Detecting Model Misspecification in Amortized Bayesian Inference with Neural Networks

Marvin Schmitt[1(✉)], Paul-Christian Bürkner[1,2], Ullrich Köthe[3], and Stefan T. Radev[4]

[1] Cluster of Excellence SimTech, University of Stuttgart, Stuttgart, Germany
mail.marvinschmitt@gmail.com
[2] Department of Statistics, TU Dortmund University, Dortmund, Germany
[3] Visual Learning Lab, Heidelberg University, Heidelberg, Germany
[4] Cluster of Excellence STRUCTURES, Heidelberg University, Heidelberg, Germany

Abstract. Recent advances in probabilistic deep learning enable efficient amortized Bayesian inference in settings where the likelihood function is only implicitly defined by a simulation program (simulation-based inference; SBI). But how faithful is such inference if the simulation represents reality somewhat inaccurately—that is, if the true system behavior at test time deviates from the one seen during training? We conceptualize the types of model misspecification arising in SBI and systematically investigate how the performance of neural posterior approximators gradually deteriorates under these misspecifications, making inference results less and less trustworthy. To notify users about this problem, we propose a new misspecification measure that can be trained in an unsupervised fashion (i.e., without training data from the true distribution) and reliably detects model misspecification at test time. Our experiments clearly demonstrate the utility of our new measure both on toy examples with an analytical ground-truth and on representative scientific tasks in cell biology, cognitive decision making, and disease outbreak dynamics. We show how the proposed misspecification test warns users about suspicious outputs, raises an alarm when predictions are not trustworthy, and guides model designers in their search for better simulators.

Keywords: Simulation-Based Inference · Model Misspecification · Robustness

1 Introduction

Computer simulations play a fundamental role in many fields of science. However, the associated *inverse* problems of finding simulation parameters that accurately reproduce or predict real-world behavior are generally difficult and analytically

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-54605-1_35.

U. Köthe and C. Rother (Eds.): DAGM GCPR 2023, LNCS 14264, pp. 541–557, 2024.
https://doi.org/10.1007/978-3-031-54605-1_35

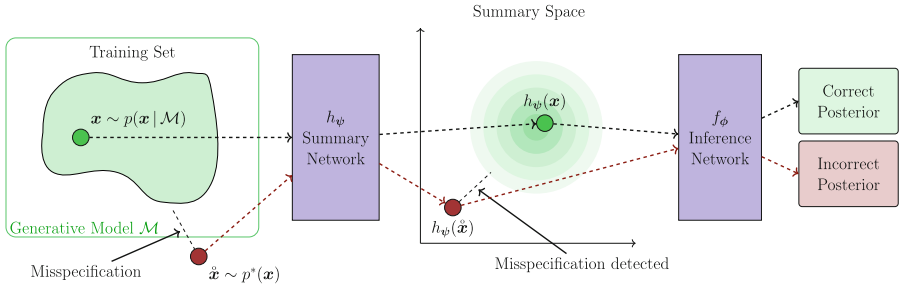


Fig. 1. Conceptual overview of our neural approach. The summary network h_ψ maps observations $\boldsymbol{x}$ to summary statistics $h_\psi(\boldsymbol{x})$, and the inference network f_ϕ estimates the posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x}, \mathcal{M})$ from the summary statistics. The generative model $\mathcal{M}$ creates training data $\boldsymbol{x}$ in the green region, and the networks learn to map these data to well-defined summary statistics and posteriors (green regions/dot/box). If the generative model $\mathcal{M}$ is misspecificed, real observations $\mathring{\boldsymbol{x}}$ fall outside the training region and are therefore mapped to outlying summary statistics and potentially incorrect posteriors (red dots/box). Since our learning approach enforces a known inlier summary distribution (e.g., Gaussian), misspecification can be detected by a distribution mismatch in summary space, as signaled by a high maximum mean discrepancy [22] score. (Color figure online)

intractable. Here, we consider *simulation-based inference* (SBI) [9] as a general approach to overcome this difficulty within a Bayesian inference framework. That is, given an assumed generative model $\mathcal{M}$ (as represented by the simulation program, see Sect. 3.2 for details) and observations $\boldsymbol{x}$ (real or simulated outcomes), we estimate the posterior distribution $p(\boldsymbol{\theta} \mid \boldsymbol{x}, \mathcal{M})$ of the simulation parameters $\boldsymbol{\theta}$ that would reproduce the observed $\boldsymbol{x}$. The recent introduction of efficient neural network approximators for this task has inspired a rapidly growing literature on SBI solutions for various application domains [4,6,18,20,29,33,48]. These empirical successes call for a systematic investigation of the trustworthiness of SBI, see Fig. 1.

We conduct an extensive analysis of neural posterior estimation (NPE) and sequential neural posterior estimation (SNPE), two deep learning algorithms to approximate the posterior distribution $p(\boldsymbol{\theta} \mid \boldsymbol{x}, \mathcal{M})$. In particular, we study their accuracy under model misspecification, where the generative model $\mathcal{M}^*$ at test time (the "true data generating process") deviates from the one assumed during training (i.e., $\mathcal{M}^* \neq \mathcal{M}$), a situation commonly known as *simulation gap*. As a consequence of a simulation gap, the observed data of interest might lie outside of the simulated data from the training phase of SBI. Paralleling the notion of "out-of-distribution" in anomaly detection and representation learning, simulation gaps may lead to "out-of-simulation" samples, and ultimately to wrong posterior estimates.

In this work, we propose a new misspecification measure that can be trained in an unsupervised fashion (i.e., without knowledge of $\mathcal{M}^*$ or training data from the true data distribution) and reliably quantifies by how much $\mathcal{M}^*$ deviates from $\mathcal{M}$ at test time. Our experiments clearly demonstrate the power of our

new measure both on toy examples with an analytical ground-truth, and on representative scientific tasks in cell biology, cognitive decision making, and disease outbreak dynamics. We show how simulation-based posterior inference gradually deteriorates as the simulation gap widens and how the proposed misspecification test warns users about suspicious outputs, raises an alarm when predictions are not trustworthy, and guides model designers in their search for better simulators. Thus, our investigations complement existing work on deep amortized SBI, whose main focus has been on network architectures and training algorithms for high accuracy in the well-specified case $\mathcal{M}^* = \mathcal{M}$ [14,21,35,38,41,45,46]. In particular, our paper makes the following key contributions:

(i) We systematically conceptualize different sources of model misspecification in amortized Bayesian inference with neural networks and propose a new detection criterion that is widely applicable to different model structures, inputs, and outputs.
(ii) We incorporate this criterion into existing neural posterior estimation methods, with hand-crafted and learned summary statistics, with sequential or amortized inference regimes, and we extend the associated learning algorithms in a largely non-intrusive manner.
(iii) We conduct a systematic empirical evaluation of our detection criterion, the influence of the summary space dimension, and the relationship between summary outliers and posterior distortion under various types and strengths of model misspecification.

