5th Symposium on Advances in Approximate Bayesian Inference, 2023 1–28

Balancing Simulation-based Inference for Conservative Posteriors

Anonymous Authors
Anonymous Institution

Abstract

Conservative inference is a major concern in simulation-based inference. It has been shown that commonly used algorithms can produce overconfident posterior approximations. Balancing has empirically proven to be an effective way to mitigate this issue. However, its application remains limited to neural ratio estimation. In this work, we extend balancing to any algorithm that provides a posterior density. In particular, we introduce a balanced version of both neural posterior estimation and contrastive neural ratio estimation. We show empirically that the balanced versions tend to produce conservative posterior approximations on a wide variety of benchmarks. In addition, we provide an alternative interpretation of the balancing condition in terms of the χ^2 divergence.

1. Introduction

Simulation-based inference (SBI) (Cranmer et al., 2020) is a statistical inference framework that solves the inverse problem of identifying which parameter $\boldsymbol{\theta}$ generated observation $\boldsymbol{x}$ by approximating the posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ with a surrogate model $\hat{p}(\boldsymbol{\theta} \mid \boldsymbol{x})$. $\hat{p}$ is constructed from simulated pairs $(\boldsymbol{\theta}, \boldsymbol{x})$ produced by a generative model where the likelihood $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ is only implicitly defined. A classic method to produce samples from the surrogate is a rejection sampling technique called Approximate Bayesian Computation (Sisson et al., 2018). Recently, there has been significant development of algorithms using machine learning for estimating the posterior (Papamakarios and Murray, 2016; Lueckmann et al., 2017; Greenberg et al., 2019; Glöckler et al., 2021; Sharrock et al., 2022; Geffner et al., 2022), which we call Neural Posterior Estimation (NPE); the likelihood (Papamakarios et al., 2019b; Gratton, 2017); or the likelihood-to-evidence ratio (Thomas et al., 2016; Tran et al., 2017; Hermans et al., 2020; Durkan et al., 2020; Miller et al., 2021, 2022), which we call Neural Ratio Estimation (NRE).

It has been demonstrated that the estimated surrogate $\hat{p}(\boldsymbol{\theta} \mid \boldsymbol{x})$ can be more confident than $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ using common SBI algorithms (Hermans et al., 2022). This poses a problem for the reliability of SBI in a scientific setting where surrogates must be *conservative*, i.e. avoid inaccurately excluding parameters at a given credibility level. There has been development

© A. Authors.

in testing for overconfidence using empirical expected coverage and related methods (Cook et al., 2006; Talts et al., 2018; Hermans et al., 2022). Lemos et al. (2023) extend expected coverage testing to be a sufficient condition for posterior surrogate correctness, using only samples from the posterior surrogate. In an algorithmic approach to encourage conservativeness, Delaunoy et al. (2022) found that the so-called *balance condition* regularizes overconfidence in expectation in surrogates trained with NRE (Hermans et al., 2020). Zhao et al. (2021) rather test for valid local coverage. Linhart et al. (2022) focuses on normalizing flows and extends this method to the multivariate setting. In a similar fashion (Dalmasso et al., 2020, 2021; Masserano et al., 2022) aim to produce valid frequentist coverage.

Contribution After providing some background, we generalize the balancing condition to NPE methods and Contrastive Neural Ratio Estimation (NRE-C) (Miller et al., 2022). We provide empirical evidence of the regularizing effect of the balance condition in all of these settings and the first expected coverage tests of NRE-C. Additionally, we relate the balance condition to the χ^2-divergence and generalize it to NPE methods and NRE-C.

2. Background

Posterior estimation (NPE) A density estimator $q_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})$ with weights $\boldsymbol{w}$, such as a mixture density network (Bishop, 1994) or normalizing flow (Papamakarios et al., 2019a), approximates $p(\boldsymbol{\theta} \,|\, \boldsymbol{x})$ when the expected Kullback-Leibler divergence

$$\mathbb{E}_{p(\boldsymbol{x})} \left[\mathrm{KL}(p(\boldsymbol{\theta} \,|\, \boldsymbol{x}) \,|\, q_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})] \right], \tag{1}$$

is minimized. In NPE, the surrogate model is directly $\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x}) \coloneqq q_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})$.

Ratio estimation (NRE) The likelihood-to-evidence ratio $r(\boldsymbol{\theta}, \boldsymbol{x}) \coloneqq \frac{p(\boldsymbol{x} \,|\, \boldsymbol{\theta})}{p(\boldsymbol{x})} = \frac{p(\boldsymbol{\theta}, \boldsymbol{x})}{p(\boldsymbol{\theta}) p(\boldsymbol{x})}$ is estimated through a supervised learning task using classifier $\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})$. The target conditional distribution $\pi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})$ comes from $\pi(\boldsymbol{\theta}, \boldsymbol{x}, y) \coloneqq \pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y)\pi(y)$ where the marginals are set to $\pi(y = 0) \coloneqq \pi(y = 1) \coloneqq \frac{1}{2}$ and the remaining conditional is defined as

$$\pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y) \coloneqq \begin{cases} p(\boldsymbol{\theta})p(\boldsymbol{x}) & y = 0 \\ p(\boldsymbol{\theta}, \boldsymbol{x}) & y = 1 \end{cases}. \tag{2}$$

The classifier $\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \coloneqq \sigma \circ f_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x})$ is parameterized by a neural network $f_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x})$ and σ is the sigmoid. $\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})$ approximates $\pi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})$ when the NRE loss

$$\mathbb{E}_{\pi(\boldsymbol{\theta}, \boldsymbol{x})} \Big[\mathrm{KL}(\pi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \,\|\, \varpi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x})) \Big], \tag{3}$$

is minimized. Let $\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x}) \coloneqq \frac{\exp \circ f_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x})}{Z_{\boldsymbol{w}}(\boldsymbol{x})} p(\boldsymbol{\theta})$, with $Z_{\boldsymbol{w}}(\boldsymbol{x}) \coloneqq \int \exp \circ f_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x}) p(\boldsymbol{\theta}) d\boldsymbol{\theta}$, define the surrogate model. When the loss is zero $f_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x}) = \log r(\boldsymbol{\theta}, \boldsymbol{x})$, recovering the posterior.

Contrastive Neural Ratio Estimation (Miller et al., 2022) introduces NRE-C, an algorithm featuring a flexible, multiclass distribution $\tilde{\pi}(y)$ for $y = 0, 1, \ldots, K$. When the objective (17) becomes zero, the NRE-C surrogate model recovers the posterior. Details about the method are in Appendix B. NRE-C represents a strict generalization of classifier-based, likelihood-to-evidence ratio estimation methods (Hermans et al., 2020; Durkan et al., 2020).

Conservative surrogates Undesirably, simulation-based inference algorithms can produce overconfident surrogate models (Hermans et al., 2022). We define overconfidence in terms of the $(1-\alpha)$ *expected coverage probability* of the posterior surrogate $\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x})$,

$$1 - \hat{\alpha}[\hat{p}; \alpha] \coloneqq \mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})} \left[\mathbb{1}(\boldsymbol{\theta} \in \Theta_{\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x})}(1-\alpha)) \right], \tag{4}$$

where $\mathbb{1}$ is an indicator function, and $\Theta_{\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x})}(1-\alpha)$ yields the $(1-\alpha)$ highest posterior density region (HPDR) of $\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x})$ with $\alpha \in [0, 1]$. The quantity $(1-\alpha)$ is called the *nominal coverage probability*. When $\exists \alpha' : 1 - \hat{\alpha}[\hat{p}; \alpha'] < 1 - \alpha'$, we say that $\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x})$ is *overconfident*. Overconfidence is problematic because the surrogate tends to exclude parameter values that are actually plausible at the considered credibility level. On the other hand, extremely *underconfident* surrogates are not informative. Although there is a tradeoff, scientific applications take a cautious approach by favoring underconfidence. Therefore, we encourage conservative surrogates at credibility level α', which have $1 - \hat{\alpha}[\hat{p}; \alpha'] \geq 1 - \alpha'$. One surrogate is more conservative than another when there are more credibility levels at which it is conservative. We show the prior $p(\boldsymbol{\theta})$ is conservative in Appendix A.

Balance condition In an effort to produce conservative surrogates, Delaunoy et al. (2022) introduced the *balance condition*. It holds for any classifier $\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})$ that satisfies

$$1 = \mathbb{E}_{p(\boldsymbol{\theta})p(\boldsymbol{x})} \left[\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \right] + \mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})} \left[\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \right]. \tag{5}$$

Delaunoy et al. (2022) show that $\mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})} \left[\frac{\pi(y=1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})}{\varpi(y=1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})} \right] \geq 1$ and $\mathbb{E}_{p(\boldsymbol{\theta})p(\boldsymbol{x})} \left[\frac{\pi(y=0 \,|\, \boldsymbol{\theta}, \boldsymbol{x})}{\varpi(y=0 \,|\, \boldsymbol{\theta}, \boldsymbol{x})} \right] \geq 1$ for *balanced* classifiers. In expectation, it implies the balanced classifier's probabilities $\varpi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x})$ are closer to uniform than the target, encouraging $\hat{r}(\boldsymbol{\theta}, \boldsymbol{x})$ to be closer to 1. This brings to surrogate closer to the prior, $p(\boldsymbol{\theta})$, which is conservative.

3. Extending the balance condition

We clarify and generalize the balance criterion: Identifying it with the χ^2-divergence, and applying it to models that can evaluate the (unnormalized) approximate posterior density.

Balance as divergence We encourage balance during training by regularizing the loss with a Lagrange multiplier; penalizing solutions that do not satisfy the balance criterion

$$B[\varpi] \coloneqq B(\boldsymbol{w}) \coloneqq \left(\mathbb{E}_{p(\boldsymbol{\theta})p(\boldsymbol{x})} \left[\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \right] + \mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})} \left[\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \right] - 1 \right)^2, \tag{6}$$

where $\boldsymbol{w}$ are the classifier weights. The effects of balance are clearer when (6) is rewritten in the form of a χ^2-divergence. The χ^2-divergence (Sason and Verdú, 2016) is defined as

$$\chi^2(\pi(y) \,\|\, \varpi(y)) \coloneqq \int \left(\frac{\varpi(y)}{\pi(y)} - 1 \right)^2 \pi(y)\, dy, \tag{7}$$

where $\varpi(y) \coloneqq \int \varpi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \pi(\boldsymbol{\theta}, \boldsymbol{x}) d\boldsymbol{\theta}\, d\boldsymbol{x}$ is the marginal classifier. With intermediate steps shown in (18), we identify that $B[\varpi] = \chi^2(\pi(y) \,\|\, \varpi(y))$! Therefore, a balanced classifier satisfies $\varpi(y) = \pi(y)$, i.e. balancing regularizes the marginal classifier towards the target distribution for y. We explore the balance condition for a multiclass y in Appendix C.

