1$ **to** N_{iter} **do**
  $\tilde{\boldsymbol{\theta}}_n \sim \pi(\boldsymbol{\theta})$
  $\boldsymbol{x}_n \sim \pi(\boldsymbol{x}|\tilde{\boldsymbol{\theta}}_n)$
  $\boldsymbol{\theta}_n^{(1)}, \boldsymbol{\theta}_n^{(2)}, ..., \boldsymbol{\theta}_n^{(N_{\text{samp}})} \sim \tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x}_n)$
  $\hat{R}_n = \frac{1}{N_{\text{samp}}} \sum_{k=1}^{N_{\text{samp}}} \mathbb{I}\{f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(k)}) < f(\boldsymbol{x}_n, \tilde{\boldsymbol{\theta}}_n)\}$
end

If $\tilde{\pi} \equiv \pi$ (i.e. if the approximate posterior is correct) then Remark 2.1 allows us to view any iteration n of Algorithm 1 as first sampling $\boldsymbol{x}_n \sim \pi(\boldsymbol{x})$ and then sampling $\tilde{\boldsymbol{\theta}}_n, \boldsymbol{\theta}_n^{(1)}, \boldsymbol{\theta}_n^{(2)} ..., \boldsymbol{\theta}_n^{(N_{\text{samp}})} \overset{i.i.d}{\sim} \pi(\boldsymbol{\theta}|\boldsymbol{x}_n)$. Thus, the samples $\tilde{\boldsymbol{\theta}}_n, \boldsymbol{\theta}_n^{(1)}, \boldsymbol{\theta}_n^{(2)} ..., \boldsymbol{\theta}_n^{(N_{\text{samp}})}$ will be indistinguishable i.i.d samples from the true posterior $\pi(\boldsymbol{\theta}|\boldsymbol{x}_n)$ if $\tilde{\pi} \equiv \pi$. In particular, the test functions $f(\boldsymbol{x}_n, \tilde{\boldsymbol{\theta}}_n)$, $f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(1)})$, $f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(2)})$, ..., $f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(N_{\text{samp}})})$ will be indistinguishable and therefore the (normalized) rank $\hat{R}_n$ of $f(\boldsymbol{x}_n, \tilde{\boldsymbol{\theta}}_n)$ in $f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(1)}), f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(2)}), ..., f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(N_{\text{samp}})})$ will be uniformly distributed on $\{0, \frac{1}{N_{\text{samp}}}, \frac{2}{N_{\text{samp}}}, ..., \frac{N_{\text{samp}}-1}{N_{\text{samp}}}, 1\}$. We formalize this intuition in Theorem 2.1 and Corollary 2.1.

Theorem 2.1 (Uniformity of rank statistic)**.** *Assume a data space X, a parameter space Θ, a Bayesian model π on (X, Θ), a test function $f : X \times \Theta \mapsto \mathbb{R}$, and let $N \in \mathbb{N}$. Then the rank statistic $\hat{R}$ defined by the process*

$$\tilde{\boldsymbol{\theta}} \sim \pi(\boldsymbol{\theta})$$

$$\boldsymbol{x} \sim \pi(\boldsymbol{x}|\tilde{\boldsymbol{\theta}})$$

$$\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, ..., \boldsymbol{\theta}^{(N)} \sim \pi(\boldsymbol{\theta}|\boldsymbol{x})$$

$$\hat{R} := \frac{1}{N} \sum_{k=1}^{N} \mathbb{I}\{f(\boldsymbol{x}, \boldsymbol{\theta}^{(k)}) < f(\boldsymbol{x}, \tilde{\boldsymbol{\theta}})\}$$

is discretely uniformly distributed as follows:

$$\hat{R} \sim \mathit{Unif}\Big\{0, \frac{1}{N}, \frac{2}{N}, ..., \frac{N-1}{N}, 1\Big\}$$

Equivalently, the true posterior π passes (N-sample) theoretical SBC (see Definition 2.3).