2 Related Work

Model misspecification has been studied both in the context of standard Bayesian inference and generalizations thereof [28,47]. To alleviate model misspecification in generalized Bayesian inference, researchers have investigated probabilistic classifiers [52], second-order PAC-Bayes bounds [36], scoring rules [19], priors over a class of predictive models [31], or Stein discrepancy as a loss function [37]. Notably, these approaches deviate from the standard Bayesian formulation and investigate alternative schemes for belief updating and learning (e.g., replacing the likelihood function with a generic loss function). In contrast, our method remains grounded in the standard Bayesian framework embodying an implicit likelihood principle [3]. Differently, power scaling methods incorporate a modified likelihood (raised to a power $0 < \alpha < 1$) in order to prevent potentially overconfident Bayesian updating [23,26]. However, the SBI setting assumes that the likelihood function is not available in closed-form, which makes an explicit modification of the implicitly defined likelihood less obvious.

Neural approaches to amortized SBI can be categorized as either targeting the posterior [21,45], the likelihood [24,43], or both [56]. These methods employ simulations for training amortized neural approximators which can either generate samples from the posterior directly [21,45,56] or in tandem with Markov chain Monte Carlo (MCMC) sampling algorithms [24,43]. Since the behavior of these methods depends on the fidelity of the simulations used as training data,

we hypothesize that their estimation quality will be, in general, unpredictable, when faced with atypical real-world data. Indeed, the critical impact of model misspecification in neural SBI has been commonly acknowledged in the scientific research community [1,8,15,16,39,58].

Recent approaches to detect model misspecification in simulation-based inference are usually based on the obtained approximate posterior distribution [12,25,30]. However, we show in **Experiment 1** and **Experiment 5 (Appendix)** that the approximate posteriors in simulation-based inference tend to show pathological behavior under misspecified models. Posteriors from misspecified models may erroneously look legitimate, rendering diagnostic methods on their basis unreliable. Moreover, the same applies for approaches based on the *posterior predictive distribution* [7,17,53] since these also rely on the fidelity of the posterior distribution and can therefore only serve as an indirect measure of misspecification.

A few novel techniques aim to *mitigate* model misspecification in simulation-based inference to achieve robust inference. [11] equip neural ratio estimation [24] with a balancing condition which tends to produce more conservative posterior approximations. [54] explore a way to alleviate model misspecification with two neural approximators and subsequent MCMC. While both approaches are appealing in theory, the computational burden of MCMC sampling contradicts the idea of amortized inference and prohibits their use in complex applications with learned summary statistics and large amounts of data. In fact, [29] used amortized neural SBI on more than a million data sets of multiple observations each and demonstrated that an alternative inference method involving non-amortized MCMC would have taken years of sampling.

For robust non-amortized ABC samplers, the possibility of utilizing hand-crafted summary statistics as an important element of misspecification analysis has already been explored [15,16]. Our work parallels these ideas and extends them to the case of *learnable summary statistics* in amortized SBI on potentially massive data sets, where ABC becomes infeasible. However, we show in **Experiment 2** that our method also works with hand-crafted summary statistics.

Finally, from the perspective of deep anomaly detection, our approach for learning informative summary statistics can be viewed as a special case of *generic normality feature learning* [40]. Standard learned summary statistics are optimized with a generic feature learning objective which is not primarily designed for anomaly detection [45]. However, since learned summary statistics are also optimized to be maximally informative for posterior inference, they will likely capture underlying data regularities [40].

3 Method

For simulation-based Bayesian inference, we define a generative model as a triple $\mathcal{M} = \big(g(\boldsymbol{\theta}, \boldsymbol{\xi}), p(\boldsymbol{\xi} \,|\, \boldsymbol{\theta}), p(\boldsymbol{\theta})\big)$. A generative model $\mathcal{M}$ generates data $\boldsymbol{x} \in \mathcal{X}$ according to the system

$$\boldsymbol{x} = g(\boldsymbol{\theta}, \boldsymbol{\xi}) \quad \text{with} \quad \boldsymbol{\xi} \sim p(\boldsymbol{\xi} \,|\, \boldsymbol{\theta}),\ \boldsymbol{\theta} \sim p(\boldsymbol{\theta}), \tag{1}$$

where g denotes a (randomized) simulator, $\boldsymbol{\xi} \in \Xi$ is a source of randomness (i.e., noise) with density function $p(\boldsymbol{\xi} \mid \boldsymbol{\theta})$, and $p(\boldsymbol{\theta})$ encodes prior knowledge about plausible simulation parameters $\boldsymbol{\theta} \in \Theta$. Throughout the paper, we use the decorated symbol $\mathring{\boldsymbol{x}}$ to mark data that was in fact *observed* in the real world and not merely simulated by the assumed model $\mathcal{M}$. The parameters $\boldsymbol{\theta}$ consist of hidden properties whose role in g we explicitly understand and model, and $\boldsymbol{\xi}$ takes care of nuisance effects that we only treat statistically. The abstract spaces $\mathcal{X}, \Xi$, and Θ denote the domain of possible output data (possible worlds), the scope of noise, and the set of admissible model parameters, respectively. The distinction between hidden properties $\boldsymbol{\theta}$ and noise $\boldsymbol{\xi}$ is not entirely clear-cut, but depends on our modeling goals and may vary across applications.

Our generative model formulation is equivalent to the standard factorization of the Bayesian joint distribution into likelihood and prior, $p(\boldsymbol{\theta}, \boldsymbol{x} \mid \mathcal{M}) = p(\boldsymbol{x} \mid \boldsymbol{\theta}, \mathcal{M})\, p(\boldsymbol{\theta} \mid \mathcal{M})$, where $\mathcal{M}$ expresses the prior knowledge and assumptions embodied in the model. The likelihood is obtained by marginalizing the joint distribution $p(\boldsymbol{\xi}, \boldsymbol{x} \mid \boldsymbol{\theta}, \mathcal{M})$ over all possible values of the nuisance parameters $\boldsymbol{\xi}$, that is, over all possible execution paths of the simulation program, for fixed $\boldsymbol{\theta}$:

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

This integral is typically intractable [9], but we assume that it exists and is non-degenerate, that is, it defines a proper density over the constrained manifold $(g(\boldsymbol{\theta}, \boldsymbol{\xi}), \boldsymbol{\xi})$, and this density can be learned. A major challenge in Bayesian inference is approximating the posterior distribution $p(\boldsymbol{\theta} \mid \boldsymbol{x}, \mathcal{M}) \propto p(\boldsymbol{x} \mid \boldsymbol{\theta}, \mathcal{M})\, p(\boldsymbol{\theta} \mid \mathcal{M})$. Below, we focus on *amortized* posterior approximation with neural networks, which aims to achieve zero-shot posterior sampling for any input data $\boldsymbol{x}$ compatible with the reference model $\mathcal{M}$.[1]

3.1 Neural Posterior Estimation

Neural Posterior Estimation (NPE) with learned summary statistics $h_{\psi}(\boldsymbol{x})$ involves a posterior network and a summary network which jointly minimize the expected KL divergence between analytic and approximate posterior

$$\boldsymbol{\psi}^*, \boldsymbol{\phi}^* = \underset{\boldsymbol{\psi}, \boldsymbol{\phi}}{\operatorname{argmin}} \, \mathbb{E}_{p(\boldsymbol{x} \mid \mathcal{M})} \Big[\mathbb{KL} \left[p(\boldsymbol{\theta} \mid \boldsymbol{x}, \mathcal{M}) \,||\, q_{\boldsymbol{\phi}} \left(\boldsymbol{\theta} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}), \mathcal{M} \right) \right] \Big], \tag{3}$$

where the expectation runs over the prior predictive distribution $p(\boldsymbol{x} \mid \mathcal{M})$. The above criterion simplifies to

$$\boldsymbol{\psi}^*, \boldsymbol{\phi}^* = \underset{\boldsymbol{\psi}, \boldsymbol{\phi}}{\operatorname{argmin}} \, \mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x} \mid \mathcal{M})} \Big[- \log q_{\boldsymbol{\phi}} \left(\boldsymbol{\theta} \mid h_{\boldsymbol{\psi}}(\boldsymbol{x}), \mathcal{M} \right) \Big], \tag{4}$$

since the analytic posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x}, \mathcal{M})$ does not depend on the trainable neural network parameters $(\boldsymbol{\psi}, \boldsymbol{\phi})$. This criterion optimizes a summary (aka embedding)

[1] We demonstrate in **Experiment 1** that model misspecification also affects the performance of non-amortized sequential neural posterior estimation.

network with parameters $\boldsymbol{\psi}$ and an inference network with parameters $\boldsymbol{\phi}$ which learn to perform zero-shot posterior estimation over the generative scope of $\mathcal{M}$. The summary network transforms input data $\boldsymbol{x}$ of variable size and structure to a fixed-length representation $\boldsymbol{z} = h_{\psi}(\boldsymbol{x})$. The inference network f_{ϕ} generates random draws from an approximate posterior q_{ϕ} via a normalizing flow, for instance, realized by a conditional invertible neural network [2] or a conditional masked autoregressive flow [42].

We approximate the expectation in Eq. 4 via simulations from the generative model $\mathcal{M}$ and repeat the process until convergence, which enables us to perform fully amortized inference (i.e., the posterior functional can be evaluated for any number of observed data sets $\boldsymbol{x}$). Moreover, this objective is self-consistent and results in correct amortized inference under optimal convergence [21,45].

3.2 Model Misspecification in Simulation-Based Inference

When modeling a complex system or process, we typically assume an unknown (true) generator $\mathcal{M}^*$, which yields an unknown (true) distribution $\mathring{\boldsymbol{x}} \sim p^*(\boldsymbol{x})$ and is available to the data analyst only via a finite realization (i.e., actually observed data $\mathring{\boldsymbol{x}}$). According to a common definition [16,32,36,55], the generative model $\mathcal{M}$ is well-specified if a "true" parameter $\theta^* \in \Theta$ exists, such that the (conditional) likelihood matches the data-generating distribution,

$$p(\boldsymbol{x} \mid \boldsymbol{\theta}^*, \mathcal{M}) = p^*(\boldsymbol{x}), \tag{5}$$

and misspecified otherwise. This likelihood-centered definition is well-established and sensible in many domains of Bayesian inference.

In *simulation-based* inference, however, there is an additional difficulty regarding model specification: Simulation-based training (see Eq. 3) takes the expectation with respect to the model-implied prior predictive distribution $p(\boldsymbol{x} \mid \mathcal{M})$, not necessarily the "true" real-world distribution $p^*(\boldsymbol{x})$. Thus, optimal convergence does not imply correct amortized inference or faithful prediction in the real world when there is a simulation gap, that is, when the assumed training model $\mathcal{M}$ deviates critically from the unknown true generative model $\mathcal{M}^*$.

Crucially, even if the generative model $\mathcal{M}$ is well-specified according to the likelihood-centered definition in Eq. 5, finite training with respect to a "wrong" prior (predictive) distribution will likely result in insufficient learning of relevant parameter (and data) regions. This scenario could also be framed as "out-of-simulation" (OOSim) by analogy with the common out-of-distribution (OOD) problem in machine learning applications [57]. In fact, we observe in **Experiment 1** that a misspecified prior distribution worsens posterior inference just like a misspecified likelihood function does.

Thus, our adjusted definition of model misspecification *in the context of simulation-based inference* considers the entire prior predictive distribution $p(\boldsymbol{x} \mid \mathcal{M})$: A generative model $\mathcal{M}$ is well-specified if the information loss through modeling $p^*(\boldsymbol{x})$ with $p(\boldsymbol{x} \mid \mathcal{M})$ falls below an acceptance threshold ϑ,

$$\mathbb{D}\big[p(\boldsymbol{x} \mid \mathcal{M}) \,||\, p^*(\boldsymbol{x})\big] < \vartheta, \tag{6}$$

and misspecified otherwise. The symbol $\mathbb{D}$ denotes a divergence metric quantifying the "distance" between the data distributions implied by reality and by the model (i.e., the marginal likelihood). A natural choice for $\mathbb{D}$ would stem from the family of $\mathcal{F}$-divergences, such as the KL divergence. However, we choose the Maximum Mean Discrepancy (MMD) because we can tractably estimate it on finite samples from $p(\boldsymbol{x} \mid \mathcal{M})$ and $p^*(\boldsymbol{x})$ and its analytic value equals zero if and only if the two densities are equal [22].

Our adjusted definition of model misspecification no longer assumes the existence of a *true* parameter vector $\boldsymbol{\theta}^*$ (cf. Eq. 5). Instead, we focus on the *marginal likelihood* $p(\boldsymbol{x} \mid \mathcal{M})$ which represents the entire prior predictive distribution of a model and does not commit to a single most representative parameter vector. In this way, multiple models whose marginal distributions are representative of $p^*(\boldsymbol{x})$ can be considered well-specified without any reference to some hypothetical ground-truth $\boldsymbol{\theta}^*$, which may not even exist for opaque systems with unknown properties.