Balance criterion for alternative models The balance criterion regularizes marginal distribution $\varpi(y)$; this makes sense for NRE which defines $\varpi(y)$ and target marginal $\pi(y)$. However, we are interested in regularizing objectives $L(\boldsymbol{w})$ that either do not introduce a binary auxiliary variable y, or use an alternative. In order to apply the balance criterion, we propose to use the same target distribution $\pi(\boldsymbol{\theta}, \boldsymbol{x}, y)$ as NRE does and define a classifier in terms of the variational (unnormalized) posterior approximant $\hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})$. We approximate $r(\boldsymbol{\theta}, \boldsymbol{x}) \coloneqq \frac{p(\boldsymbol{\theta}, \boldsymbol{x})}{p(\boldsymbol{\theta}) p(\boldsymbol{x})} = \frac{p(\boldsymbol{\theta} \,|\, \boldsymbol{x})}{p(\boldsymbol{\theta})}$ with $\frac{\hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})}{p(\boldsymbol{\theta})}$ which yields the classifier $\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}; \hat{q}_{\boldsymbol{w}}) \coloneqq \frac{\hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})/p(\boldsymbol{\theta})}{1 + \hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})/p(\boldsymbol{\theta})}$ and regularize $L(\boldsymbol{w})$ with $B(\boldsymbol{w})$ and Lagrange multiplier λ

$$L(\boldsymbol{w}) + \lambda B(\boldsymbol{w}). \tag{8}$$

The main contribution is reformulating $B(\boldsymbol{w})$ to be expressed in terms of $\hat{q}_{\boldsymbol{w}}$, which generalizes the balance condition to models which allow for approximate density evaluation! We consider losses $L(\boldsymbol{w})$ that go to zero if and only if $\hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x}) = p(\boldsymbol{\theta} \,|\, \boldsymbol{x})$. When this is true, the balance condition (5) also holds since $B(\boldsymbol{w})$ becomes zero:

$$\begin{aligned} B(\boldsymbol{w}) &\coloneqq \left(\int (\pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y = 0) + \pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y = 1)) \varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}; \hat{q}_{\boldsymbol{w}}) d\boldsymbol{\theta}\, d\boldsymbol{x} - 1 \right)^2 \\ &= \left(\int (p(\boldsymbol{\theta}) p(\boldsymbol{x}) + p(\boldsymbol{\theta}, \boldsymbol{x})) \frac{\hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})/p(\boldsymbol{\theta})}{1 + \hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})/p(\boldsymbol{\theta})} d\boldsymbol{\theta}\, d\boldsymbol{x} - 1 \right)^2 \end{aligned} \tag{9}$$

That means for loss $L(\boldsymbol{w}) = 0$ the regularized version (8) also goes to zero; just like BNRE.

Balanced Neural Posterior Estimation (BNPE) We propose BNPE which regularizes NPE's objective (1) with the balance criterion to train a normalized density estimator. We have $\hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x}) \coloneqq q_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})$. The corresponding classifier becomes $\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}; \hat{q}_{\boldsymbol{w}}) \coloneqq \frac{q_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})/p(\boldsymbol{\theta})}{1 + q_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x})/p(\boldsymbol{\theta})}$. Training minimizes (8), substituting the appropriate loss and classifier.

Balanced Contrastive Neural Ratio Estimation (BNRE-C) We propose BNRE-C which regularizes NRE-C's objective (17) with the balance criterion to train a ratio estimator. The density estimator is $\hat{q}_{\boldsymbol{w}}(\boldsymbol{\theta} \,|\, \boldsymbol{x}) \coloneqq \exp \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x}) p(\boldsymbol{\theta})$. The corresponding classifier becomes $\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}; \hat{q}) \coloneqq \frac{\exp \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x}) p(\boldsymbol{\theta})/p(\boldsymbol{\theta})}{1 + \exp \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x}) p(\boldsymbol{\theta})/p(\boldsymbol{\theta})} = \sigma \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x})$. Training minimizes (8), substituting the appropriate loss and classifier.

4. Experiments

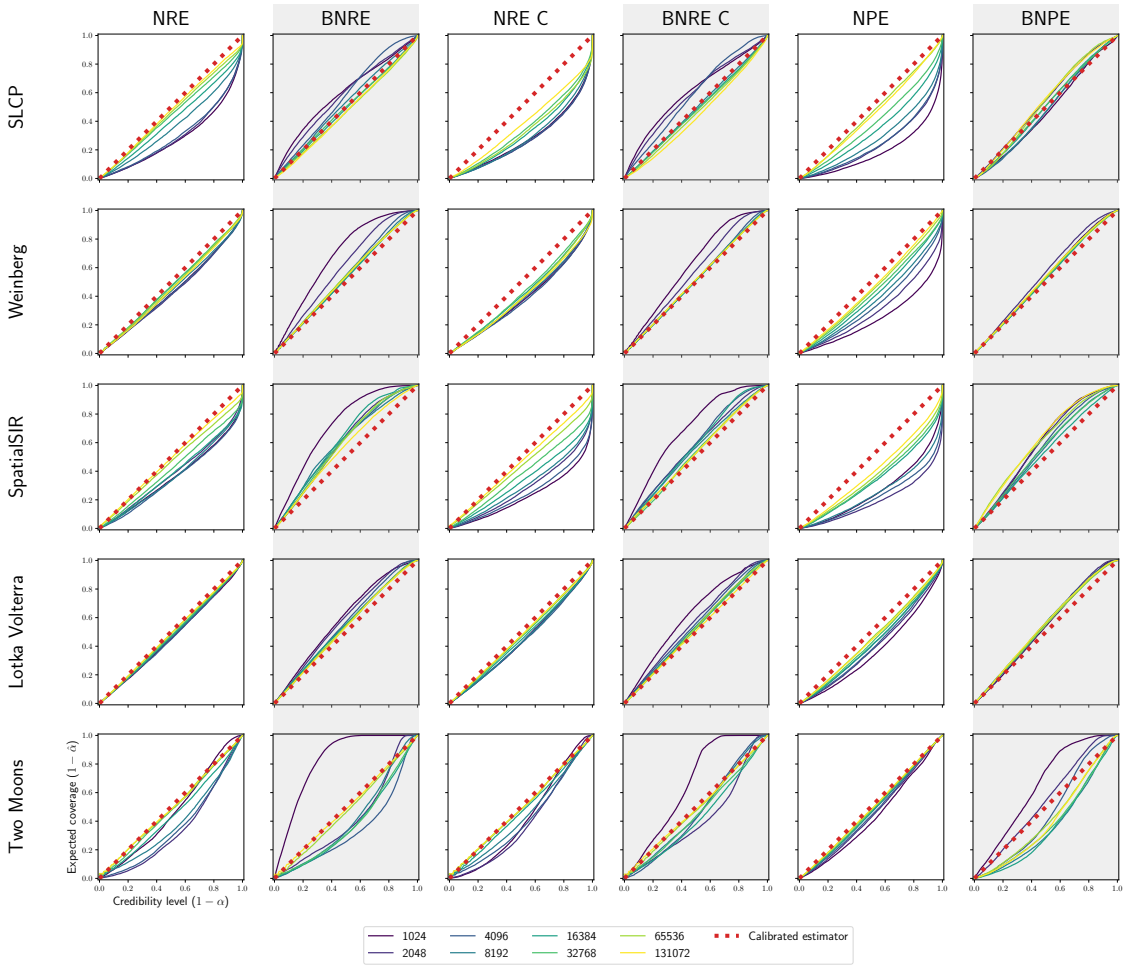

Figure 1: Empirical expected coverage for increasing simulation budgets. A perfectly calibrated surrogate has an expected coverage probability equal to the nominal coverage probability and produces a diagonal line. A conservative surrogate has an expected coverage curve at or above the diagonal line. An overconfident surrogate produces curves below the diagonal line. Balanced algorithms tend to produce conservative surrogates. 5 runs are performed for each simulation budget with the median $\hat{\alpha}$ at each nominal credibility reported.

We investigate whether our proposed algorithms BNPE and BNRE-C lead to more conservative surrogate models compared to their non-balanced counterparts. Our diagnostics test the empirical expected coverage probability (4) of considered algorithms trained with ($\lambda > 0$) and without ($\lambda = 0$) the balance criterion. We also benchmark BNRE (Delaunoy et al., 2022), an algorithm that trains a neural network to minimize (8) using (3) as $L(\boldsymbol{w})$. The empirical expected coverage tests are performed on a set of benchmarks of varying difficulty. Simulator descriptions can be found in Appendix D and neural network architectures in Appendix E. Expected coverage for these benchmarks are shown in Figure 1.

For both BNPE and BNRE-C, enforcing the balance criterion tends to produce conservative surrogates on most of the benchmarks, just as it did with BNRE. The only exception was Two Moons, which consistently produced overconfident surrogates for all algorithms. Taken as a whole, this provides some evidence that regularizing models allowing posterior density evaluation with the balance criterion makes them more conservative. Our results suggest that enforcing the balance criterion did not negatively impact the informativeness of the surrogate for higher simulation budgets. This is quantified in Appendix F where we estimate the information contained in the surrogates and the empirical balance error. A qualitative analysis of the posteriors is provided in Appendix G.

We observed that normalizing flows struggle to minimize the balance criterion on the SLCP and Weinberg benchmarks for low simulation budgets. Normalizing flows must learn the prior as a multiplicative component in their density estimate, while likelihood-to-evidence ratio methods use the ground truth prior to construct the surrogate. We hypothesize that this difference may play a role in the observed behavior. We discuss this in Appendix H and empirically reduce balance error by initializing the flow close to the prior.

5. Conclusions

In this work, we have shown that balancing can be applied to any simulation-based inference algorithm that yields a posterior density estimator, hence extending the applicability of balancing. We have also shown that the balancing condition can be expressed as a χ^2 divergence. We believe that this reformulation is a stepping stone towards a better understanding of the balancing condition and could inspire novel algorithms.