Proof. It clearly suffices to show $N\hat{R} \sim \text{Unif}\{0, 1, 2, ..., N\}$. Observe that $\hat{R}$ is a random variable dependent on the random variables $\tilde{\boldsymbol{\theta}}, \boldsymbol{x}, \boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, ..., \boldsymbol{\theta}^{(N)}$. Furthermore, the random variable $N\hat{R}|(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) = \sum_{k=1}^{N} \mathbb{I}\{f(\boldsymbol{x}, \boldsymbol{\theta}^{(k)}) < f(\boldsymbol{x}, \tilde{\boldsymbol{\theta}})\}$ follows a $\text{Bin}(N, u)$ distribution, since it is the sum of N i.i.d Bernoulli(u) random variables, where

$$u|(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) = \mathbb{P}_{\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}|\boldsymbol{x})} \Big(f(\boldsymbol{x}, \boldsymbol{\theta}) < f(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \Big) = \mathbb{P}_{\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}|\boldsymbol{x})} \Big(f(\boldsymbol{x}, \boldsymbol{\theta}) \leq f(\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \Big)$$

with the last equality holding by the assumption that π has no ties w.r.t f. Note that u is itself a random variable dependent on the random variables $(\boldsymbol{x}, \tilde{\boldsymbol{\theta}})$ and, furthermore, $u|\boldsymbol{x} \sim \text{Unif}[0,1]$ by the probability integral transform since $\tilde{\boldsymbol{\theta}} \sim \pi(\boldsymbol{\theta}|\boldsymbol{x})$. For any $j \in \{0, 1, 2, ..., N\}$, we can apply the law of total probability by conditioning on the value of u

$$\mathbb{P}_{\tilde{\boldsymbol{\theta}},\boldsymbol{x},\boldsymbol{\theta}^{(1)},\boldsymbol{\theta}^{(2)},...,\boldsymbol{\theta}^{(N)}}\left(N\hat{R} = j\right) = \int_{u=0}^{1} \underbrace{\mathbb{P}_{\boldsymbol{\theta}^{(1)},\boldsymbol{\theta}^{(2)},...,\boldsymbol{\theta}^{(N)}}\left(N\hat{R} = j \middle| u = u\right)}_{\text{pmf of Bin}(N,u)} \underbrace{p_{\text{Unif}[0,1]}(u)}_{\equiv 1} du$$

$$= \int_{u=0}^{1} \binom{N}{j} u^j (1-u)^{N-j} du$$

where, after integrating out u, we dropped the subscripts $\tilde{\boldsymbol{\theta}}, \boldsymbol{x}$ from $\mathbb{P}$ inside the integral since $\hat{R} \perp\!\!\!\perp (\boldsymbol{x}, \tilde{\boldsymbol{\theta}}) \mid u$ (i.e. $\hat{R}$ only depends on $(\boldsymbol{x}, \tilde{\boldsymbol{\theta}})$ through u). The integral over u can be expressed in terms of the beta function $\text{B}(\cdot, \cdot)$

$$\int_{u=0}^{1} \binom{N}{j} u^j (1-u)^{N-j} du = \binom{N}{j} \int_{u=0}^{1} u^j (1-u)^{N-j} du$$

$$= \binom{N}{j} \text{B}(j+1, N-j+1) = \frac{\Gamma(j+1)\Gamma(N-j+1)}{\Gamma(N+2)}$$

$$= \frac{N!}{j!(N-j)!} \frac{j!(N-j)!}{(N+1)!}$$

$$= \frac{1}{N+1}$$

where the third to last equality used the fact that $\text{B}(z_1, z_2) = \frac{\Gamma(z_1)\Gamma(z_2)}{\Gamma(z_1+z_2)}$ and the second to last equality used the fact that $\Gamma(n) = (n-1)!$ for $n \in \mathbb{N}$. The derived pmf of $N\hat{R}$ is precisely the pmf of a discrete uniform distribution on $\{0, 1, ..., N\}$, completing the proof. □

Corollary 2.1. *Assume a data space X, a parameter space Θ, a Bayesian model π on (X, Θ), a test function $f : X \times \Theta \mapsto \mathbb{R}$, and an approximate posterior $\tilde{\pi}$. If $\tilde{\pi} \equiv \pi$, then the rank statistics $\hat{R}_1, ..., \hat{R}_{N_{iter}}$ from Algorithm 1 are i.i.d discretely uniformly distributed:*

$$\hat{R}_1, ..., \hat{R}_{N_{iter}} \overset{i.i.d}{\sim} Unif\left\{0, \frac{1}{N_{samp}}, \frac{2}{N_{samp}}, ..., \frac{N_{samp}-1}{N_{samp}}, 1\right\}$$

Proof. The ranks are i.i.d for any proposal posterior by construction (see Algorithm 1). Since $\tilde{\pi} = \pi$, Theorem 2.1 with $N = N_{\text{samp}}$ guarantees that they are discretely uniformly distributed. □

Theorem 2.1 and Corollary 2.1 motivate us to define what it means for an approximate posterior $\tilde{\pi}$ to pass theoretical SBC.

