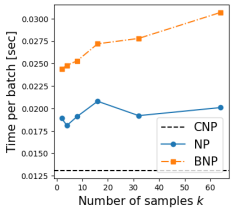
We thank all the reviewers for their constructive comments. Although the reviewers were generally happy with the novelty of our method, there were some concerns on the experimental results for which we address in this rebuttal.

[All reviewers] Typos We appreciate for pointing them out, and will correct them in the revised version.

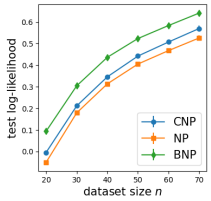
[R1] Improvements over vanila NP We respectfully disagree. Please note that BNP and BANP outperform baselines in most experiments in terms of log-likelihood, especially for mismatch data. Even though the absolute difference may look small in our metric, the differences are significant because they are 1) logs of likelihoods and 2) measured per datapoint. Note that confidence intervals are on the order of 10^{-3} and that our model matches or even outperforms the ensemble of five models. Fig 2. shows the difference in uncertainty quantification of ANP and BANP. The two models show similar behavior for normal RBF data, but BANP produces wider credible intervals for mismatch data (Periodic and t-noise). In other words, BANP tends to be less confident for mismatch data and thus better calibrated. We further demonstrate this tendency in the additional qualitative results given in Fig D. 5. in the supplementary.



(a) k vs trainng time

	context	target
BNP	$1.012_{\pm 0.006}$	$0.523_{\pm 0.004}$
$-p_{\text{base}}$	$0.981_{\pm 0.008}$	$0.487_{\pm 0.007}$
BANP	$1.379_{\pm 0.000}$	$0.849_{\pm 0.001}$
$-p_{\text{base}}$	$1.378_{\pm 0.000}$	$0.836_{\pm 0.003}$

(b) Loss ablation



(c) Context n vs LL

[R1] Why would one consider using BNP despite additional computation? We stress again that BNP/BANP show clear improvements over baseline in terms of log-likelihoods. Another benefit is generality: one can apply the bootstrap idea to any NP model (e.g., convolutional CNP) without having to carefully design variational distributions and tune the hyperparameters to train them properly (e.g., choosing latent dimensions, KL annealing, ...).

[R1] Is log-likelihood a proper performance measure? How they are computed? Log-likelihood is a proper scoring rule (ref [13]) that gives a higher value for a better calibrated model. We computed log-likelihood values following the convention in the NP literature (ref [14]). As you pointed out, the log-likelihood values computed are lower-bounds, but they approach the true values as the number of samples k increases. The log-likelihood values reported in the paper were computed with $k = 50$ samples. We will make this more clear in our revised version.

[R1] Bayesian optimization is less highlighted The bayesian optimization experiment quantifies the quality of the uncertainty estimates of models through the minimum simple regret and cumulative minimum regret metrics. In this experiment, BNP/BANP ourperformed other NP baselines and was even comparable to GP. We will discuss and highlight this more in our revised version.

[R1, R2] Training time We measured the average processing time per batch for CNP, NP, and BNP in (1a). The computation time of BNP is less than twice of CNP and NP because the first pass to compute residuals uses only the context set (X_c, Y_c), which is a subset of the entire batch. Thanks to the parallelization, the computation time for NP and BNP scales sub-linearly with k. We will discuss computing time requirements more in the revised version.

[R2] Comparison to naïve bootstrap Please refer to Table D.5 in the supplementary, where we performed ablation studies including the naïve bootstrap.

[R2] Failure cases We agree that the analysis of failure cases can improve the understanding of our model. Though not exactly a failure, we think that our models for image completion tasks have room for improvement since we are restricting the output range to be in $[-1, 1]$, but we do not consider this during residual resampling.

[R4] Definition of model-data mismatch Thanks for pointing this out. For now, we roughly define model-data mismatch to be the case where the test task distribution differs from the training task distribution. As you mentioned, the difference can be in terms of domains, or even in generative processes within the same domain. We will add more discussion on this matter in the revised version.

[R4] Justification for the objective (14) We empirically confirmed that the objective without p_{base} performs bad (still better than CNP or NP). Partial results for 1D regression are in (1b); we will include full results in the revised version.

[R4] BNP/BANP do not always performs better Yes, but please note that ours perform better than baselines for the most of the cases, especially for mismatch settings. The qualitative results on EMNIST are well reflected in the log-likelihood values, showing significant improvements over baselines.

[R4] Benefit from parallelization? Our implementation is already utilizing parallelization by packing multiple bootstrap contexts into a single tensor and process them in parallel. A naïve iterative implementation scales poorly and takes horribly long to train. Please refer to our source code for implementation details.

[R4] Number of context points vs performance Thanks for the suggestion. In (1c), we measured the target log-likelihood of CNP, NP, and BNP on 1D regression tasks with varying task size n (thus varying number of contexts). BNP consistently performed better with a significant margin. Although we did not put it here due to space constraints, the same was true of models with attention.

[R5] Proofread Sorry for the inconvenience, we will do our best to revise our paper.