Let us note that there exist algorithms that do not directly provide the posterior density and hence do not fall into this framework. Some algorithms only aim to provide an unnormalized posterior approximation. This is the case of algorithms that model the likelihood (Papamakarios et al., 2019b) or an unnormalized version of the likelihood-to-evidence ratio (Durkan et al., 2020). A direct consequence is that those algorithms are not necessarily balanced at optimum. Enforcing the balancing condition may hence prevent reaching the optimal solution. Some algorithms also only allow sampling from the posterior surrogate. This is the case of score-based methods (Song et al., 2020; Geffner et al., 2022). In such a setting, the balancing condition cannot be computed. Future work would then include a reformulation of the balancing condition that uses only samples from the posterior surrogate.

It should also be noted that enforcing that the marginal classification matches the prior over classes is something that can be extended to more than two classes. The χ^2 formulation of the balancing condition could hence be extended to multi-class classification methods such as Durkan et al. (2020) and Miller et al. (2022) as shown in Appendix C. Whether the effect of this regularization on the posterior would be the same as in the two-classes setting remains to be studied and is left for future work.

References

Christopher M Bishop. Mixture density networks. *Technical Report*, 1994.

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.

Kyle Cranmer, Lukas Heinrich, Tim Head, and Gilles Louppe. "Active Sciencing" with Reusable Workflows. https://github.com/cranmer/active_sciencing, 2017.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proc. Natl. Acad. Sci. U. S. A.*, May 2020.

Niccolo Dalmasso, Rafael Izbicki, and Ann Lee. Confidence sets and hypothesis testing in a likelihood-free inference setting. In *International Conference on Machine Learning*, pages 2323–2334. PMLR, 2020.

Niccolo Dalmasso, David Zhao, Rafael Izbicki, and Ann B Lee. Likelihood-free frequentist inference: Bridging classical statistics and machine learning in simulation and uncertainty quantification. *arXiv preprint arXiv:2107.03920*, 2021.

Arnaud Delaunoy, Joeri Hermans, François Rozet, Antoine Wehenkel, and Gilles Louppe. Towards reliable simulation-based inference with balanced neural ratio estimation. In *Advances in Neural Information Processing Systems*, 2022.

S. Dragomir. An upper bound for the csiszar f-divergence in terms of the variational distance and applications. *Panamerican Mathematical Journal*, 12, 01 2002.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. *Advances in neural information processing systems*, 32, 2019.

Conor Durkan, Iain Murray, and George Papamakarios. On contrastive learning for likelihood-free inference. In *International Conference on Machine Learning*, pages 2771–2781. PMLR, 2020.

Tomas Geffner, George Papamakarios, and Andriy Mnih. Score modeling for simulation-based inference. *arXiv preprint arXiv:2209.14249*, 2022.

Alison L. Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International Statistical Review / Revue Internationale de Statistique*, 70(3):419–435, 2002. ISSN 03067734, 17515823. URL http://www.jstor.org/stable/1403865.

Manuel Glöckler, Michael Deistler, and Jakob H Macke. Variational methods for simulation-based inference. In *International Conference on Learning Representations*, 2021.

Steven Gratton. Glass: A general likelihood approximate solution scheme. *arXiv preprint arXiv:1708.08479*, 2017.

David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic posterior transformation for likelihood-free inference. In *International Conference on Machine Learning*, pages 2404–2414. PMLR, 2019.

Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free mcmc with amortized approximate ratio estimators. In *International Conference on Machine Learning*, pages 4239–4248. PMLR, 2020.

Joeri Hermans, Arnaud Delaunoy, François Rozet, Antoine Wehenkel, Volodimir Begy, and Gilles Louppe. A crisis in simulation-based inference? beware, your posterior approximations can be unfaithful. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=LHAbHkt6Aq.

Pablo Lemos, Adam Coogan, Yashar Hezaveh, and Laurence Perreault-Levasseur. Sampling-based accuracy testing of posterior estimators for general inference. *arXiv preprint arXiv:2302.03026*, 2023.

Julia Linhart, Alexandre Gramfort, and Pedro LC Rodrigues. Validation diagnostics for sbi algorithms based on normalizing flows. *arXiv preprint arXiv:2211.09602*, 2022.

Alfred J Lotka. Analytical note on certain rhythmic relations in organic systems. *Proceedings of the National Academy of Sciences*, 6(7):410–415, 1920.

Jan-Matthis Lueckmann, Pedro J Gonçalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H Macke. Flexible statistical inference for mechanistic models of neural dynamics. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1289–1299, 2017.

Luca Masserano, Tommaso Dorigo, Rafael Izbicki, Mikael Kuusela, and Ann B Lee. Simulation-based inference with waldo: Perfectly calibrated confidence regions using any prediction or posterior estimation algorithm. *arXiv preprint arXiv:2205.15680*, 2022.

Benjamin K Miller, Alex Cole, Patrick Forré, Gilles Louppe, and Christoph Weniger. Truncated marginal neural ratio estimation. *Advances in Neural Information Processing Systems*, 34:129–143, 2021.

Benjamin Kurt Miller, Christoph Weniger, and Patrick Forré. Contrastive neural ratio estimation. *arXiv preprint arXiv:2210.06170*, 2022.

George Papamakarios and Iain Murray. Fast ε-free inference of simulation models with bayesian conditional density estimation. *Advances in neural information processing systems*, 29, 2016.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *arXiv preprint arXiv:1912.02762*, 2019a.

George Papamakarios, David Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 837–848. PMLR, 2019b.

François Rozet. Zuko, 10 2022. URL https://pypi.org/project/zuko.

Igal Sason and Sergio Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11):5973–6006, 2016.

Louis Sharrock, Jack Simons, Song Liu, and Mark Beaumont. Sequential neural score estimation: Likelihood-free inference with conditional score based diffusion models. *arXiv preprint arXiv:2210.04872*, 2022.

Scott A Sisson, Yanan Fan, and Mark Beaumont. *Handbook of approximate Bayesian computation*. CRC Press, 2018.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

F.E. Su. *Methods for Quantifying Rates of Convergence for Random Walks on Groups*. UMI Dissertation services. Harvard University, 1995. URL https://books.google.nl/books?id=d7d0NAAACAAJ.

Sean Talts, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. Validating bayesian inference algorithms with simulation-based calibration. *arXiv preprint arXiv:1804.06788*, 2018.

Owen Thomas, Ritabrata Dutta, Jukka Corander, Samuel Kaski, Michael U Gutmann, et al. Likelihood-free inference by ratio estimation. *Bayesian Analysis*, 2016.

Dustin Tran, Rajesh Ranganath, and David M Blei. Hierarchical implicit models and likelihood-free variational inference. *arXiv preprint arXiv:1702.08896*, 2017.

Vito Volterra. Fluctuations in the abundance of a species considered mathematically. *Nature*, 118(2972):558–560, 1926.

David Zhao, Niccolò Dalmasso, Rafael Izbicki, and Ann B Lee. Diagnostics for conditional density models and bayesian inference algorithms. In *Uncertainty in Artificial Intelligence*, pages 1830–1840. PMLR, 2021.

Appendix A. The Prior is Conservative

In this section, we prove that the prior $p(\boldsymbol{\theta})$ is conservative at credibility level α. We begin by gathering the necessary facts and definitions. First, we give a definition of the function

$$\Theta_{\hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x})}(1-\alpha) \coloneqq \bar{\Theta}_\alpha \in \arg\min_{\bar{\Theta}} \int_{\bar{\Theta}} d\boldsymbol{\theta} \qquad \text{s.t.} \qquad \int_{\bar{\Theta}} \hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x}) d\boldsymbol{\theta} = 1-\alpha \tag{10}$$

which yields the $(1-\alpha)$ HPDR of $\hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x})$, denoted $\bar{\Theta}_\alpha$, with $\alpha \in [0,1]$ and $\bar{\Theta}$ denoting a measure space. Recall, the $(1-\alpha)$ expected coverage probability of surrogate $\hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x})$:

$$1-\hat{\alpha}[\hat{p};\alpha] \coloneqq \mathbb{E}_{p(\boldsymbol{\theta},\boldsymbol{x})}\left[\mathbb{1}(\boldsymbol{\theta} \in \Theta_{\hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x})}(1-\alpha))\right] = \int_{\bar{\Theta}\times\bar{\mathcal{X}}} p(\boldsymbol{\theta},\boldsymbol{x})\mathbb{1}(\boldsymbol{\theta} \in \Theta_{\hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x})}(1-\alpha)) d\boldsymbol{\theta}\, d\boldsymbol{x},$$

which was defined in (4), and where $\bar{\mathcal{X}}$ denotes another measure space. Finally, we say that posterior surrogate $\hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x})$ is conservative at credibility level α when

$$1-\hat{\alpha}[\hat{p};\alpha] \geq 1-\alpha. \tag{11}$$

For the proof we let $\hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x}) \coloneqq p(\boldsymbol{\theta})$. The $(1-\alpha)$ expected coverage probability of $p(\boldsymbol{\theta})$ is

$$\begin{aligned} 1-\hat{\alpha}[p(\boldsymbol{\theta});\alpha] &= \int_{\bar{\Theta}\times\bar{\mathcal{X}}} p(\boldsymbol{\theta},\boldsymbol{x})\mathbb{1}(\boldsymbol{\theta} \in \Theta_{p(\boldsymbol{\theta})}(1-\alpha)) d\boldsymbol{\theta}\, d\boldsymbol{x} \\ &= \int_{\bar{\Theta}} p(\boldsymbol{\theta})\mathbb{1}(\boldsymbol{\theta} \in \Theta_{p(\boldsymbol{\theta})}(1-\alpha)) d\boldsymbol{\theta} \\ &= \int_{\bar{\Theta}_\alpha} p(\boldsymbol{\theta}) d\boldsymbol{\theta}, \end{aligned} \tag{12}$$

where we first integrated over $\boldsymbol{x}$ then applied the indicator function to the bounds of integration, reducing to $\bar{\Theta}_\alpha$ by (10). We substitute the region of integration into the constraint for our surrogate $p(\boldsymbol{\theta})$ and recover

$$\int_{\bar{\Theta}_\alpha} \hat{p}(\boldsymbol{\theta}\,|\,\boldsymbol{x}) d\boldsymbol{\theta} = \int_{\bar{\Theta}_\alpha} p(\boldsymbol{\theta}) d\boldsymbol{\theta} = 1-\alpha. \tag{13}$$

Our constraint is equal to the last line of (12), implying

$$1-\hat{\alpha}[p(\boldsymbol{\theta});\alpha] = 1-\alpha \geq 1-\alpha, \tag{14}$$

which means that $p(\boldsymbol{\theta})$ has exact coverage at every credibility level α, which is conservative.