Definition 2.3 (Theoretical SBC pass)**.** *Assume a data space X, a parameter space Θ, a Bayesian model π on (X, Θ), a test function $f : X \times \Theta \mapsto \mathbb{R}$, an approximate posterior $\tilde{\pi}$, and let $N \in \mathbb{N}$. We say $\tilde{\pi}$ passes N-sample theoretical SBC with respect to f if the rank statistic $\hat{R}$ defined by the process*

$$\tilde{\boldsymbol{\theta}} \sim \pi(\boldsymbol{\theta})$$

$$\boldsymbol{x} \sim \pi(\boldsymbol{x}|\tilde{\boldsymbol{\theta}})$$

$$\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, ..., \boldsymbol{\theta}^{(N)} \sim \tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$$

$$\hat{R} := \frac{1}{N} \sum_{k=1}^{N} \mathbb{I}\{f(\boldsymbol{x}, \boldsymbol{\theta}^{(k)}) < f(\boldsymbol{x}, \tilde{\boldsymbol{\theta}})\}$$

is discretely uniformly distributed on $\left\{0, \frac{1}{N}, \frac{2}{N}, ..., \frac{N-1}{N}, 1\right\}$. Furthermore, we say that $\tilde{\pi}$ passes theoretical SBC with respect to f if there exists some $N \in \mathbb{N}$ such that $\tilde{\pi}$ passes N-sample theoretical SBC with respect to f.

Remark 2.2. *We use the term "theoretical SBC pass" in Definition 2.3 because, in practice, we can never be sure whether or not the rank statistic is truly uniform. Instead, we use Algorithm 1 to generate a large number of independent samples of the rank statistic $\hat{R}_1, ..., \hat{R}_{N_{iter}}$ and perform a hypothesis test under the null hypothesis that the samples are uniform. In principle, N_{iter} should be as large as possible to reduce sampling error and increase the power of the test, but large values of N_{iter} can be computationally prohibitive for some SBI methods (e.g. SNPE variants).*

Theorem 2.1 guarantees that if an approximate posterior $\tilde{\pi}$ fails theoretical SBC w.r.t some test function f, then we can conclude with certainty that $\tilde{\pi} \not\equiv \pi$; that is, failure of theoretical SBC implies that the approximate posterior is incorrect. The original paper proposing SBC (Talts et al. [5]) only considers parameter-dependent test functions $f : \Theta \to \mathbb{R}$; that is, test functions of the form $f(\boldsymbol{x}, \boldsymbol{\theta}) = f(\boldsymbol{\theta})$ that do not depend on the data $\boldsymbol{x}$. In this setting, the converse does not hold: if $\tilde{\pi}$ passes theoretical SBC w.r.t some test function $f : \Theta \to \mathbb{R}$, this does not imply that $\tilde{\pi} \equiv \pi$. In other words, passing theoretical SBC with respect to a data-independent test function does not guarantee that the approximate posterior is correct. Theorem 2.2 provides the canonical counterexample by proving that the prior passes theoretical SBC for any data-independent test function.

Theorem 2.2 (Prior passes theoretical SBC w.r.t any data-independent test function)**.** *Assume a data space X, a parameter space Θ, a Bayesian model π on (X, Θ), and a data-independent test function $f : \Theta \mapsto \mathbb{R}$. Then the prior passes theoretical SBC with respect to f.*

Proof. Let $N \in \mathbb{N}$. We can proceed analogously to the proof of Theorem 2.1 using the prior $\pi(\boldsymbol{\theta})$ in place of the posterior $\pi(\boldsymbol{\theta}|\boldsymbol{x})$. In this case, the samples $\tilde{\boldsymbol{\theta}}, \boldsymbol{\theta}^{(1)}, ..., \boldsymbol{\theta}^{(N)}$ are i.i.d from the prior, and we simply ignore the data $\boldsymbol{x}$ generated from the likelihood simulator. We have $N\hat{R}|\tilde{\boldsymbol{\theta}} \sim \text{Bin}(N, u)$ where $u|\tilde{\boldsymbol{\theta}} = \mathbb{P}_{\boldsymbol{\theta}\sim\pi(\boldsymbol{\theta})}\left(f(\boldsymbol{\theta}) \leq f(\tilde{\boldsymbol{\theta}})\right)$, and all remaining steps of the proof carry over from 2.1 without modification. We conclude that the prior passes N-sample theoretical SBC with respect to f. □

Motivated by this counterexample, Modrák et al. [3] proposed the use of data-dependent test statistics $f : X \times \Theta \to \mathbb{R}$, with the aim of shrinking the set of distributions that pass theoretical SBC by introducing data dependence. Ideally, we would like a test function f for which only the true posterior passes theoretical SBC with respect to f, since this would make passing theoretical SBC equivalent to being equal to the true posterior. Surprisingly, Theorem 2.3 below shows that we can always find such an "oracle" test function.

