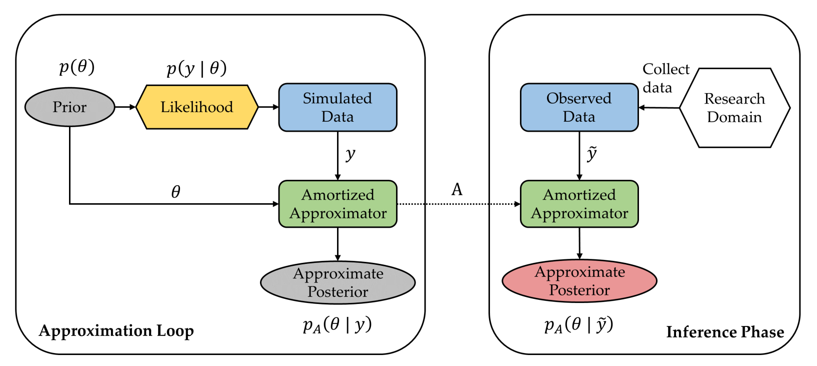
Robust Amortized Bayesian Inference with Self-Consistency Losses on Unlabeled Data

https://arxiv.org/abs/2501.13483

Aayush Mishra, Daniel Habermann, Marvin Schmitt, Stefan T. Radev, Paul-Christian Bürkner

Amortized Bayesian Inference



In the following, I use x for data and $q(\theta \mid x)$ for the neural approximator

Standard neural posterior estimation (NPE)

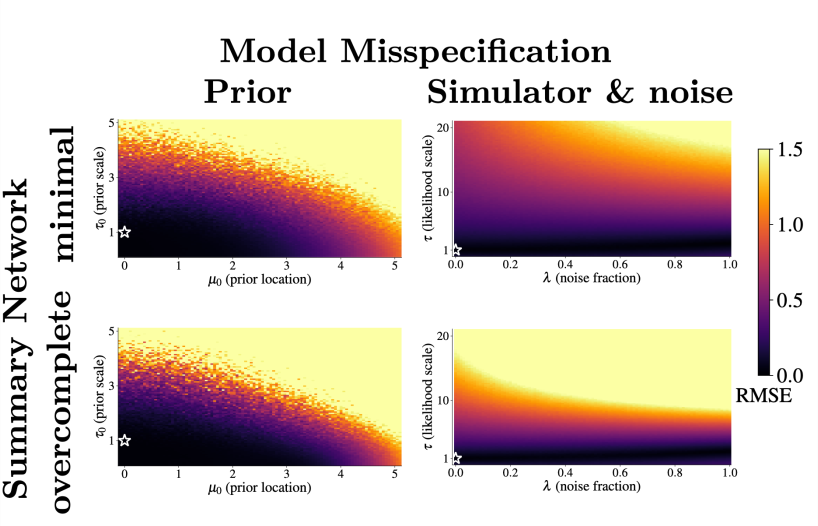
General form of (standard) NPE losses in SBI:

$$\mathsf{NPELoss(q)} = \mathbb{E}_{(\theta,x)\sim p(\theta,x)}\left[S(q(\theta \mid x), \theta)\right]$$

For normalizing flows with invertible neural networks:

$$\mathsf{NPELoss(q)} = \mathbb{E}_{(\theta,x)\sim p(\theta,x)}\left[-\log q(\theta \mid x)\right]$$

Standard NPE on misspecified models fails



Source: https://arxiv.org/abs/2406.03154

Bayesian Self-consistency

For any set of parameter values $\theta^{(1)}, \dots, \theta^{(L)}$, the following holds:

$$p(x) = \frac{p(x \mid \theta^{(1)})\, p(\theta^{(1)})}{p(\theta^{(1)} \mid x)} = \dots = \frac{p(x \mid \theta^{(L)})\, p(\theta^{(L)})}{p(\theta^{(L)} \mid x)}.$$

This implies that the variance of the log-ratios must be zero:

$$\mathsf{Var}_{l=1}^{L} \left[\log \left(\frac{p(x \mid \theta^{(l)})\, p(\theta^{(l)})}{p(\theta^{(l)} \mid x)} \right) \right] = 0$$

Our intial paper on Bayesian self-consistency:
https://arxiv.org/abs/2310.04395

Bayesian self-consistency loss

Replace the true posterior $p(\theta \mid x)$ with the neural approximate posterior $q(\theta \mid x)$.