Appendix B. Contrastive Neural Ratio Estimation

Here we introduce a summary of NRE-C (Miller et al., 2022) in our notation and indicate the specific hyperparameter choice in the experiments. First, we define the target distribution for the supervised learning problem. We let marginal distribution $\tilde{\pi}(y)$ with $y \in \{0, 1, \ldots, K\}$ have probabilities $\tilde{\pi}(y = k) \coloneqq \tilde{\pi}_K$ for all $k \geq 1$ and $\tilde{\pi}(y = 0) \coloneqq \tilde{\pi}_0$ which yields the relationship $\tilde{\pi}_0 = 1 - K\tilde{\pi}_K$. The remaining conditional is

$$\tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x} \,|\, y = k) \coloneqq \begin{cases} p(\boldsymbol{\theta}_1) \cdots p(\boldsymbol{\theta}_K) p(\boldsymbol{x}) & k = 0 \\ p(\boldsymbol{\theta}_1) \cdots p(\boldsymbol{\theta}_K) p(\boldsymbol{x} \,|\, \boldsymbol{\theta}_k) & k = 1, \ldots, K \end{cases}, \tag{15}$$

where $\boldsymbol{\Theta} \coloneqq (\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K)$ are contrastive parameters, sampled from the prior $p(\boldsymbol{\theta})$. We fit the variational, multiclass classifier

$$\tilde{\varpi}(y = k \,|\, \boldsymbol{\Theta}, \boldsymbol{x}) \coloneqq \begin{cases} \frac{K}{K + \sum_{i=1}^{K} \exp \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}_i, \boldsymbol{x})} & k = 0 \\ \frac{\exp \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}_k, \boldsymbol{x}))}{K + \sum_{i=1}^{K} \exp \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}_i, \boldsymbol{x})} & k = 1, \ldots, K \end{cases} \tag{16}$$

by minimizing $\mathbb{E}_{\tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x})} \left[\mathrm{KL}(\tilde{\pi}(y \,|\, \boldsymbol{\Theta}, \boldsymbol{x}) \,|\, \tilde{\varpi}(y \,|\, \boldsymbol{\Theta}, \boldsymbol{x})) \right]$, where $h_{\boldsymbol{w}}$ is a neural network parameterized by weights $\boldsymbol{w}$. We write the loss function out explicitly:

$$\begin{aligned} L[\tilde{\varpi}] \coloneqq &- \frac{1}{1+\gamma} \mathbb{E}_{\tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x} \,|\, y=0)} \left[\log \tilde{\varpi}(y = 0 \,|\, \boldsymbol{\Theta}, \boldsymbol{x}) \right] \\ &- \frac{\gamma}{1+\gamma} \mathbb{E}_{\tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x} \,|\, y=K)} \left[\log \tilde{\varpi}(y = K \,|\, \boldsymbol{\Theta}, \boldsymbol{x}) \right], \end{aligned} \tag{17}$$

where we introduced $\gamma \coloneqq \frac{\tilde{\pi}(y \geq 1)}{\tilde{\pi}(y=0)} = \frac{K\tilde{\pi}_K}{\tilde{\pi}_0}$. There are only two terms in this sum because we exploited symmetries in the conditionals $\tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x} \,|\, y = k)$ when $k \neq 0$. We define the surrogate model as $\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x}) \coloneqq \frac{\exp \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x})}{Z_{\boldsymbol{w}}(\boldsymbol{x})} p(\boldsymbol{\theta})$ with $Z_{\boldsymbol{w}}(\boldsymbol{x}) \coloneqq \int \exp \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x}) p(\boldsymbol{\theta}) \, d\boldsymbol{\theta}$.

In the experiments we took $\gamma \coloneqq 1$ and $K \coloneqq 5$.

In Section 3, we introduce the balance criterion to NRE-C's loss function. We emphasize here that the $B(\boldsymbol{w}) \coloneqq B[\varpi]$ term is a functional of the *binary classifier* $\varpi \coloneqq \sigma \circ h_{\boldsymbol{w}}(\boldsymbol{\theta}, \boldsymbol{x})$ parameterized by the same neural network as the multiclass $\tilde{\varpi}$.

Appendix C. Balance condition in terms of f-divergences

Balance and χ^2-divergence χ^2 is an f-divergence, as defined in (7). The steps to show the connection between χ^2 and the balancing condition are as follows:

$$\begin{aligned}
\chi^2(\pi(y) \,\|\, \varpi(y)) &\coloneqq \int \left(\frac{\varpi(y)}{\pi(y)} - 1\right)^2 \pi(y)\, dy \\
&= \left(\frac{\varpi(y=0)}{\pi(y=0)} - 1\right)^2 \pi(y=0) + \left(\frac{\varpi(y=1)}{\pi(y=1)} - 1\right)^2 \pi(y=1) \\
&= \frac{(\varpi(y=0) - \pi(y=0))^2}{\pi(y=0)} + \frac{(\varpi(y=1) - \pi(y=1))^2}{\pi(y=1)} \\
&= \frac{(\varpi(y=1) - \pi(y=1))^2}{\pi(y=0)} + \frac{(\varpi(y=1) - \pi(y=1))^2}{\pi(y=1)} \\
&= (\varpi(y=1) - \pi(y=1))^2 \left(\frac{1}{\pi(y=0)} + \frac{1}{\pi(y=1)}\right) \\
&= 4(\varpi(y=1) - \pi(y=1))^2 \\
&= (2\varpi(y=1) - 1)^2 \\
&= \left(2 \int \varpi(y=1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \pi(\boldsymbol{\theta}.\boldsymbol{x}) d\boldsymbol{\theta}\, d\boldsymbol{x} - 1\right)^2 \\
&= \left(2 \int \varpi(y=1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \left(\int \pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y) \pi(y) dy\right) d\boldsymbol{\theta}\, d\boldsymbol{x} - 1\right)^2 \\
&= \left(\int \varpi(y=1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) (\pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y=0) + \pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y=1)) d\boldsymbol{\theta}\, d\boldsymbol{x} - 1\right)^2 \\
&= \left(\mathbb{E}_{p(\boldsymbol{\theta})p(\boldsymbol{x})}\left[\varpi(y=1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})\right] + \mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})}\left[\varpi(y=1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})\right] - 1\right)^2
\end{aligned} \tag{18}$$

where we used $\pi(y=0) = \pi(y=1) = \frac{1}{2}$ and $\varpi(\boldsymbol{\theta}, \boldsymbol{x}) \coloneqq \pi(\boldsymbol{\theta}, \boldsymbol{x})$.

Kullback–Leibler divergence chain rule An important result from Delaunoy et al. (2022) was that the optimal classifier is balanced. It can be seen from the chain rule:

$$\begin{aligned}
\mathrm{KL}(\pi(\boldsymbol{\theta}, \boldsymbol{x}, y) \,\|\, \varpi(\boldsymbol{\theta}, \boldsymbol{x}, y)) &= \underbrace{\mathrm{KL}\left(\pi(\boldsymbol{\theta}, \boldsymbol{x}) \,\|\, \varpi(\boldsymbol{\theta}, \boldsymbol{x})\right)}_{=0} + \mathbb{E}_{\pi(\boldsymbol{\theta}, \boldsymbol{x})}\Big[\mathrm{KL}(\pi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \,\|\, \varpi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x})\Big] \\
&= \mathrm{KL}\left(\pi(y) \,\|\, \varpi(y)\right) + \mathbb{E}_{\pi(y)}\Big[\mathrm{KL}(\pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y) \,\|\, \varpi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y)\Big],
\end{aligned} \tag{19}$$

since $\varpi(\boldsymbol{\theta}, \boldsymbol{x}) \coloneqq \pi(\boldsymbol{\theta}, \boldsymbol{x})$. Minimizing the left hand side of this equation also minimizes the right hand side with a one-to-one trade off between terms. When $\varpi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x}) = \pi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x})$, the left left hand side of the equation becomes zero, implying that every term on the right hand side becomes zero as well. The balance condition enforced $\varpi(y) = \pi(y)$, which holds when the left hand side is zero, due to $\mathrm{KL}\left(\pi(y) \,\|\, \varpi(y)\right) = 0$; therefore, $\pi(y \,|\, \boldsymbol{\theta}, \boldsymbol{x})$ is balanced and enforcing the balance condition does *not* exclude this optimal classifier, i.e. the target distribution, from our solution set!

Consider the effects of including the balance condition in the loss function. Given Lagrange multiplier λ, the loss becomes

$$L[\varpi] + \lambda B[\varpi] = \mathbb{E}_{\pi(\boldsymbol{\theta},\boldsymbol{x})}\Big[\mathrm{KL}(\pi(y\,|\,\boldsymbol{\theta},\boldsymbol{x})\,\|\,\varpi(y\,|\,\boldsymbol{\theta},\boldsymbol{x})\Big] + \lambda\chi^2\left(\pi(y)\,\|\,\varpi(y)\right). \tag{20}$$

The balance criterion *effectively* puts extra weight on the "balance term" in the unregularized objective (19). This statement is not strictly true because the functional form of KL and χ^2 are different; however, they both become zero when $\varpi(y) = \pi(y)$ and increase when $\varpi(y)$ becomes increasingly "different" from $\pi(y)$. The first claim is based on the inequality $\mathrm{KL}(\pi\|\varpi) \leq \chi^2(\pi\|\varpi)$, which is shown in (25). The second is a property of divergences.

Explicit BNRE loss definition Together this yields our regularized sample approximation of the loss for BNRE under combination of the objective and regularizer

$$\begin{aligned}
\ell(\boldsymbol{w}) &= \mathbb{E}_{\pi(\boldsymbol{\theta},\boldsymbol{x})}\Big[\mathrm{KL}(\pi(y\,|\,\boldsymbol{\theta},\boldsymbol{x})\,\|\,\varpi(y\,|\,\boldsymbol{\theta},\boldsymbol{x})\Big] + \lambda\chi^2\left(\pi(y)\,\|\,\varpi(y)\right) \\
&\approx -\frac{1}{2B}\left[-\sum_{b=1}^{B}\log\Big(1-\sigma\circ h_{\boldsymbol{w}}(\boldsymbol{\theta}^{(b)},\boldsymbol{x}^{(b)})\Big) + \sum_{b'=1}^{B}\log\Big(\sigma\circ h_{\boldsymbol{w}}(\boldsymbol{\theta}^{(b')},\boldsymbol{x}^{(b')})\Big)\right] \\
&\quad + \lambda\left[\sum_{b=1}^{B}\sigma\circ h_{\boldsymbol{w}}(\boldsymbol{\theta}^{(b)},\boldsymbol{x}^{(b)}) + \sum_{b'=1}^{B}\sigma\circ h_{\boldsymbol{w}}(\boldsymbol{\theta}^{(b')},\boldsymbol{x}^{(b')}) - 1\right]^2,
\end{aligned} \tag{21}$$

where $\boldsymbol{\theta}^{(b)},\boldsymbol{x}^{(b)} \sim p(\boldsymbol{\theta})p(\boldsymbol{x})$ and $\boldsymbol{\theta}^{(b')},\boldsymbol{x}^{(b')} \sim p(\boldsymbol{\theta},\boldsymbol{x})$.