Theorem 2.3 (Oracle test function)**.** *Assume a data space X, a parameter space Θ, a Bayesian model π on (X, Θ), an approximate posterior $\tilde{\pi}$, and let $N \in \mathbb{N}$. Define the oracle test function $f^* : X \times \Theta \to \mathbb{R}$ as*

$$f^*(\boldsymbol{x}, \boldsymbol{\theta}) := \frac{\pi(\boldsymbol{\theta}|\boldsymbol{x})}{\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})}$$

Then $\tilde{\pi}$ passes N-sample theoretical SBC with respect to f^* if and only if $\tilde{\pi} \equiv \pi$.

Proof. The proof of this theorem is omitted since it is tedious and depends on several technical lemmas. We refer the reader to Theorem 6 in Appendix A of Modrák et al. [3], where a full proof is provided. □

Theorem 2.3 guarantees that for any approximate posterior $\tilde{\pi}$, we can always find a test function f^* such that $\tilde{\pi}$ passing theoretical SBC on f^* is equivalent to $\tilde{\pi} \equiv \pi$. As expected, this is too good to be true since the oracle test function f^* requires knowledge of the true posterior—precisely the distribution that is unknown in practice and that Bayesian inference seeks to approximate. However, Theorem 2.3 is evidence that test statistics with a non-trivial dependence on the data $\boldsymbol{x}$ tend to be stronger than those that ignore data in the sense that they admit a smaller class of distributions that pass theoretical SBC. Thus, when using theoretical SBC to analyze a computational inference method $\tilde{\pi}$, practitioners who observe many SBC passes should consider employing data-dependent test functions to further scrutinize the method and gain greater confidence in the quality of the posterior approximation.

2.3 Simulation-Based Calibration In Practice

As mentioned in Remark 2.2, we can never be sure whether or not the rank statistic is truly uniform. Equivalently, we can never be sure whether or not the approximate posterior passes theoretical SBC. Instead, we use Algorithm 1 to generate i.i.d samples of the rank statistic $\hat{R}_1, ..., \hat{R}_{N_{\text{iter}}}$ and perform a hypothesis test under the null hypothesis that the samples are uniform. This motivates us to define an SBC pass in Definition 2.4

Definition 2.4 (SBC pass)**.** *Assume a data space X, a parameter space Θ, a Bayesian model π on (X, Θ), a test function $f : X \times \Theta \mapsto \mathbb{R}$, an approximate posterior $\tilde{\pi}$, and let $N_{samp}, N_{iter} \in \mathbb{N}$. We say $\tilde{\pi}$ passes (N_{samp}, N_{iter})-sample SBC with respect to f if the rank statistics $\hat{R}_1, ..., \hat{R}_{N_{iter}}$ from Algorithm 1 pass a prespecified hypothesis test for uniformity. Furthermore, we say that $\tilde{\pi}$ passes SBC with respect to f if there exist some $N_{samp}, N_{iter} \in \mathbb{N}$ such that $\tilde{\pi}$ passes (N_{samp}, N_{iter})-sample SBC with respect to f.*

Definition 2.4 uses the phrase "hypothesis test" loosely; formal hypothesis tests in the traditional sense (i.e. predefine a significance level, compute a p-value, and accept or reject the null) are seldom used to test the uniformity of SBC ranks in practice. While one can, in principle, formally test the SBC rank distribution using standard uniformity tests, such tests are often overly sensitive for large sample sizes N_{iter}. In particular, even state-of-the-art posterior inference methods inevitably produce approximations that differ slightly from the true posterior, differences that are typically negligible for practical inference. Nevertheless, these differences will propagate to the rank statistics[1], meaning that, for large N_{iter}, these tests could have high enough power to detect even these minor deviations from uniformity. This can lead to an otherwise high-quality posterior approximation failing SBC despite being entirely adequate for the applied inference problem at hand. Moreover, formal hypothesis tests are insufficiently informative about how an approximation fails, whereas visual inspection of SBC rank histograms provide immediate and interpretable diagnostic information about the posterior approximation (see Section 2.3.1).