3.3 Structured Summary Statistics

In simulation-based inference, summary statistics have a dual purpose because (i) they are fixed-length vectors, even if the input data $\boldsymbol{x}$ have variable length; and (ii) they usually contain crucial features of the data, which simplifies neural posterior inference. However, in complex real-world scenarios such as COVID-19 modeling (see **Experiment 3**), it is not feasible to rely on hand-crafted summary statistics. Thus, combining neural posterior estimation with *learned summary statistics* leverages the benefits of summary statistics (i.e., compression to fixed-length vectors) while avoiding the virtually impossible task of designing hand-crafted summary statistics for complex models.

In simulation-based inference, the summary network h_{ψ} acts as an interface between the data $\boldsymbol{x}$ and the inference network f_{ϕ}. Its role is to learn maximally informative summary vectors of fixed size S from complex and structured observations (e.g., sets of *i.i.d.* measurements or multivariate time series). Since the learned summary statistics are optimized to be maximally informative for posterior inference, they are forced to capture underlying data regularities (see Sect. 2). Therefore, we deem the summary network's representation $\boldsymbol{z} = h_{\psi}(\boldsymbol{x})$ as an adequate target to detect simulation gaps.

Specifically, we prescribe an S-dimensional multivariate unit (aka. standard) Gaussian distribution to the summary space, $p\big(\boldsymbol{z} = h_{\psi}(\boldsymbol{x}) \mid \mathcal{M}\big) \approx \mathcal{N}(\boldsymbol{z} \mid \mathbf{0}, \mathbb{I})$, by minimizing the MMD between summary network outputs and random draws from a unit Gaussian distribution. To ensure that the summary vectors comply with the support of the Gaussian density, we use a linear (bottleneck) output layer with S units in the summary network. A random vector in summary space takes the form $h_{\psi}(\boldsymbol{x}) \equiv \boldsymbol{z} \equiv (z_1, \ldots, z_S) \in \mathbb{R}^S$. The extended optimization objective follows as

$$\begin{aligned}\boldsymbol{\psi}^*, \boldsymbol{\phi}^* = \underset{\boldsymbol{\psi},\boldsymbol{\phi}}{\operatorname{argmin}}\, \mathbb{E}_{p(\boldsymbol{\theta},\boldsymbol{x}\,|\,\mathcal{M})}\Big[-\log q_{\boldsymbol{\phi}}\big(\boldsymbol{\theta}\,|\,h_{\boldsymbol{\psi}}(\boldsymbol{x}),\mathcal{M}\big)\Big] \\ + \gamma\, \mathbb{MMD}^2\big[p\big(h_{\boldsymbol{\psi}}(\boldsymbol{x})\,|\,\mathcal{M}\big)\,||\,\mathcal{N}\big(\mathbf{0},\mathbb{I}\big)\big]\end{aligned} \tag{7}$$

with a hyperparameter γ to control the relative weight of the MMD term. Intuitively, this objective encourages the approximate posterior $q_{\boldsymbol{\phi}}\big(\boldsymbol{\theta}\,|\,h_{\boldsymbol{\psi}}(\boldsymbol{x}),\mathcal{M}\big)$ to match the correct posterior and the summary distribution $p\big(h_{\boldsymbol{\psi}}(\boldsymbol{x})\,|\,\mathcal{M}\big)$ to match a unit Gaussian. The extended objective does not constitute a theoretical trade-off between the two terms, since the MMD merely reshapes the summary distribution in an information-preserving manner. In practice, the extended objective may render optimization of the summary network more difficult, but our experiments suggest that it does not restrict the quality of the amortized posteriors.

This method is also directly applicable to hand-crafted summary statistics. Hand-crafted summary statistics already have fixed length and usually contain rich information for posterior inference. Thus, the task of the summary network $h_{\boldsymbol{\psi}}$ simplifies to transforming the hand-crafted summary statistics to a unit Gaussian (Eq. 7) to enable model misspecification via distribution matching during test time (see below). We apply our method to a problem with hand-crafted summary statistics in **Experiment 2**.

3.4 Detecting Model Misspecification with Finite Data

Once the simulation-based training phase is completed, we can generate M validation samples $\{\boldsymbol{\theta}^{(m)}, \boldsymbol{x}^{(m)}\}$ from our generative model $\mathcal{M}$ and pass them through the summary network to obtain a sample of latent summary vectors $\{\boldsymbol{z}^{(m)}\}$, where $\boldsymbol{z} = h_{\boldsymbol{\psi}}(\boldsymbol{x})$ denotes the output of the summary network. The properties of this sample contain important convergence information: If $\boldsymbol{z}$ is approximately unit Gaussian, we can assume a structured summary space given the training model $\mathcal{M}$. This enables model misspecification diagnostics via distribution checking during inference on real data (see the Appendix for the detailed algorithm).

Let $\{\mathring{\boldsymbol{x}}^{(n)}\}$ be an *observed* sample, either simulated from a different generative model, or arising from real-world observations with an unknown generator. Before invoking the inference network, we pass this sample through the summary network to obtain the summary statistics for the sample: $\{\mathring{\boldsymbol{z}}^{(n)}\}$. We then compare the validation summary distribution $\{\boldsymbol{z}^{(m)}\}$ with the summary statistics of the observed data $\{\mathring{\boldsymbol{z}}^{(n)}\}$ according to the sample-based MMD estimate $\widehat{\mathrm{MMD}}^2\,(p(\boldsymbol{z})\,||\,p(\mathring{\boldsymbol{z}}))$ [22]. Importantly, we are not limited to pre-determined sizes of simulated or real-world data sets, as the MMD estimator is defined for arbitrary M and N. To allow MMD estimation for data sets with single instances ($N = 1$ or $M = 1$), we do not use the unbiased MMD version from [22]. Singleton data sets are an important use case for our method in practice, and potential advantages of unbiased estimators do not justify exclusion of such data. To enhance visibility, the figures in the experimental section will depict the square root of the originally squared MMD estimate.

