We thank all reviewers for their insightful comments. First, we will address some major themes from the reviews:

Non-probabilistic Methods & Constrained Optimization Our paper is motivated by the ability to impose constraints probabilistically, i.e. the challenge is to incorporate constraints in a model that infers output *distributions*. Bayesian techniques deviate significantly from classical optimization. As such, we did not include non-probabilistic methods like [Stewart and Ermon, 2017] as baselines, or use notation conventional to constraint optimization.

Hard vs. Soft Constraints Figure 3 shows minor violations of constraints as the priors used are soft and assign small (but > 0) probability to infractions. The idiosyncratic nature of SVGD inference (used in Figure 3b) in learning "repulsive/diverse functions" also makes it more likely to violate a constraint; for example, Figure S1 below shows that for the same example, having only 10 SVGD particles eliminate the few violating functions.

More generally, in probabilistic systems, while hard constraints are theoretically possible by assigning 0 probability to violations, (i) numerical instability issues could arise, and (ii) we tend to obey Cromwell's rule in Bayesian inference, where the support of the prior is usually the entire output space and unlikely functions are naturally weeded out. **We stress that workarounds do exist**: (i) we can specify extremely small (≈ 0) probability to the order of numerical insignificance, so long as the prior remains differentiable, (ii) guarantees *on top of* soft constraints can be enforced, e.g. rejection sampling over OC-BNNs (which will be tractable), (iii) in the amortized setting, we can set a threshold for ϵ directly. We note that soft constraints are often useful too, e.g. for learning (alongside the training data) where the function might be *outside of* the constrained region.

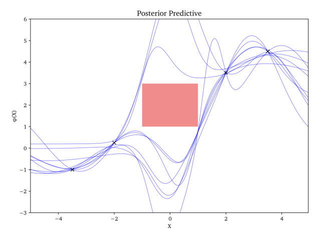

Figure S1: Same example as Figure 3b, except using 10 SVGD particles (instead of 50).

Novelty **(1)** While regularization techniques are common, it is not immediately clear that data-based regularization leads to "well-behaved" and useful priors, (e.g. smooth functions with suitable OOD variance), especially for the amortized variant, or for non-Gaussian likelihoods (e.g. constraints over output ranges). Tractability of sampling with input dimensionality is also not obvious (and not demonstrated by [18]), for example, we found that sampling at the border of constraints proved reasonably well at guiding the model towards good functions. **(2)** We acknowledge points about similarity to [18] made by R4. A more accurate comparison would be that our framework is more general and more versatile at incorporating a diverse range of constraint formulations, without the need to make various Gaussian approximations or sacrifice tractability. **(3)** We want to highlight the strength and novelty of our suite of experiments, which shows that OC-BNNs are useful and work well on a diverse set of real-life problems and constraints.

Additional Comments

[All] ***Technicality:*** The stochastic process setup in Section 4 is to ensure a formal and principled definition keeping with Bayesian inference, not to deceive the reader into unnecessary complexity. We acknowledge that a more intuitive explanation and/or an explicit algorithm box might suffice and technical details be left to supplementary material.

[R1][R3] ***Def. 4.2:*** Indeed, we omitted marginalizing y. Equation (1) in Definition 4.2 should read:

$$p(Y \circ \mathcal{C}_y(\mathbf{x})|\mathbf{x}) = \int_{\mathcal{Y}} \mathbb{I}[y \circ \mathcal{C}_y(\mathbf{x})] \underbrace{\int_{\mathcal{W}} p(Y = y|\mathbf{x}, \mathbf{w})\, p(\mathbf{w})\, \mathrm{d}\mathbf{w}\ \mathrm{d}y}_{\text{prior predictive}} \quad \leq \epsilon$$

Also, as R1 pointed out, $\circ$ should be swapped here: for a positive constraint, $\circ = \notin$ (and vice versa).

[R3] ***Section 3:*** We optimize w.r.t. the parameters of a Gaussian variational representation, hence "variational".

[R1][R4] ***Fig. 2:*** The imperfect fit is due to an idiosyncratic combination of low model capacity and sampling for this particular example. Note that the plot was shaded at a specific confidence level; it is challenging for a 10-node RBF network to fit a specific rectangle at *identical levels of confidence*. A key takeaway from Figure 2 is that we are not overly confident far away from the green rectangle, especially in the prior predictive.

[R4] ***Section 4.1:*** There is a one-to-one correspondence: each constraint $(\mathcal{C}_\mathbf{x}, \mathcal{C}_y, \circ)$ is modeled with a *single* stochastic process, whereby the points of the input region $\mathcal{C}_\mathbf{x}$ is the index of the process. (The equation on Line 135, on the other hand, represents the product of multiple, independent constraints.) The confusion here may arise from the fact that the points within a single constraint are also "independent" as their correct output has been directly defined by $\mathcal{C}_y$.

Russell Stewart and Stefano Ermon. Label-free Supervision of Neural Networks with Physics and Domain Knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.