2.3.1 The SBC histogram

Let N_{samp} denote the number of samples taken from the approximate posterior $\tilde{\pi}$ inside the calculation of a single rank (this notation coincides with Algorithm 1). Since the empirical SBC ranks $\hat{R}_1, ..., \hat{R}_{N_{\text{iter}}}$ lie in the discrete space $\{0, \frac{1}{N_{\text{samp}}}, ..., \frac{N_{\text{samp}}-1}{N_{\text{samp}}}, 1\}$, a natural and intuitive way to visualize their distribution would be via a bar plot[2]. However, in practice, N_{samp} is typically chosen to be very large (e.g. 10,000) in order to reduce sampling error. As a result, the number of bins becomes too large to render on a plot, and with high probability many bins contain no observations, making such bar plots difficult to interpret and visually uninformative. This issue is caused by the rank statistics lying on a grid that has been discretized so finely that it is essentially continuous. Thus, it is reasonable to make the approximation

$$\left\{0, \frac{1}{N_{\text{samp}}}, \frac{2}{N_{\text{samp}}}, ..., \frac{N_{\text{samp}} - 1}{N_{\text{samp}}}, 1\right\} \approx [0, 1]$$

and visualize the ranks as if they had continuous support $[0, 1]$. The most natural way to visualize such data is using a histogram, and this is the main visualization method employed in Talts et al. [5]. The SBC histogram is

[1] Here, we assume that the posterior approximation does not erroneously pass theoretical SBC (Definition 2.3). For example, the prior passes theoretical SBC for data-independent test functions (Theorem 2.2). While this phenomenon has not been studied in detail, this suggests that test functions can be "tricked" into passing theoretical SBC by certain pathological posterior approximations.

[2] In this case, a bar plot divides the horizontal axis into exactly $N_{\text{samp}} + 1$ bins, representing each of the possible rank values $0, \frac{1}{N_{\text{samp}}}, ..., \frac{N_{\text{samp}}-1}{N_{\text{samp}}}, 1$. A vertical bar is placed above each bin whose height equals the number of ranks that take that bin's value.

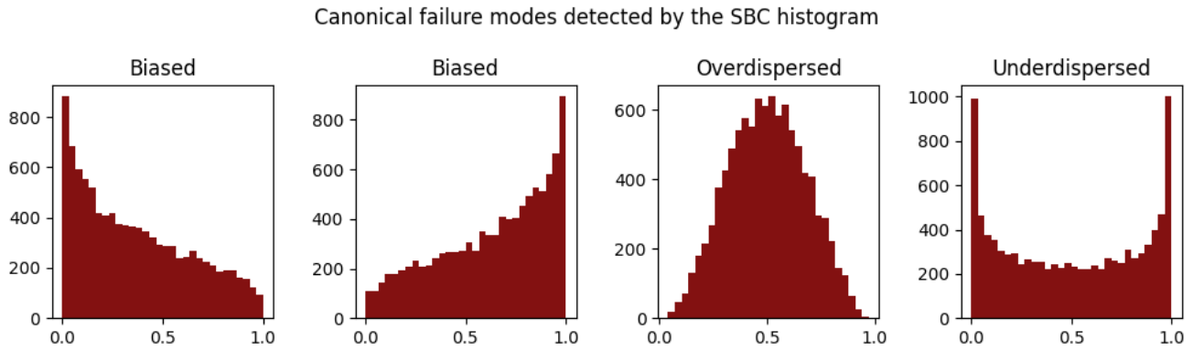

Figure 1: Canonical failure modes detected by the SBC histogram. The two leftmost plots represent biased posterior approximations. Note that whether the rank histogram is left- or right- skewed depends on the test function. The third plot represents an over-dispersed posterior approximation, and the fourth plot represents an under-dispersed posterior approximation.

especially useful since, beyond flagging posterior approximations as incorrect, its shape provides insight into their failure modes. In particular, it can detect the two primary failure modes of posterior inference algorithms: bias and under- or over-dispersion (see below).