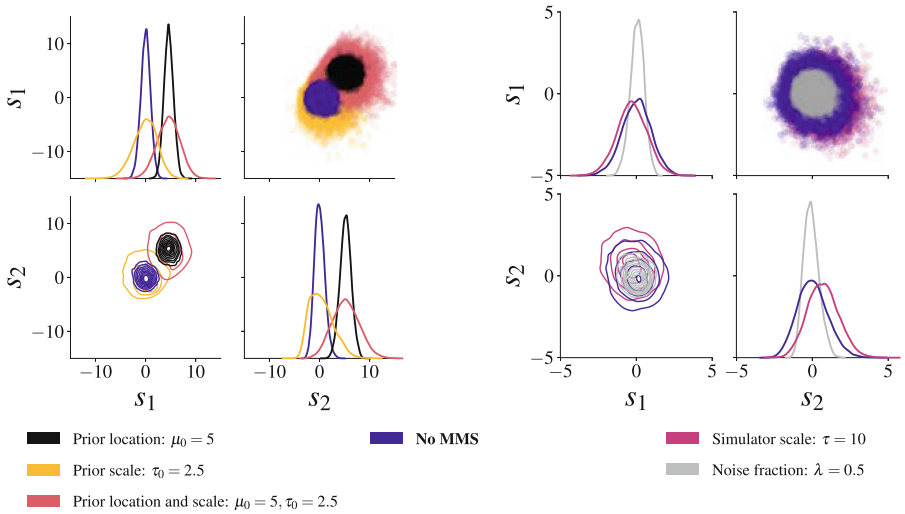


Fig. 2. Experiment 1. Summary space samples for the minimal sufficient summary network ($S = 2$) from a well-specified model $\mathcal{M}$ (blue) and several misspecified configurations. **Left:** Prior misspecification can be detected. **Right:** Noise misspecification can be detected, while simulator scale misspecification is indistinguishable from the validation summary statistics.

Whenever we estimate the MMD from finite data, its estimates vary according to a sampling distribution and we can resort to a frequentist hypothesis test to determine the probability of observed MMD values under well-specified models. Although this sampling distribution under the null hypothesis is unknown, we can estimate it from multiple sets of simulations from the generative model, $\{\boldsymbol{z}^{(m)}\}$ and $\{\boldsymbol{z}^{(n)}\}$, with M large and N equal to the number of real data sets. Based on the estimated sampling distribution, we can obtain a critical MMD value for a fixed Type I error probability (α) and compare it to the one estimated with the observed data. In general, a larger α level corresponds to a more conservative modeling approach: A larger type I error implies that more tests reject the null hypothesis, which corresponds to more frequent model misspecification alarms and a higher chance that incorrect models will be recognised. The Type II error probability (β) of this test will generally be high (i.e., the *power* of the test will be low) whenever the number of real data sets N is very small. However, we show in **Experiment 3** that even as few as 5 real data sets suffice to achieve $\beta \approx 0$ for a complex model on COVID-19 time series.

4 Experiments

4.1 Experiment 1: 2D Gaussian Means

We set the stage by estimating the means of a 2-dimensional conjugate Gaussian model with $K = 100$ observations per data set and a known analytic posterior in

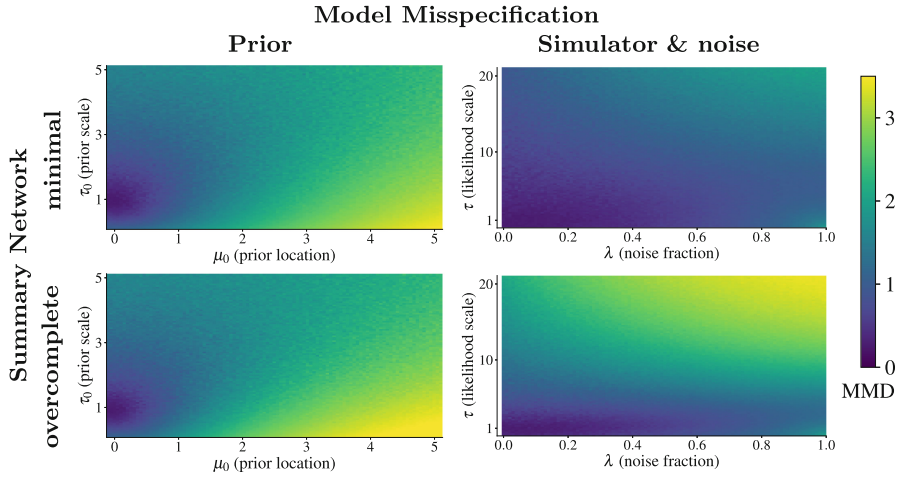


Fig. 3. Experiment 1. Summary space MMD as a function of misspecification severity. White stars indicate the well-specified model configuration (i.e., equal to the training model $\mathcal{M}$).

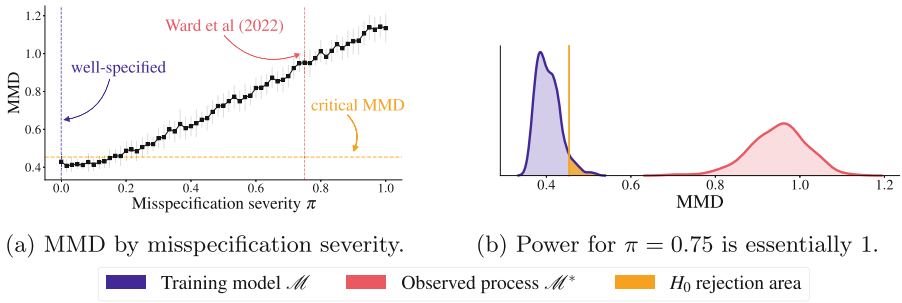
order to illustrate our method. This experiment contains the Gaussian examples from [16] and [54], and extends them by (i) studying misspecifications beyond the likelihood variance (see below); and (ii) implementing continuously widening simulation gaps, as opposed to a single discrete misspecification. The data generating process is defined as

$$\boldsymbol{x}_k \sim \mathcal{N}(\boldsymbol{x} \,|\, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad \text{for } k = 1, ..., K \qquad \text{with} \quad \boldsymbol{\mu} \sim \mathcal{N}(\boldsymbol{\mu} \,|\, \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0). \tag{8}$$

As a summary network, we use a permutation invariant network [5] with $S = 2$ output dimensions, which equal the number of minimal sufficient statistics implied by the analytic posterior. The terms "minimal", "sufficient", and "overcomplete" refer to the inference task and *not* to the data. Thus, $S = 2$ summary statistics are *sufficient* to solve the inference task, namely recover two means. For training the posterior approximator, we set the prior of the generative model $\mathcal{M}$ to a unit Gaussian and the likelihood covariance $\boldsymbol{\Sigma}$ to an identity matrix.

We induce prior misspecification by altering the prior location $\boldsymbol{\mu}_0$ and covariance $\boldsymbol{\Sigma}_0 = \tau_0 \mathbb{I}$ (only diagonal covariance, controlled through the factor τ_0). Further, we achieve misspecified likelihoods by manipulating the likelihood covariance $\boldsymbol{\Sigma} = \tau \mathbb{I}$ (only diagonal covariance, controlled through τ). We induce noise misspecification by replacing a fraction $\lambda \in [0, 1]$ of the data $\boldsymbol{x}$ with samples from a scaled Beta$(2, 5)$ distribution.