For any (**unlabled**) dataset x^* and any parameter generating distribution $p_C(\theta)$, we define:

$$\mathsf{SCLoss(q)} = \mathsf{Var}_{\theta \sim p_C(\theta)} \left[\log p(x^* \mid \theta) + \log p(\theta) - \log q(\theta \mid x^*)\right]$$

The SC-Loss alone doesn't work well most of the time so we combine it with the standard NPE loss:

$$\mathsf{SemiSupervisedLoss(q)} = \mathsf{NPELoss(q)} + \lambda \cdot \mathsf{SCLoss(q)}.$$

Bayesian Self-Consistency losses a strictly proper

Let C be a score that is globally minimized if and only if its functional argument is constant across the support of the posterior $p(\theta \mid x)$ almost everywhere. Then, C applied to the Bayesian self-consistency ratio with known likelihood

$$C\left(\frac{p(x \mid \theta)\, p(\theta)}{q(\theta \mid x)}\right)$$

is a strictly proper loss: It is globally minimized if and only if $q(\theta \mid x) = p(\theta \mid x)$ almost everywhere.

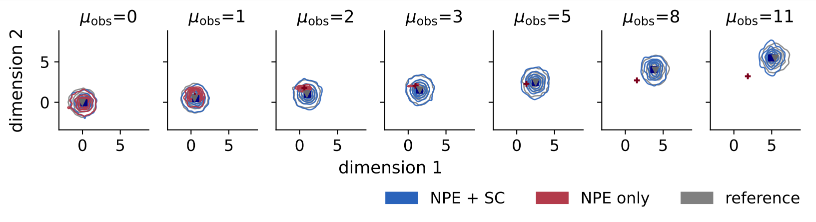
This implies that also the semi-supervised loss is strictly proper.

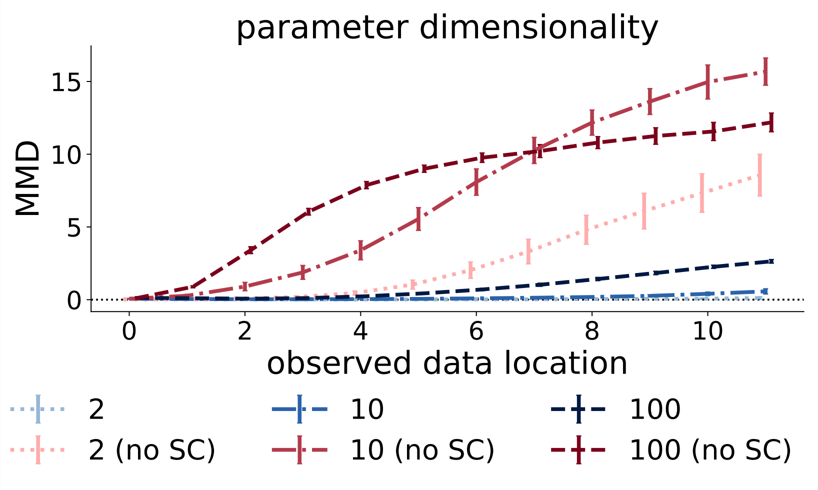
Case Study 1: Multivariate normal model

$$\theta \sim \text{Normal}(\mu_{\text{prior}}, I_D), \quad x \sim \text{Normal}(\theta, I_D)$$

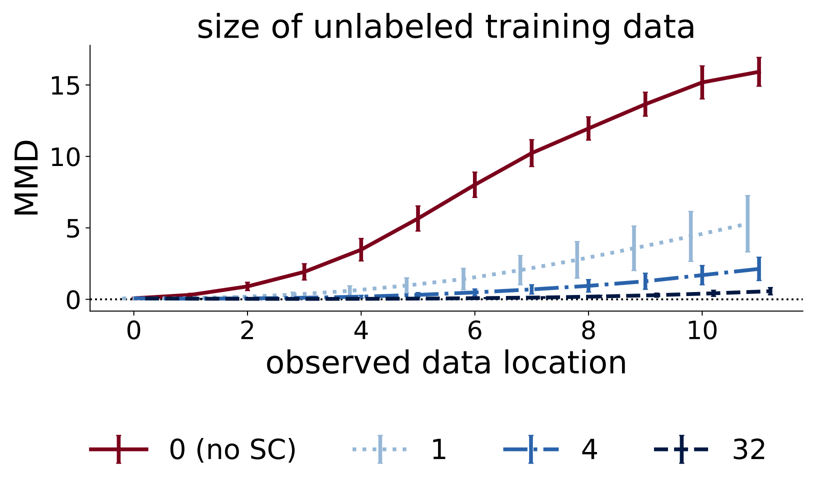
- For the NPE loss, we simulate from the model with $\mu_{\text{prior}} = 0$
- For the SC loss, we simulate **few unalabled datasets** from the model with $\mu_{\text{prior}} = 2$

Illustrative results:




parameter dimensionality
MMD
15
10
5
0
0
2
4
6
8
10
observed data location
2
10
100
2 (no SC)
10 (no SC)
100 (no SC)