Analyzing the SBC histogram: Consider a biased posterior inference algorithm that generates an approximate posterior $\tilde{\pi}$ whose mean is significantly different from that of the true posterior, meaning $||\mathbb{E}_{\boldsymbol{\theta}\sim\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})}[\boldsymbol{\theta}] - \mathbb{E}_{\boldsymbol{\theta}\sim\pi(\boldsymbol{\theta}|\boldsymbol{x})}[\boldsymbol{\theta}]||$ is large for all[3] $\boldsymbol{x} \in X$ (where $||\cdot||$ denotes some distance measure on Θ). Recall that, at any round n of Algorithm 1, the parameter sample $\tilde{\boldsymbol{\theta}}_n$ is from $\pi(\boldsymbol{\theta})$, and the data sample $\boldsymbol{x}_n$ is from $\pi(\boldsymbol{x}|\tilde{\boldsymbol{\theta}}_n)$. Using Remark 2.1, this is equivalent to first sampling $\boldsymbol{x}_n$ from $\pi(\boldsymbol{x})$, followed by $\tilde{\boldsymbol{\theta}}_n$ from the true posterior conditioned on $\boldsymbol{x}_n$, $\pi(\boldsymbol{\theta}|\boldsymbol{x}_n)$. We then sample $\boldsymbol{\theta}_n^{(1)}, ..., \boldsymbol{\theta}_n^{(N_{\text{samp}})}$ from the approximate posterior conditioned on $\boldsymbol{x}_n$, $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x}_n)$. Since $\tilde{\pi}$ is biased, the N_{samp} samples $\boldsymbol{\theta}_n^{(1)}, ..., \boldsymbol{\theta}_n^{(N_{\text{samp}})}$ from $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x}_n)$ are likely to be considerably different from the sample $\tilde{\boldsymbol{\theta}}_n$ from the true posterior $\pi(\boldsymbol{\theta}|\boldsymbol{x}_n)$. Applying the test function to these samples will result in $f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(1)}), ..., f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(N_{\text{samp}})})$ being considerably different from $f(\boldsymbol{x}_n, \tilde{\boldsymbol{\theta}}_n)$. These test function values are ordered since they live on the real line $\mathbb{R}$, meaning that the aforementioned difference will result in $f(\boldsymbol{x}_n, \tilde{\boldsymbol{\theta}}_n)$ lying above/below[4] the cluster of points $f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(1)}), ..., f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(N_{\text{samp}})})$ - the former would lead to a rank near 1 and the latter would lead to a rank near 0. This results in a skewed SBC histogram, with ranks either pushed towards 0 (first plot in Figure 1) or 1 (second plot in Figure 1) depending on the test function f. Arguing in an analogous way, the SBC histogram will be $\cap$-shaped (rsp. $\cup$-shaped) if the approximate posterior is over- (rsp. under-) dispersed relative to the true posterior, as illustrated in the third (rsp. fourth) plot in Figure 1. Importantly, SBC histograms are capable of detecting combinations of these failure modes at once, such cases manifest themselves with histograms that exhibit characteristics from a mixture of plots in Figure 1.

2.3.2 The SBC Empirical CDF Plot

[3]By construction, SBC only tests the validity of posterior approximations $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$ for $\boldsymbol{x}$ that are plausible under the marginal $\pi(\boldsymbol{x})$. Differences between $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$ and $\pi(\boldsymbol{\theta}|\boldsymbol{x})$ for $\boldsymbol{x}$ that are not plausible under $\boldsymbol{x}$ are not detected by SBC.

[4]Whether $f(\boldsymbol{x}_n, \tilde{\boldsymbol{\theta}}_n)$ consistently lies above or below the cluster of points $f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(1)}), ..., f(\boldsymbol{x}_n, \boldsymbol{\theta}_n^{(N_{\text{samp}})})$ depends on the test function.

3 Simulation-Based Inference

3.1 NPE-A (Fast ϵ-free inference.)

The method introduced in Papamakarios and Murray [4] approximates the true posterior $\pi(\boldsymbol{\theta}|\boldsymbol{x})$ using a K-component Mixture Density Network (MDN)

$$q_\phi(\boldsymbol{\theta}|\boldsymbol{x}) = \sum_{k=1}^{K} \alpha_\phi^{(k)}(\boldsymbol{x}) \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}_\phi^{(k)}(\boldsymbol{x}), \Sigma_\phi^{(k)}(\boldsymbol{x}))$$

where the weights $\alpha_\phi^{(k)}(\boldsymbol{x})$, means $\boldsymbol{\mu}_\phi^{(k)}(\boldsymbol{x})$, and covariance matrices $\Sigma_\phi^{(k)}(\boldsymbol{x})$ for $k = 1, 2, ..., K$ are computed by passing the data $\boldsymbol{x}$ through a feedforward neural network as follows:
We first compute a latent vector $\mathbf{z}$ by passing $\boldsymbol{x}$ through a (usually deep) neural network f_{ϕ_z} parameterized by trainable weights and biases ϕ_z:

$$\boldsymbol{z} = f_{\phi_z}(\boldsymbol{x})$$

We can think of $\mathbf{z}$ as a learnable embedding of the data $\boldsymbol{x}$ that will be used to compute the coefficients, means, and covariances of the MDN as follows: We compute the mixing coefficients $\boldsymbol{\alpha}(\boldsymbol{x}) := (\alpha^{(1)}(\boldsymbol{x}), ..., \alpha^{(K)}(\boldsymbol{x}))$ by passing the embedding $\boldsymbol{z}$ through a 1-layer neural network f_{ϕ_α} with the softmax activation function (to ensure the coefficients are positive and sum to 1):

$$\boldsymbol{\alpha}(\boldsymbol{x}) = f_{\phi_\alpha}(\boldsymbol{z}) = \text{softmax}\left(\boldsymbol{W}_\alpha \boldsymbol{z} + \boldsymbol{b}_\alpha\right)$$

where $\phi_\alpha = (\boldsymbol{W}_\alpha, \boldsymbol{b}_\alpha)$ denotes the trainable weights and biases of the neural network for the mixing coefficients $\boldsymbol{\alpha}(\boldsymbol{x})$. Similarly, for each $k = 1, ..., K$, the k'th mean $\boldsymbol{\mu}_k(\boldsymbol{x})$ is computed by passing the embedding $\boldsymbol{z}$ through a linear transformation $f_{\phi_{\mu_k}}$

$$\boldsymbol{\mu}_k(\boldsymbol{x}) = f_{\phi_{\mu_k}}(\boldsymbol{z}) = \boldsymbol{W}_{\mu_k} \boldsymbol{z} + \boldsymbol{b}_{\mu_k}$$

where $\phi_{\mu_k} = (\boldsymbol{W}_{\mu_k}, \boldsymbol{b}_{\mu_k})$ denotes the trainable weights and biases of the neural network for the k'th mean $\boldsymbol{\mu}_k(\boldsymbol{x})$. Lastly, for each $k = 1, ..., K$, we parameterize the k'th precision matrix[5] using a Cholesky decomposition $(\Sigma^{(k)}(\boldsymbol{x}))^{-1} = U_k(\boldsymbol{x})^T U_k(\boldsymbol{x})$ where the Cholesky factor $U_k(\boldsymbol{x})$ is an upper bidiagonal matrix (this formulation ensures that the covariance matrix is symmetric positive definite). The diagonal of $U_k(\boldsymbol{x})$ is computed by passing the embedding $\boldsymbol{z}$ through a 1-layer neural network $f_{\phi_{\text{diag}_k}}$ with the exponential activation function (to ensure all diagonal entries are positive):

$$\text{diag}\left(U_k(\boldsymbol{x})\right) = f_{\phi_{\text{diag}_k}}(\boldsymbol{z}) = \exp\left(\boldsymbol{W}_{\text{diag}_k} \boldsymbol{z} + \boldsymbol{b}_{\text{diag}_k}\right)$$

where $\phi_{\text{diag}_k} = (\boldsymbol{W}_{\text{diag}_k}, \boldsymbol{b}_{\text{diag}_k})$ denotes the trainable weights and biases of the neural network for the k'th diagonal $\text{diag}\left(U_k(\boldsymbol{x})\right)$. The upper diagonal of $U_k(\boldsymbol{x})$ is computed by passing the embedding $\boldsymbol{z}$ through a linear transformation $f_{\phi_{\text{udiag}_k}}$:

$$\text{udiag}\left(U_k(\boldsymbol{x})\right) = f_{\phi_{\text{udiag}_k}}(\boldsymbol{z}) = \boldsymbol{W}_{\text{udiag}_k} \boldsymbol{z} + \boldsymbol{b}_{\text{udiag}_k}$$

where $\phi_{\text{udiag}_k} = (\boldsymbol{W}_{\text{udiag}_k}, \boldsymbol{b}_{\text{udiag}_k})$ denotes the trainable weights and biases of the neural network for the k'th upper diagonal $\text{udiag}\left(U_k(\boldsymbol{x})\right)$. See Appendix B from Papamakarios and Murray [4] for more details on the MDN. Grouping all the trainable parameters of the network, we conclude that the MDN q_ϕ has trainable parameters given by

$$\phi = (\phi_z, \phi_\alpha, \phi_{\mu_1}, ..., \phi_{\mu_K}, \phi_{\text{diag}_1}, ..., \phi_{\text{diag}_K}, \phi_{\text{udiag}_1}, \phi_{\text{udiag}_K})$$

NPE-A has two main settings: amortized (which we will refer to as NPE-A) or sequential (which we will refer to as SNPE-A, where the "S" represents "sequential").