Balance as a Kullback-Liebler divergence Rather than using the balance criterion, it is also possible to apply the Kullback-Liebler divergence to regularize $\varpi(y)$:

$$\begin{aligned}
\mathrm{KL}(\pi(y)\,\|\,\varpi(y)) &\coloneqq \int \pi(y)\log\frac{\pi(y)}{\varpi(y)}\,dy \\
&= \frac{1}{2}\int\left(\log\pi(y=0) - \log\varpi(y=0) + \log\pi(y=1) - \log\varpi(y=1)\right)\,dy \\
&= -\frac{1}{2}\int\left(\log\varpi(y=0) + \log\varpi(y=1)\right)\,dy - \log 2 \\
&= -\frac{1}{2}\int\Bigg[\log\left(\int\varpi(y=0\,|\,\boldsymbol{\theta},\boldsymbol{x})\left(\int\pi(\boldsymbol{\theta},\boldsymbol{x}\,|\,y)\pi(y)dy\right)d\boldsymbol{\theta}\,d\boldsymbol{x}\right) \\
&\quad + \log\left(\int\varpi(y=1\,|\,\boldsymbol{\theta},\boldsymbol{x})\left(\int\pi(\boldsymbol{\theta},\boldsymbol{x}\,|\,y)\pi(y)dy\right)d\boldsymbol{\theta}\,d\boldsymbol{x}\right)\Bigg]\,dy - \log 2 \\
&= -\frac{1}{2}\int\Bigg[\log\left(\frac{1}{2}\int\varpi(y=0\,|\,\boldsymbol{\theta},\boldsymbol{x})\left(\pi(\boldsymbol{\theta},\boldsymbol{x}\,|\,y=0) + \pi(\boldsymbol{\theta},\boldsymbol{x}\,|\,y=1)\right)d\boldsymbol{\theta}\,d\boldsymbol{x}\right) \\
&\quad + \log\left(\frac{1}{2}\int\varpi(y=1\,|\,\boldsymbol{\theta},\boldsymbol{x})\left(\pi(\boldsymbol{\theta},\boldsymbol{x}\,|\,y=0) + \pi(\boldsymbol{\theta},\boldsymbol{x}\,|\,y=1)\right)d\boldsymbol{\theta}\,d\boldsymbol{x}\right)\Bigg]\,dy \\
&\quad - \log 2.
\end{aligned} \tag{22}$$

The objective is motivated information theoretically because it appears in the chain rule for the Kullback-Leibler divergence, like in (19), but it is unappealing from an optimization perspective: The log of the integrals leads to biased empirical estimates based-on samples.

Kullback-Leibler and χ^2 divergences For probability measures P and Q on measure space $\mathcal{X}$, the χ^2 divergence is

$$\chi^2(P \,\|\, Q) \coloneqq \int_{\mathcal{X}} \left(\frac{dP}{dQ} - 1\right)^2 dQ = \int_{\mathcal{X}} \left[\left(\frac{dP}{dQ}\right)^2 - 1\right] dQ. \tag{23}$$

Therefore,

$$\mathrm{KL}(P \,\|\, Q) = \int_{\mathcal{X}} \ln \frac{dP}{dQ} dP \leq \int_{\mathcal{X}} \left(\frac{dP}{dQ} - 1\right) dP = \int_{\mathcal{X}} \left[\left(\frac{dP}{dQ}\right)^2 - 1\right] dQ \tag{24}$$

where we used the concavity of the logarithm $\ln x \leq x - 1, \forall x > 0$. This proves

$$\mathrm{KL}(P \,\|\, Q) \leq \chi^2(P \,\|\, Q), \quad \forall P \ll Q. \tag{25}$$

Alternative proofs and other relevant inequalities can be found in the materials Sason and Verdú (2016); Dragomir (2002); Su (1995); Gibbs and Su (2002). The consequence of this is that if we minimize the χ^2-divergence, we will Kullback-Leibler divergence as well.

Balancing multiclass objectives Since the balance condition can be specified as a divergence, we can define a multiclass balance conditions for ratio estimators featuring arbitrary multiclass y variables such as those introduced by Durkan et al. (2020) and Miller et al. (2022). Additionally, the χ^2-divergence formulation enables regularization of classifiers fitting to $\tilde{\pi}(y \,|\, \boldsymbol{\Theta}, \boldsymbol{x})$ that have arbitrary marginal distributions $\tilde{\pi}(y)$. We make no statements about the effects changing $\tilde{\pi}(y)$ might have on the estimated posterior $\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x})$.

$$
\begin{aligned}
&\chi^2(\tilde{\pi}(y) \,\|\, \tilde{\varpi}(y)) \coloneqq \int \left(\frac{\tilde{\varpi}(y)}{\tilde{\pi}(y)} - 1 \right)^2 \tilde{\pi}(y)\, dy \\
&= \sum_{i=0}^{K} \left(\frac{\tilde{\varpi}(y=i)}{\tilde{\pi}(y=i)} - 1 \right)^2 \tilde{\pi}(y=i) \\
&= \sum_{i=0}^{K} \frac{(\tilde{\varpi}(y=i) - \tilde{\pi}(y=i))^2}{\tilde{\pi}(y=i)} \\
&= (K+1) \sum_{i=0}^{K} \left(\tilde{\varpi}(y=i) - \frac{1}{K+1} \right)^2 \qquad \text{(uniform } \tilde{\pi}(y)) \\
&= \frac{1}{K+1} \sum_{i=0}^{K} \left((K+1)\tilde{\varpi}(y=i) - 1 \right)^2 \\
&= \frac{1}{K+1} \sum_{i=0}^{K} \left(\int (K+1)\tilde{\varpi}(y=i \,|\, \boldsymbol{\Theta}, \boldsymbol{x}) \left(\sum_{j=0}^{K} \tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x} | y=j) \tilde{\pi}(y=j) \right) d\boldsymbol{\Theta} d\boldsymbol{x} - 1 \right)^2 \\
&= \frac{1}{K+1} \sum_{i=0}^{K} \left(\int \tilde{\varpi}(y=i \,|\, \boldsymbol{\Theta}, \boldsymbol{x}) \left(\sum_{j=0}^{K} \tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x} \,|\, y=j) \right) d\boldsymbol{\Theta}\, d\boldsymbol{x} - 1 \right)^2 \qquad \text{(uniform } \tilde{\pi}(y))
\end{aligned}
\tag{26}
$$

where $\tilde{\varpi}(\boldsymbol{\Theta}, \boldsymbol{x}, y) \coloneqq \tilde{\varpi}(y \,|\, \boldsymbol{\Theta}, \boldsymbol{x}) \tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x})$ and $\tilde{\varpi}(y \,|\, \boldsymbol{\Theta}, \boldsymbol{x})$ is a variational multiclass classifier. Some problems may offer optimization using symmetries in the definition of the sampling distribution $\tilde{\pi}(\boldsymbol{\Theta}, \boldsymbol{x} \,|\, y=j)$. Here we followed the NRE-C convention that $y = 0, 1, \ldots, K$.

We want to note here that we did not apply (26) in our experiments, rather we applied (6) even to the applications featuring a multiclass y. The analysis about the effectiveness of a multiclass regularizer versus the binary balance criterion are left for future work.

Appendix D. Benchmarks description

The *SLCP* simulator models a fictive problem with 5 parameters. The observable $\boldsymbol{x}$ is composed of 8 scalars which represent the 2D-coordinates of 4 points. The coordinate of each point is sampled from the same multivariate Gaussian whose mean and covariance matrix are parametrized by $\boldsymbol{\theta}$. We consider an alternative version of the original task (Papamakarios et al., 2019b) by inferring the marginal posterior density of 2 of those parameters. In contrast to its original formulation, the likelihood is not tractable due to the marginalization.

The *Weinberg* problem (Cranmer et al., 2017) concerns a simulation of high energy particle collisions $e^+e^- \to \mu^+\mu^-$. The angular distributions of the particles can be used to measure the Weinberg angle $\boldsymbol{x}$ in the standard model of particle physics, leading to an observable composed of 20 scalars. From the scattering angle, we are interested in inferring Fermi's constant $\boldsymbol{\theta}$.

The *Spatial SIR* model (Hermans et al., 2022) involves a grid-world of susceptible, infected, and recovered individuals. Based on initial conditions and the infection and recovery rate $\boldsymbol{\theta}$, the model describes the spatial evolution of an infection. The observable $\boldsymbol{x}$ is a snapshot of the grid-world after some fixed amount of time. The grid used is of size 50 by 50.

The *Lotka-Volterra* population model (Lotka, 1920; Volterra, 1926) describes a process of interactions between a predator and a prey species. The model is conditioned on 4 parameters $\boldsymbol{\theta}$ which influence the reproduction and mortality rate of the predator and prey species. We infer the marginal posterior of the predator parameters from time series of 2001 steps representing the evolution of both populations over time. The specific implementation is based on a Markov Jump Process as in Papamakarios et al. (2019b).

The *Two Moons* (Greenberg et al., 2019) simulator models a fictive problem with 2 parameters. The observable $\boldsymbol{x}$ is composed of 2 scalars which represent the 2D-coordinates of a random point sampled from a crescent-shaped distribution shifted and rotated around the origin depending on the parameters' values. Those transformations involve the absolute value of the sum of the parameters leading to a second crescent in the posterior and hence making it multi-modal.

Appendix E. Architectures and hyper-parameters

Table 1 summarizes the architectures and hyper-parameters used for each benchmark. The architectures are separated into two parts: the embedding and the head networks. The embedding network compresses the observable into a set of features. The head network then uses those features concatenated with the parameters to produce the target quantity. The head can either be a classifier or a normalizing flow depending on the algorithm considered. The learning rate is scheduled during training. Table 1 provides the initial learning rates.

Those are then divided by 10 each time no improvement was observed on the validation loss for 10 epochs.