Case Study 1: More Results



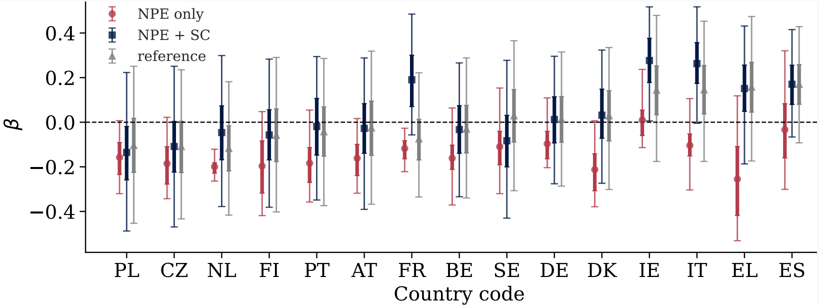
Case Study 2: Time Series of Air Traffic data

Predicting the change in air traffic for different European countries

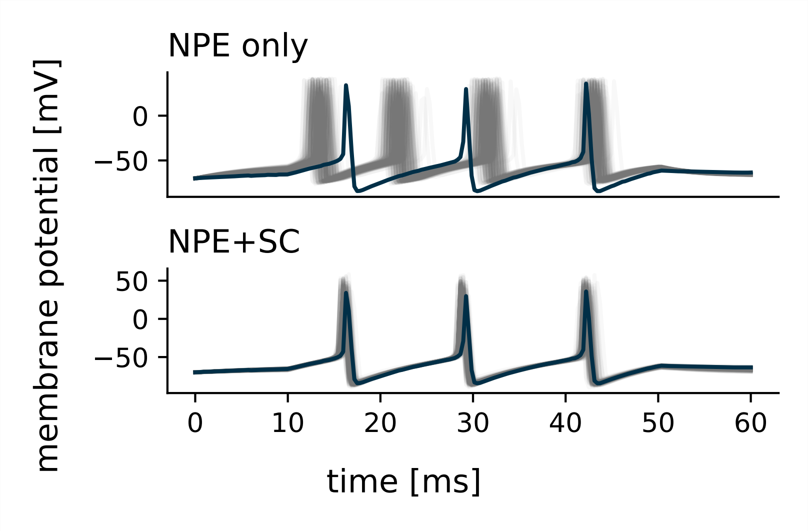
$$y_{j,t+1} \sim \mathrm{Normal}(\alpha_j + \beta_j y_{j,t} + \ldots, \sigma_j)$$

- $y_{j,t}$ number of passengers for country j at year t
- α_j intercept parameter
- β_j auto-correlation parameter
- σ_j residual standard deviation

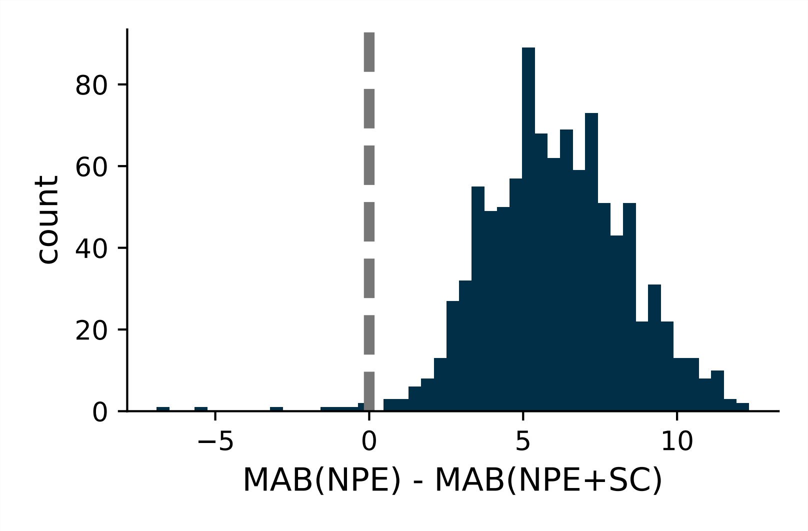
Case Study 2: Results



Case Study 3: Hodgkin-Huxley model of neuron activation



Case Study 3: More results



Conclusion

- The SC loss can strongly improve robustness to model misspecification if the SC data x^* are outside of the typical set of the simulator $p(x, \theta)$ used in the NPE loss
- The SC loss is strictly proper so it has the same target (the true posterior) as the NPE loss
- Challenge 1: The SC loss requires a known or estimated likelihood density: stronger robustness if the likelihood density is known
- Challenge 2: We need neural approximators that have fast density evaluation (currently excludes flow matching or diffusion models)