3.1.1 Amortized

Amortized NPE-A generates an approximate posterior $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$ by training the MDN q_ϕ on samples $\{(\boldsymbol{x}_n, \boldsymbol{\theta}_n)\}_{n=1}^N$ where each $\boldsymbol{\theta}_n \sim \pi(\boldsymbol{\theta})$ and $\boldsymbol{x}_n \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}_n)$ (i.e. $(\boldsymbol{x}_n, \boldsymbol{\theta}_n) \sim \pi(\boldsymbol{x}, \boldsymbol{\theta})$). In words, we use the true prior distribution $\pi(\boldsymbol{\theta})$ to propose the parameter samples $\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_N$ and, for each parameter sample $\boldsymbol{\theta}_n$, we sample data $\boldsymbol{x}_n$ from the simulator given that parameter. Given this simulated dataset $\{(\boldsymbol{x}_n, \boldsymbol{\theta}_n)\}_{n=1}^N$, we train the MDN's neural networks by minimizing the negative log likelihood (i.e. maximizing the log likelihood) of the parameter samples given their data samples:

$$\phi^* = \arg\min_\phi \left[-\sum_{n=1}^{N} \log\left(q_\phi(\boldsymbol{\theta}_n|\boldsymbol{x}_n)\right) \right]$$

Assuming the function class $\{q_\phi\}_\phi$ is rich enough such that $q_\phi(\boldsymbol{\theta}|\boldsymbol{x}) = \pi(\boldsymbol{\theta}|\boldsymbol{x})$ for some ϕ, it can easily be shown that, in the limit $N \to \infty$,

$$q_{\phi^*}(\boldsymbol{\theta}|\boldsymbol{x}) = \pi(\boldsymbol{\theta}|\boldsymbol{x})$$

for all $(\boldsymbol{x}, \boldsymbol{\theta}) \in X \times \Theta$ (see Proposition 1 and Appendix A from Papamakarios and Murray [4]). The NPE-A algorithm is summarized in Algorithm 2.

[5] The k'th precision matrix of an MDN is the inverse of k'th covariance matrix.

Algorithm 2: NPE-A Algorithm

Input: Prior $\pi(\boldsymbol{\theta})$, likelihood sampler $\pi(\boldsymbol{x}|\tilde{\boldsymbol{\theta}})$, number of samples N, number of MDN components K.
Output: Amortized posterior approximation $\tilde{\pi}(\boldsymbol{\theta}|\boldsymbol{x})$.

initialize an MDN q_ϕ with K components.
for $n = 1$ **to** N **do**
  sample $\boldsymbol{\theta}_n \sim \pi(\boldsymbol{\theta})$
  sample $\boldsymbol{x}_n \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}_n)$
end
Train the MDN parameters ϕ on $\{(\boldsymbol{\theta}_n, \boldsymbol{x}_n)\}_{n=1}^N$:

$$\phi^* = \arg\min_\phi \left[-\sum_{n=1}^{N} \log\left(q_\phi(\boldsymbol{\theta}_n|\boldsymbol{x}_n)\right) \right]$$

return $\tilde{\pi} := q_{\phi^*}$ as the amortized posterior approximation.

References

[1] Samantha R. Cook, Andrew Gelman, and Donald B. Rubin. Validation of software for bayesian models using posterior quantiles. *Journal of Computational and Graphical Statistics*, 15(3):675–692, 2006. ISSN 10618600. URL http://www.jstor.org/stable/27594203.

[2] John Geweke. Getting it right: Joint distribution tests of posterior simulators. *Journal of the American Statistical Association*, 99(467):799–804, 2004. ISSN 01621459. URL http://www.jstor.org/stable/27590449.

[3] Martin Modrák, Angie H. Moon, Shinyoung Kim, Paul Bürkner, Niko Huurre, Kateřina Faltejsková, Andrew Gelman, and Aki Vehtari. Simulation-based calibration checking for bayesian computation: The choice of test quantities shapes sensitivity. *Bayesian Analysis*, 20(2), June 2025. ISSN 1936-0975. doi: 10.1214/23-ba1404. URL http://dx.doi.org/10.1214/23-BA1404.

[4] George Papamakarios and Iain Murray. Fast ϵ-free inference of simulation models with bayesian conditional density estimation, 2018. URL https://arxiv.org/abs/1605.06376.

[5] Sean Talts, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. Validating bayesian inference algorithms with simulation-based calibration, 2020. URL https://arxiv.org/abs/1804.06788.