	SLCP	Weinberg	Lotka-V.	Spatial SIR	Two Moons
Embedding network	None	None	CNN	CNN	None
Embedding layers	/	/	8	8	/
Embedding channels	/	/	8	16	/
Convolution type	/	/	Conv1D	Conv2D	/
Classifier Head	MLP	MLP	MLP	MLP	MLP
Classifier layers	6	6	6	6	6
Classifier hidden neurons	256	256	256	256	256
Flow Head	NSF	NSF	NSF	NSF	NSF
Flow layers	3	3	3	3	3
Flow hidden neurons	256	256	256	256	256
Learning rate	0.001	0.001	0.001	0.001	0.001
Epochs	500	500	500	500	500
Batch size	256	256	256	256	256
λ *(balanced algorithms)*	100	100	100	100	100
γ *(NRE-C)*	1	1	1	1	1
K *(NRE-C)*	5	5	5	5	5

Table 1: Architectures and training hyper-parameters

Appendix F. Additional experiments

In this section, we provide the expected coverage, nominal log posterior and balancing error for all the algorithm/benchmark pairs considered in this paper. We have added the Ratio Neural Posterior Estimation algorithm (RNPE) which corresponds to a normalizing flow trained as a ratio estimator obtained from the following transformation

$$\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}; \hat{p}) := \frac{\hat{r}(\boldsymbol{\theta}, \boldsymbol{x})}{1 + \hat{r}(\boldsymbol{\theta}, \boldsymbol{x})} = \frac{\hat{p}(\boldsymbol{\theta}, \boldsymbol{x})/p(\boldsymbol{\theta})}{1 + \hat{p}(\boldsymbol{\theta}, \boldsymbol{x})/p(\boldsymbol{\theta})}. \tag{27}$$

The expected coverage is defined as

$$\mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})} \left[\mathbb{1}(\boldsymbol{\theta} \in \Theta_{\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x})}(1 - \alpha)) \right],$$

and is shown in Figure 2.

The nominal log posterior is defined as

$$\mathbb{E}_{p(\boldsymbol{\theta}, \boldsymbol{x})} \left[\log \hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{x}) \right],$$

and is shown in Figure 3. We observe that balanced algorithms have a lower nominal log posterior than their corresponding non-balanced algorithms. However, they have similar nominal log posterior as the simulation budget increases.

The balancing error is defined as

$$\begin{aligned} &\left| \int (p(\boldsymbol{\theta})p(\boldsymbol{x}) + p(\boldsymbol{\theta}, \boldsymbol{x}))\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) \, d\boldsymbol{\theta} \, d\boldsymbol{x} - 1 \right| \\ = &\left| \int (p(\boldsymbol{\theta})p(\boldsymbol{x}) + p(\boldsymbol{\theta}, \boldsymbol{x})) \frac{\hat{p}(\boldsymbol{\theta}|\boldsymbol{x})/p(\boldsymbol{\theta})}{1 + \hat{p}(\boldsymbol{\theta}|\boldsymbol{x})/p(\boldsymbol{\theta})} \, d\boldsymbol{\theta} \, d\boldsymbol{x} - 1 \right|, \end{aligned}$$

and is shown in Figure 4. Without surprise, we observe that enforcing the balancing condition leads to more balanced surrogates. Non-balanced algorithms show a high balancing error for low simulation budgets but this balancing error diminishes as the simulation budget gets higher. This is due to the fact that the posterior surrogate gets closer to the true posterior as the simulation increases and that the true posterior is always balanced.

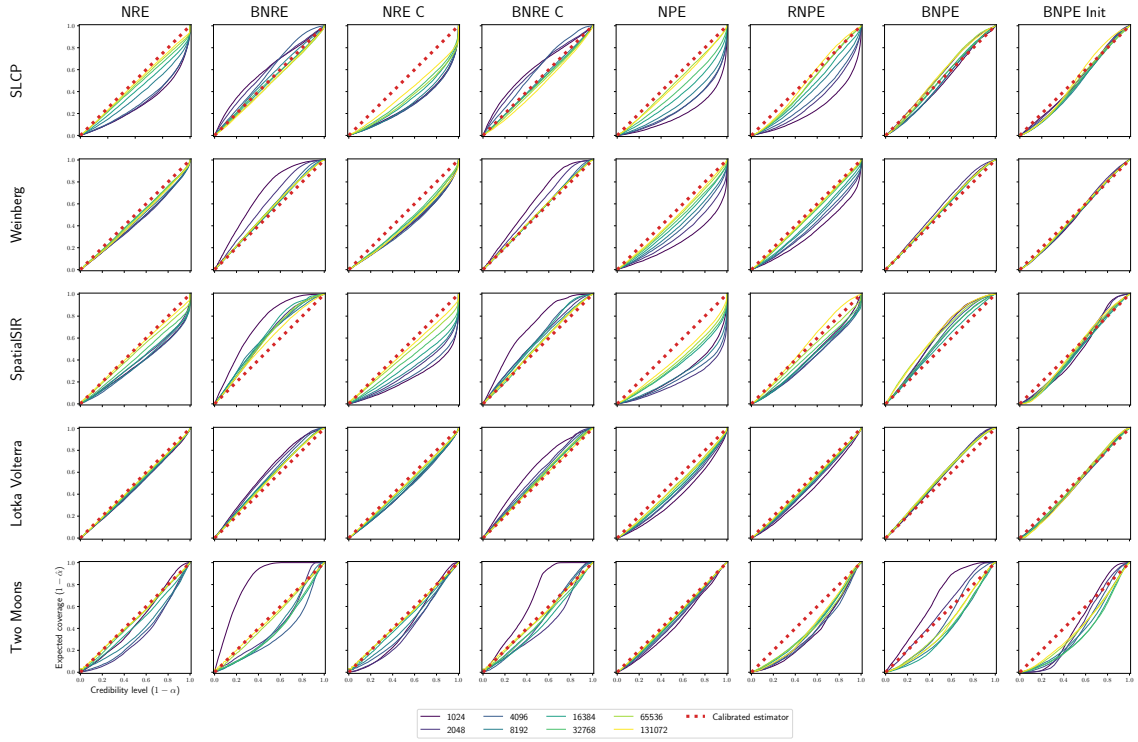

Figure 2: Expected coverage for increasing simulation budgets. A perfectly calibrated posterior has an expected coverage probability equal to the nominal coverage probability and hence produces a diagonal line. A conservative estimator has an expected coverage curve at or above the diagonal line, while an overconfident estimator produces curves below the diagonal line. 5 runs are performed for each simulation budget and the median is reported.

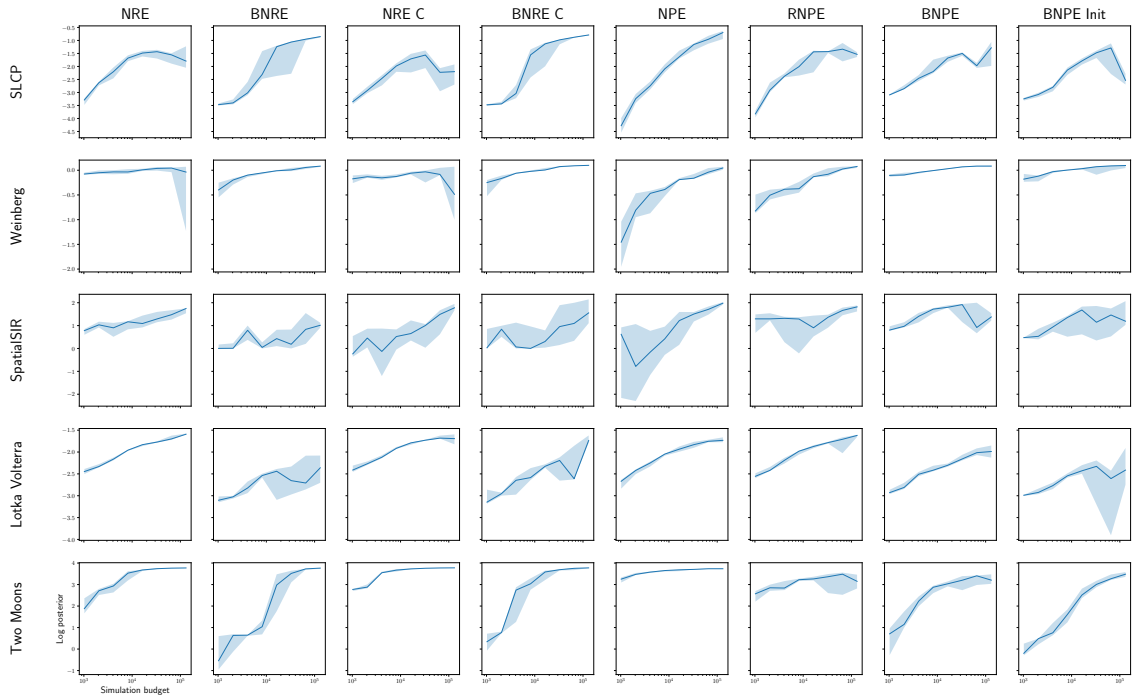

Figure 3: Nominal log posterior for increasing simulation budgets. 5 runs are performed for each simulation budget. Solid lines represent the median and the shaded areas represent the minimum and maximum. Larger values are desirable.