Results. The neural posterior estimator trained to minimize the augmented objective (Eq. 7) exhibits excellent recovery and calibration [49,50] in the well-specified case, as shown in the Appendix. All prior misspecifications manifest themselves in anomalies in the summary space which are directly detectable



(a) MMD by misspecification severity. (b) Power for $\pi = 0.75$ is essentially 1.

Fig. 4. Experiment 2. MMD increases with misspecification severity (a; mean, SD of 20 repetitions). Our test easily detects the setting from [54] (b).

through visual inspection of the 2-dimensional summary space in Fig. 2 (left). Note that the combined prior misspecification (location and scale) exhibits a summary space pattern that combines the location and scale of the respective location and scale misspecifications. However, based on the 2-dimensional summary space, misspecifications in the fixed parameters of the likelihood (τ) and mixture noise are not detectable via an increased MMD (see Fig. 3, top right).

We further investigate the effect of an *overcomplete* summary space with respect to the inference task, namely $S = 4$ learned summary statistics with an otherwise equal architecture. In addition to prior misspecifications, the overcomplete summary network also captures misspecifications in the noise and simulator via the MMD criterion (see Fig. 3, bottom row). Furthermore, the induced misspecifications in the noise and simulator are visually detectable in the summary space samples (see Appendix). Recall that the 2-dimensional summary space fails to capture these misspecifications (see Fig. 3, top right). The effect of model misspecificaiton on the posterior recovery error is described in the Appendix.

SNPE-C. Our method successfully detects model misspecification using SNPE-C [21] with a structured summary space (see Appendix). The results are largely equivalent to those obtained with NPE, as implemented in the BayesFlow framework [45].

4.2 Experiment 2: Cancer and Stromal Cell Model

This experiment illustrates model misspecification detection in a marked point process model of cancer and stromal cells [27]. We use the original implementation of [54] with hand-crafted summary statistics and showcase the applicability of our method in scenarios where good summary statistics are known. The inference parameters are three Poisson rates $\lambda_c, \lambda_p, \lambda_d$, and the setup in [54] extracts four hand-crafted summary statistics from the 2D plane data: (1–2) number of cancer and stromal cells; (3–4) mean and maximum distance from stromal cells to the nearest cancer cell. All implementation details are described in the Appendix.

We achieve misspecification during inference by mimicking necrosis, which often occurs in core regions of tumors. A Bernoulli distribution with parameter π controls whether a cell is affected by necrosis or not. Consequently, $\pi = 0$ implies no necrosis (and thus no simulation gap), and $\pi = 1$ entails that all cells are affected. The experiments by [54] study a single misspecification, namely the case $\pi = 0.75$ in our implementation. In order to employ our proposed method for model misspecification detection, we add a small summary network $h_\psi : \mathbb{R}^4 \to \mathbb{R}^4$ consisting of three hidden fully connected layers with 64 units each. This network h_ψ merely transforms the hand-crafted summary statistics into a 4-D unit Gaussian, followed by NPE for posterior inference.

Results. Our MMD misspecification score increases with increasingly severe model misspecification (i.e., increasing necrosis rate π), see Fig. 4a. What is more, for the single misspecification $\pi = 0.75$ studied by [54], we illustrate (i) the power of our proposed hypothesis test; and (ii) the summary space distribution for misspecified data. The power $(1 - \beta)$ essentially equals 1, as shown in Fig. 4b: The MMD sampling distributions under the training model (H_0) and under the observed data generating process ($\mathcal{M}^*$) are completely separated.

4.3 Experiment 3: Epidemiological Model for COVID-19

As a final real-world example, we treat a high-dimensional compartmental model representing the early months of the COVID-19 pandemic in Germany [44]. We investigate the utility of our method to detect simulation gaps in a much more realistic and non-trivial extension of the SIR settings in [34] and [54] with substantially increased complexity. Moreover, we perform inference on real COVID-19 data from Germany and use our new method to test whether the model used in [44] is misspecified, possibly undermining the trustworthiness of political conclusions that are based on the inferred posteriors. To achieve this, we train an NPE setup with the BayesFlow framework [45] identical to [44] but using our new optimization objective (Eq. 7) to encourage a structured summary space. We then simulate 1000 time series from the training model $\mathcal{M}$ and 1000 time series from three misspecified models: (i) a model $\mathcal{M}_1$ without an intervention sub-model; (ii) a model $\mathcal{M}_2$ without an observation sub-model; (iii) a model $\mathcal{M}_3$ without a "carrier" compartment [10].

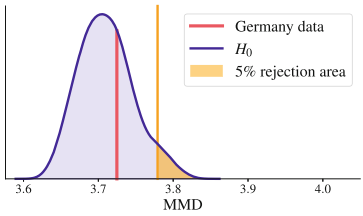


Fig. 5. Representation of Germany's COVID-19 time series w.r.t. the MMD distribution under $H_0 : p^*(\boldsymbol{x}) = p(\boldsymbol{x} \mid \mathcal{M})$.

Results. Table 1 shows the MMD between the summary representation of $N = 1, 2, 5$ bootstrapped time series from each model and the summary representation of the 1000 time series from model $\mathcal{M}$ (see the Appendix for bootstrapping details). We also calculate the power $(1 - \beta)$ of our hypothesis test for each misspecified model under the sampling distribution estimated from 1 000 samples of the 1 000 time series from $\mathcal{M}$ at a type I error probability of $\alpha = .05$. We

Table 1. Experiment 3. Results for different variations of the COVID-19 compartmental model. We report the median and 95% CI of 100 bootstrap samples.

Model	Bootstrap MMD			Power $(1-\beta)$		
	$N=1$	$N=2$	$N=5$	$N=1$	$N=2$	$N=5$
$\mathcal{M}$	3.70 [3.65, 3.79]	2.61 [2.54, 2.91]	1.66 [1.59, 1.84]	—	—	—
$\mathcal{M}_1$	3.76 [3.72, 3.80]	2.86 [2.62, 3.16]	2.11 [1.82, 2.50]	.998	.958	≈ 1.0
$\mathcal{M}_2$	3.80 [3.73, 3.83]	2.81 [2.65, 3.00]	2.01 [1.82, 2.19]	.789	.804	≈ 1.0
$\mathcal{M}_3$	3.78 [3.74, 3.83]	2.81 [2.68, 3.11]	2.07 [1.92, 2.41]	.631	.690	≈ 1.0

observe that the power of the test rapidly increases with more data sets and the Type II error probability (β) is essentially zero for as few as $N = 5$ time series.