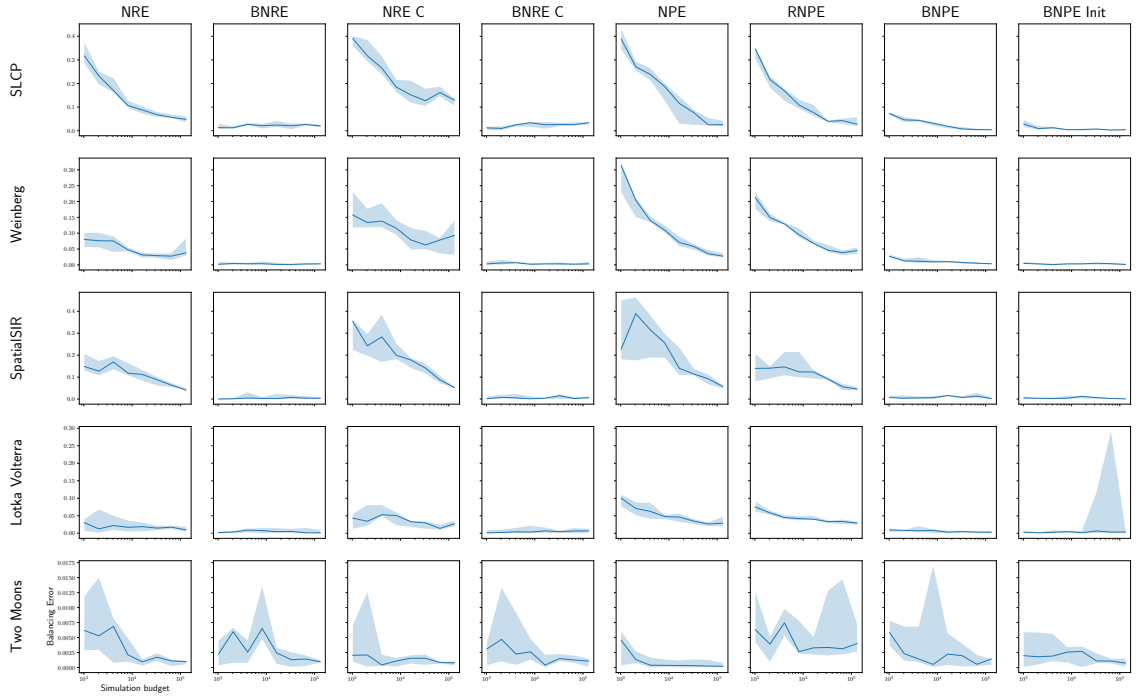

Figure 4: Balancing error for increasing simulation budgets. 5 runs are performed for each simulation budget. Solid lines represent the median and the shaded areas represent the minimum and maximum. Smaller values are desirable.

Appendix G. Qualitative posterior analysis

In this section, we provide a visualization of the posteriors obtained with the different algorithms and simulation budgets. A visualization of the SLCP benchmark which is provided in Figure 5 and the two moons benchmark is shown in Figure 6. We observe that balanced versions of the algorithms indeed produce larger posterior approximations than non-balanced ones for low simulation budgets. Balanced versions then include the nominal parameter more often. Nevertheless, Balanced algorithms' approximate posteriors shrink as the simulation budget increases. BNRE and BNRE-C recover similar posterior surrogates as their non-balanced versions. However, BNPE leads to larger posteriors on the SLCP benchmark. Let us note that the architectures used for classifier-based and flow-based methods are different. Using a more flexible flow architecture would certainly lead to narrower posterior approximations as the simulation budget increases.

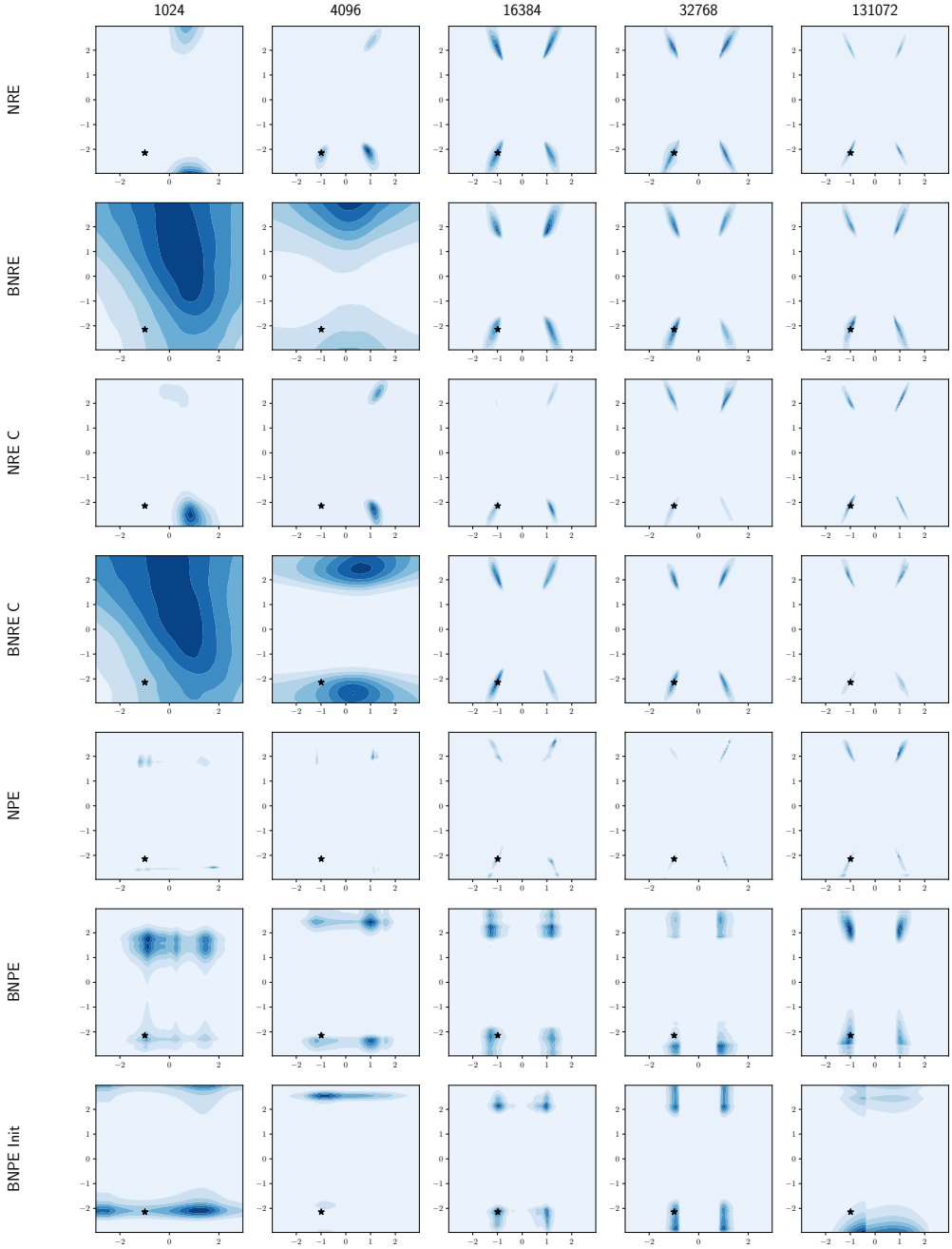

Figure 5: Visualization of posteriors obtained with different algorithms and simulation budgets on the SLCP benchmark. The blue areas represent the posterior and the black star represents the nominal parameters used for generating the observation.

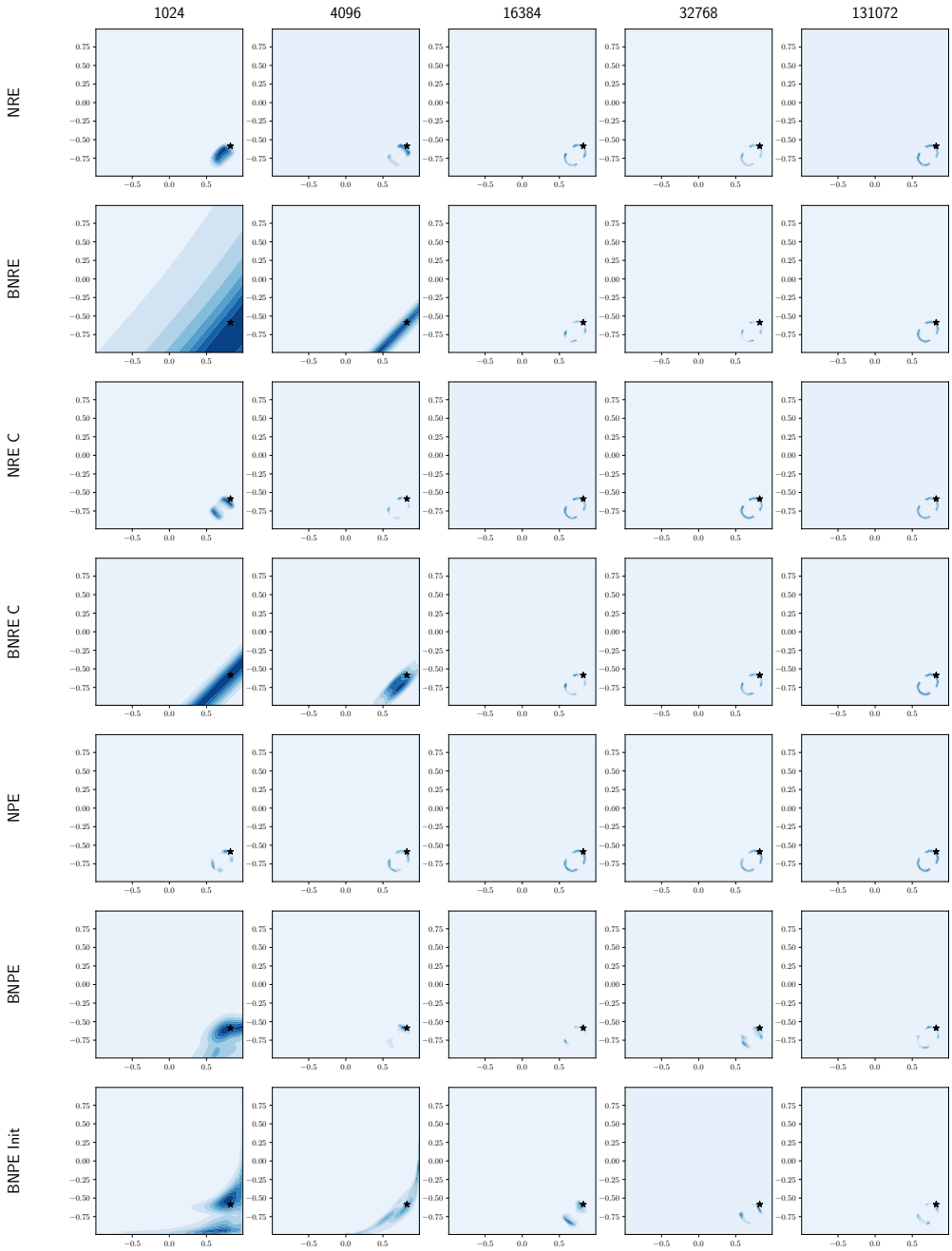

Figure 6: Visualization of posteriors obtained with different algorithms and simulation budgets on the two moons benchmark. The blue areas represent the posterior and the black star represents the nominal parameters used for generating the observation.