As a next step, we pass the reported COVID-19 data between 1 March and 21 April 2020 [13] through the summary network and compute the critical MMD value for a sampling-based hypothesis test with an α level of .05 (see Fig. 5). The MMD of the Germany data is well below the critical MMD value (it essentially lies in the bulk of the distribution), leading to the conclusion that the assumed training model $\mathcal{M}$ is well-specified for this time period.

5 Conclusions

This paper approached a fundamental problem in amortized simulation-based Bayesian inference, namely, flagging potential posterior errors due to model misspecification. We argued that misspecified models might cause so-called *simulation gaps*, resulting in deviations between simulations during training time and actual observed data at test time. We further showed that simulation gaps can be detrimental for the performance and faithfulness of simulation-based inference relying on neural networks. We proposed to increase the networks' awareness of posterior errors by compressing simulations into a structured latent summary space induced by a modified optimization objective in an unsupervised fashion. We then applied the maximum mean discrepancy (MMD) estimator, equipped with a sampling-based hypothesis test, as a criterion to spotlight discrepancies between model-implied and actually observed distributions in summary space. While we focused on the application to NPE (BayesFlow implementation [45]) and SNPE (sbi implementation [51]), the proposed method can be easily integrated into other inference algorithms and frameworks as well. Our software implementations are available in the BayesFlow library (http://www.bayesflow.org) and can be seamlessly integrated into an end-to-end workflow for amortized simulation-based inference.

Acknowledgments. MS and PCB were supported by the Cyber Valley Research Fund (grant number: CyVy-RF-2021-16) and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy EXC-2075

- 390740016 (the Stuttgart Cluster of Excellence SimTech). UK was supported by the Informatics for Life initiative funded by the Klaus Tschira Foundation. STR was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC-2181 - 390900948 (the Heidelberg Cluster of Excellence STRUCTURES).

References

1. Alquier, P., Ridgway, J.: Concentration of tempered posteriors and of their variational approximations. arXiv:1706.09293 [cs, math, stat] (2019). arXiv: 1706.09293
2. Ardizzone, L., Lüth, C., Kruse, J., Rother, C., Köthe, U.: Guided image generation with conditional invertible neural networks (2019)
3. Berger, J.O., Wolpert, R.L.: The Likelihood Principle. No. v. 6 in Lecture Notes-Monograph Series. 2nd edn. Institute of Mathematical Statistics, Hayward (1988)
4. Bieringer, S., et al.: Measuring QCD splittings with invertible networks. SciPost Phys. Proc. **10**(6), 126 (2021)
5. Bloem-Reddy, B., Teh, Y.W.: Probabilistic symmetries and invariant neural networks. J. Mach. Learn. Res. **21**, 90–1 (2020)
6. Butter, A., et al.: Machine learning and LHC event generation. arXiv preprint arXiv:2203.07460 (2022)
7. Bürkner, P.C., Gabry, J., Vehtari, A.: Approximate leave-future-out cross-validation for Bayesian time series models. J. Stat. Comput. Simul. **90**(14), 2499–2523 (2020). https://doi.org/10.1080/00949655.2020.1783262. arXiv:1902.06281 [stat]
8. Cannon, P., Ward, D., Schmon, S.M.: Investigating the impact of model misspecification in neural simulation-based inference (2022). arXiv:2209.01845 [cs, stat]
9. Cranmer, K., Brehmer, J., Louppe, G.: The frontier of simulation-based inference. Proc. Natl. Acad. Sci. **117**(48), 30055–30062 (2020)
10. Dehning, J., et al.: Inferring change points in the spread of COVID-19 reveals the effectiveness of interventions. Science **369**(6500) (2020)
11. Delaunoy, A., Hermans, J., Rozet, F., Wehenkel, A., Louppe, G.: Towards reliable simulation-based inference with balanced neural ratio estimation (2022). arXiv:2208.13624 [cs, stat]
12. Dellaporta, C., Knoblauch, J., Damoulas, T., Briol, F.X.: Robust Bayesian inference for simulator-based models via the MMD Posterior Bootstrap (2022). https://doi.org/10.48550/ARXIV.2202.04744
13. Dong, E., Du, H., Gardner, L.: An interactive web-based dashboard to track COVID-19 in real time. Lancet. Infect. Dis **20**(5), 533–534 (2020). https://doi.org/10.1016/S1473-3099(20)30120-1
14. Durkan, C., Murray, I., Papamakarios, G.: On contrastive learning for likelihood-free inference. In: International Conference on Machine Learning, pp. 2771–2781. PMLR (2020)
15. Frazier, D.T., Drovandi, C.: Robust approximate Bayesian inference with synthetic likelihood. J. Comput. Graph. Stat. **30**(4), 958–976 (2021). https://doi.org/10.1080/10618600.2021.1875839
16. Frazier, D.T., Robert, C.P., Rousseau, J.: Model misspecification in approximate Bayesian computation: consequences and diagnostics. J. Royal Stat. Soc. Ser. B (Stat. Method.) **82**(2), 421–444 (2020). https://doi.org/10.1111/rssb.12356

17. Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., Gelman, A.: Visualization in Bayesian workflow. J. Royal Stat. Soc. Ser. A (Stat. Soc.) **182**(2), 389–402 (2019)
18. Ghaderi-Kangavari, A., Rad, J.A., Nunez, M.D.: A general integrative neurocognitive modeling framework to jointly describe EEG and decision-making on single trials. Comput. Brain Behav. (2023). https://doi.org/10.1007/s42113-023-00167-4
19. Giummolè, F., Mameli, V., Ruli, E., Ventura, L.: Objective Bayesian inference with proper scoring rules. TEST **28**(3), 728–755 (2019)
20. Gonçalves, P.J., et al.: Training deep neural density estimators to identify mechanistic models of neural dynamics. Elife **9**, e56261 (2020)
21. Greenberg, D., Nonnenmacher, M., Macke, J.: Automatic posterior transformation for likelihood-free inference. In: International Conference on Machine Learning, pp. 2404–2414. PMLR (2019)
22. Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., Smola, A.: A Kernel two-sample test. J. Mach. Learn. Res. **13**, 723–773 (2012)
23. Grünwald, P., Van Ommen, T., et al.: Inconsistency of Bayesian inference for misspecified linear models, and a proposal for repairing it. Bayesian Anal. **12**(4), 1069–1103 (2017)
24. Hermans, J., Begy, V., Louppe, G.: Likelihood-free MCMC with amortized approximate ratio estimators. In: International Conference on Machine Learning, pp. 4239–4248. PMLR (2020)
25. Hermans, J., Delaunoy, A., Rozet, F., Wehenkel, A., Louppe, G.: Averting a crisis in simulation-based inference. arXiv preprint arXiv:2110.06581 (2021)
26. Holmes, C.C., Walker, S.G.: Assigning a value to a power likelihood in a general Bayesian model. Biometrika **104**(2), 497–503 (2017)
27. Jones-Todd, C.M., et al.: Identifying prognostic structural features in tissue sections of colon cancer patients using point pattern analysis: Point pattern analysis of colon cancer tissue sections. Stat. Med. **38**(8), 1421–1441 (2019). https://doi.org/10.1002/sim.8046
28. Knoblauch, J., Jewson, J., Damoulas, T.: Generalized variational inference: three arguments for deriving new posteriors. arXiv preprint arXiv:1904.02063 (2019)
29. von Krause, M., Radev, S.T., Voss, A.: Mental speed is high until age 60 as revealed by analysis of over a million participants. Nat. Hum. Behav. **6**(5), 700–708 (2022). https://doi.org/10.1038/s41562-021-01282-7
30. Leclercq, F.: Simulation-based inference of Bayesian hierarchical models while checking for model misspecification (2022). arXiv:2209.11057 [astro-ph, q-bio, stat]
31. Loaiza-Maya, R., Martin, G.M., Frazier, D.T.: Focused Bayesian prediction. J. Appl. Economet. **36**(5), 517–543 (2021)
32. Lotfi, S., Izmailov, P., Benton, G., Goldblum, M., Wilson, A.G.: Bayesian model selection, the marginal likelihood, and generalization. arXiv preprint arXiv:2202.11678 (2022)
33. Lueckmann, J.M., Boelts, J., Greenberg, D., Goncalves, P., Macke, J.: Benchmarking simulation-based inference. In: International Conference on Artificial Intelligence and Statistics, pp. 343–351. PMLR (2021)
34. Lueckmann, J.M., Boelts, J., Greenberg, D., Goncalves, P., Macke, J.: Benchmarking simulation-based inference. In: Banerjee, A., Fukumizu, K. (eds.) Proceedings of The 24th International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 130, pp. 343–351. PMLR (2021)
35. Lueckmann, J.M., Goncalves, P.J., Bassetto, G., Öcal, K., Nonnenmacher, M., Macke, J.H.: Flexible statistical inference for mechanistic models of neural dynamics. In: Advances in Neural Information Processing Systems, vol. 30 (2017)

36. Masegosa, A.: Learning under model misspecification: applications to variational and ensemble methods. In: Advances in Neural Information Processing Systems, vol. 33, pp. 5479–5491 (2020)
37. Matsubara, T., Knoblauch, J., Briol, F.X., Oates, C.J.: Robust generalised bayesian inference for intractable likelihoods (2022). arXiv:2104.07359 [math, stat]
38. Pacchiardi, L., Dutta, R.: Likelihood-free inference with generative neural networks via scoring rule minimization. arXiv preprint arXiv:2205.15784 (2022)
39. Pacchiardi, L., Dutta, R.: Score matched neural exponential families for likelihood-free inference (2022). arXiv:2012.10903 [stat]
40. Pang, G., Shen, C., Cao, L., Hengel, A.V.D.: Deep learning for anomaly detection: a review. ACM Comput. Surv. **54**(2), 1–38 (2022). https://doi.org/10.1145/3439950. arXiv:2007.02500 [cs, stat]
41. Papamakarios, G., Murray, I.: Fast ε-free inference of simulation models with Bayesian conditional density estimation. In: Advances in Neural Information Processing Systems, vol. 29 (2016)
42. Papamakarios, G., Pavlakou, T., Murray, I.: Masked autoregressive flow for density estimation. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
43. Papamakarios, G., Sterratt, D., Murray, I.: Sequential neural likelihood: fast likelihood-free inference with autoregressive flows. In: The 22nd International Conference on Artificial Intelligence and Statistics, pp. 837–848. PMLR (2019)
44. Radev, S.T., et al.: OutbreakFlow: model-based Bayesian inference of disease outbreak dynamics with invertible neural networks and its application to the COVID-19 pandemics in Germany. PLoS Comput. Biol. **17**(10), e1009472 (2021)
45. Radev, S.T., Mertens, U.K., Voss, A., Ardizzone, L., Köthe, U.: BayesFlow: learning complex stochastic models with invertible neural networks. IEEE Trans. Neural Netw. Learn. Syst. **33**, 1452–1466 (2020)
46. Ramesh, P., et al.: GATSBI: generative adversarial training for simulation-based inference. arXiv preprint arXiv:2203.06481 (2022)
47. Schmon, S.M., Cannon, P.W., Knoblauch, J.: Generalized posteriors in approximate Bayesian computation (2021). arXiv:2011.08644 [stat]
48. Shiono, T.: Estimation of agent-based models using Bayesian deep learning approach of BayesFlow. J. Econ. Dyn. Control **125**, 104082 (2021)
49. Säilynoja, T., Bürkner, P.C., Vehtari, A.: Graphical test for discrete uniformity and its applications in goodness of fit evaluation and multiple sample comparison (2021). arXiv:2103.10522 [stat]
50. Talts, S., Betancourt, M., Simpson, D., Vehtari, A., Gelman, A.: Validating Bayesian inference algorithms with simulation-based calibration (2020). arXiv:1804.06788 [stat]
51. Tejero-Cantero, A., et al.: SBI-a toolkit for simulation-based inference. arXiv preprint arXiv:2007.09114 (2020)
52. Thomas, O., Corander, J.: Diagnosing model misspecification and performing generalized Bayes' updates via probabilistic classifiers. arXiv preprint arXiv:1912.05810 (2019)
53. Vehtari, A., Ojanen, J.: A survey of Bayesian predictive methods for model assessment, selection and comparison. Stat. Surv. **6** (2012). https://doi.org/10.1214/12-SS102
54. Ward, D., Cannon, P., Beaumont, M., Fasiolo, M., Schmon, S.M.: Robust neural posterior estimation and statistical model criticism (2022). arXiv:2210.06564 [cs, stat]
55. White, H.: Maximum likelihood estimation of misspecified models. Econometrica **50**(1), 1–25 (1982)

56. Wiqvist, S., Frellsen, J., Picchini, U.: Sequential neural posterior and likelihood approximation. arXiv preprint arXiv:2102.06522 (2021)
57. Yang, J., Zhou, K., Li, Y., Liu, Z.: Generalized out-of-distribution detection: a survey. arXiv:2110.11334 (2021)
58. Zhang, F., Gao, C.: Convergence rates of variational posterior distributions. Ann. Stat. **48**(4), 2180–2207 (2020). https://doi.org/10.1214/19-AOS1883