Appendix H. Refining Balanced Neural Posterior Estimation

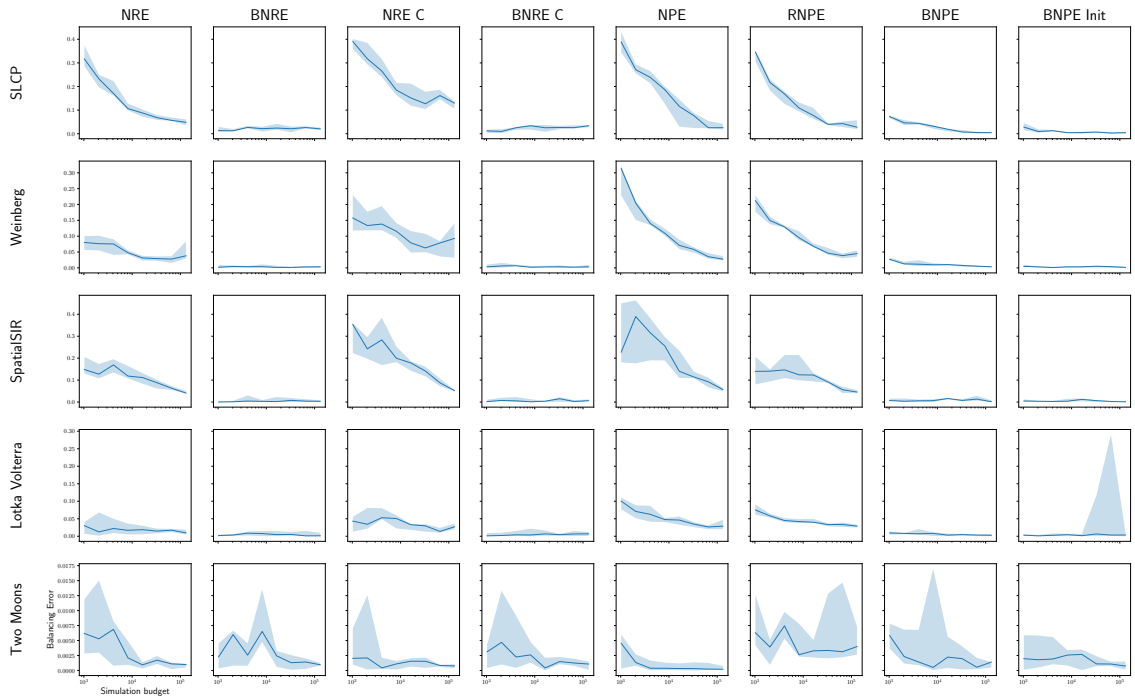

Figure 7: Comparison of different algorithms in terms balancing error $\left|\int (p(\boldsymbol{\theta})p(\boldsymbol{x}) + p(\boldsymbol{\theta}, \boldsymbol{x}))\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})\, d\boldsymbol{\theta}\, d\boldsymbol{x} - 1\right|$. 5 runs are performed for each simulation budget. Solid lines represent the median and the shaded areas represent the minimum and maximum. Lower values are desirable. Note: This is the same as Figure 4.

Figure 7 shows the balancing error defined as

$$\left|\int (p(\boldsymbol{\theta})p(\boldsymbol{x}) + p(\boldsymbol{\theta}, \boldsymbol{x}))\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x})\, d\boldsymbol{\theta}\, d\boldsymbol{x} - 1\right|.$$

We observe that while BNRE and BNRE-C show a lower balancing error on all the benchmarks, BNPE sometimes has a high balancing error. This suggests that BNPE sometimes struggles to learn to be balanced. In this section, we provide a way to simplify learning.

H.1. Improving Balanced Neural Posterior Estimation

Let us first observe that the prior $p(\boldsymbol{\theta})$ is balanced:

$$\int (\pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y = 0) + \pi(\boldsymbol{\theta}, \boldsymbol{x} \,|\, y = 1)) \varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) d\boldsymbol{\theta}\, d\boldsymbol{x} \tag{28}$$

$$= \int (p(\boldsymbol{\theta})p(\boldsymbol{x}) + p(\boldsymbol{\theta}, \boldsymbol{x})) \frac{\hat{p}(\boldsymbol{\theta}|\boldsymbol{x})/p(\boldsymbol{\theta})}{1 + \hat{p}(\boldsymbol{\theta}|\boldsymbol{x})/p(\boldsymbol{\theta})} d\boldsymbol{\theta}\, d\boldsymbol{x} \tag{29}$$

$$= \int (p(\boldsymbol{\theta})p(\boldsymbol{x}) + p(\boldsymbol{\theta}, \boldsymbol{x})) \frac{p(\boldsymbol{\theta})/p(\boldsymbol{\theta})}{1 + p(\boldsymbol{\theta})/p(\boldsymbol{\theta})} d\boldsymbol{\theta}\, d\boldsymbol{x} = 1 \tag{30}$$

BNRE can easily model the prior as this is achieved for $\varpi(y = 1 \,|\, \boldsymbol{\theta}, \boldsymbol{x}) = 0.5 \quad \forall \boldsymbol{\theta}, \boldsymbol{x}$. In opposition, BNPE has to explicitly learn a balanced distribution. In order to ease the task of learning a balanced distribution, we propose to initialize the posterior surrogate to the prior distribution. In the case of a normalizing flow, this can be achieved by initializing all the transformations to the identity function and either using the prior as base distribution or adding a transform that maps the base distribution to the prior at the end of the flow. We will refer to this algorithm as Initialized Balanced Neural Posterior Estimation (BNPE Init). In this work, we use Neural Spline Flows (Durkan et al., 2019). We discuss how to initialize this architecture to the prior in Appendix H.2.

It can be observed from Figure 7 that initializing the posterior surrogate to the prior indeed leads to a lower balancing error. Let us note that although the posterior surrogate is initialized to the prior, with a high enough simulation budget, it is able to learn a balanced surrogate that carries information about the parameter as seen from the log posterior quantity in Figure 3.

H.2. Intializing neural spline flows to the prior distribution

A normalizing flow models a complex distribution as a sequence of transformations of some base distribution. Consequently, the flow models the prior in the two following scenarios:

- the transformations are all identity transformations and the base distribution is the prior, or
- the transformations are all identity transformations and a fixed transformation that maps the base distribution to the prior is added at the end of the flow.

In the following, we describe how to obtain the two core components: identity transformations and a transformation that maps the base distribution to the prior. We then discuss the advantages and drawbacks of the two methods. In the experiments, we used a transformation from the base distribution to the prior.

Neural spline transformations (Durkan et al., 2019) are transformations defined by K rational-quadratic functions, with boundaries set by $K+1$ knots denoted by $(x^{(k)}, y^k)_{k=0}^K$. The boundaries are defined by $(x^{(0)}, y^0) = (-B, -B)$ and $(x^{(K)}, y^K) = (B, B)$. The knots are parametrized by two vectors θ^w and θ^h of length K. Those vectors are passed through a softmax and multiplied by $2B$ to define the bins' width and height.

The rational-quadratic in the kth bin is defined as

$$\frac{\alpha^{(k)}(\xi)}{\beta^{(k)}(\xi)} = y^{(k)} + \frac{(y^{(k+1)} - y^{(k)})[s^{(k)}\xi^2 + \delta^{(k)}\xi(1-\xi)]}{s^{(k)} + [\delta^{(k+1)} + \delta^{(k)} - 2s^{(k)}]\xi(1-\xi)}. \tag{31}$$

The terms $\left\{\delta^{(k)}\right\}_{k=1}^{K-1}$ define the derivatives at the internal points and are parametrized by the vector θ^d of length $K-1$. The other terms are defined as $s_k = (y^{k+1} - y^k)/(x^{k+1} - x^k)$ and $\xi = (x - x^k)/(x^{k+1} - x^k)$.

Initializing transformations to identity To let the spline transformation be close to identity, we need to initialize the knots such that all the weights and widths are the same. This can be done by initializing the vectors θ^w and θ^h to zeros for all conditionings. In addition, all $s^{(k)}$ are then equal to 1. We also need $\delta^{(k)} = 1,\ \forall k$. In the implementation used (Rozet, 2022), the vector θ^d models the log derivatives and hence must be initialized to a vector full of zeros.

To achieve the identity transform, the outputs of the neural network θ^h, θ^w and θ^d must be zeros at initialization for all conditionings. This is achieved when biases and weights are all set 0. However, this initialization is not optimal for learning via stochastic gradient descent, so we aim for a trade-off between ease of training and closeness to the prior. We initialize the neural network using some standard initialization, set the biases to 0 and divide the weights by 5 to obtain a function close to an identity transformation.

Mapping the base distribution to the prior In our case, the base distribution is a normal distribution $\mathcal{N}(0,1)$ and the priors are uniform in all the benchmarks considered. Therefore, we need to define a mapping from a Uniform distribution $\mathcal{U}(a,b)$, which p.d.f. is denoted by $p_u(u)$, to a normal distribution $\mathcal{N}(0,1)$, which p.d.f. is denoted by $p_n(n)$ and c.d.f denoted $F_n(n)$. Let us first consider an intermediate mapping to a Uniform distribution $\mathcal{U}(0,1)$ which p.d.f is denoted $p_{\tilde{u}}(\tilde{u})$. Going from u to $\tilde{u}$ can be achieved with the following transformation

$$\tilde{u} = \frac{u+a}{b-a}. \tag{32}$$

The Jacobian linked to this transformation is then

$$\frac{1}{b-a} \tag{33}$$

Going from $\tilde{u}$ to n can be achieved with the following transformation

$$\tilde{u} = F_n(n) \Leftrightarrow n = F_n^{-1}(\tilde{u}). \tag{34}$$

The Jacobian linked to this transformation is then

$$\left|(F_n^{-1})'(\tilde{u})\right| = \left|\frac{1}{F_n'(F_n^{-1}(\tilde{u}))}\right| \tag{35}$$

$$= \frac{1}{p_n(F_n^{-1}(\tilde{u}))}. \tag{36}$$

Comparison of the different initialization schemes We have considered two initialization schemes: using the prior as base distribution or adding a transformation that maps the base distribution to the prior distribution. Using the prior distribution as base distribution has the advantage to be easy to implement as it does not require defining a mapping between both distributions. However, we have empirically observed that modifying the base distribution can lead to worse performance. We hypothesize that this is due to the fact that all the considered benchmarks have a uniform prior while flows work better with a Gaussian base. Let us note that using such a transformation could be beneficial in the more general setting as it solves leakage! Leakage is the fact that NPE algorithms can lead to a posterior surrogate that has density outside of the prior support. This is a problem in sequential settings where this surrogate is used as a proposal for simulating new data points. The transformation from a normal distribution to the prior is a transformation that maps any distribution with infinite support to a distribution that has the same support as the prior. Therefore, leakage cannot happen when such a transformation is applied.