An Empirical Study of Prior-Data Conflicts in Bayesian Neural Networks

Alexander Marquardt, Julian Rodemann, Thomas Augustin

Department of Statistics, LMU Munich, Germany

amarquard@campus.lmu.de, julian.rodemann@stat.uni-muenchen.de , thomas.augustin@stat.uni-muenchen.de

Bayesian Neural Networks (BNNs)

Fundamentals

BNNs employ a probabilisitc approach to Neural Networks (NNs) by leveraging Bayes´ Theorem and specifying a distribution over the weights of a NN.

$$P(\theta|X, y) = \frac{P(y|X, \theta) \times P(\theta)}{P(y|X)}$$

For parameters $\theta = (w, b)$, where w represents the weights of the network and b denotes the biases, and the data X and y, where X represents the covariates and y represents the target, respectively, [7].

Algorithm 1 Inference procedure in a BNN

```
Define $p(\theta|X, y) = \frac{p(y|X,\theta)p(/theta)}{p(y|X)}$
for $i = 0$ to $N$ do
    Draw $\theta_i \sim p(\theta|X, y)$
    $y_i = \Phi_{\theta_i}(x)$
end for
return $Y = \{y_i|i \in \{0, \cdots, N\}\}, \Theta = \{\theta_i|i \in \{0, \dots, N\})\}$
```

By adopting a probabilistic approach, each feed-forward step is coupled with draws of the weights from the posterior $w \sim P(\theta|X, y)$ which allows modelling uncertainty, expressed by the trust in the posterior parameter $Var(\theta|X, y)$. Higher posterior-variances result in an increased variation in the drawn weights, leading to a greater variation in the network´s output. This type of uncertainty is commonly referred to as epistemic uncertainty, [6].

Variational Inference

An common approach to handle the posterior distribution is the usage of approximation-based procedures like Variational Inference (VI).
Here, the posterior is approximated by a variational posterior $q_\psi(\theta)$, parameterized by ψ. For that, KL-Divergence is used:

$$KL(q_\psi(\theta)||p(\theta|X, y)) = KL(q_\psi(\theta)||p(\theta)) - \mathbb{E}_{q_\psi(\theta)}(log(p(y|X, \theta) + log(p(y|X))))$$

Since the KL-Divergence is always ≥ 0 and the logarithmic evidence $log(p(y|X))$ is independent of θ, the first part of the equation, $J(\theta) = KL(q_\psi(\theta)||p(\theta)) - \mathbb{E}_{q_\psi(\theta)}(log(p(y|X, \theta)$ is used as the optimization objective. $J(\theta)$ is commonly referred to as Evidence-Lower-Bound (ELBO), [7], [4]. In case of mini-batches, where parameters are updated after each batch gradient calculation, the KL-Divergence is scaled, [5]

$$J(\theta)^{(i)} = \frac{1}{M} KL(q_\psi(\theta)||p(\theta)) - \mathbb{E}_{q_\psi(\theta)}(log(p(y|X, \theta)$$

for $i \in \{1, \dots, M\}$ where M is the number of equally-sized batches, [5].

Outlook: How Credal Sets Could Help

Motivation: Credal Sets have proven very fruitful for uncertainty quantification in situations involving prior-data conflicts [2, 10], when classical (precise) models fail to capture the uncertainty. Our analysis [9] suggests that this might be transferrable to BNNs, as their uncertainty estimates were insensitive to prior-data conflict.

Imprecise Bayesian Neural Networks [3]: Very recently, [3] proposed to equip BNNs with a finite set of priors and update each of them. They then compute α-level Imprecise Highest Density Region (IHDR) $IR_\alpha := \cup_{k=1}^{K} R\left(\hat{p}_k^\alpha\right)$ as union of classical HDRs $R\left(\hat{p}_k^\alpha\right) := \left\{y : \hat{p}_k(y) \geq \hat{p}_k^\alpha\right\}$, where $\alpha \in (0, 1)$ and $\hat{p}_k^\alpha$ is the largest constant such that $\hat{P}_k\left[Y \in R\left(\hat{p}_k^\alpha\right)\right] \geq 1 - \alpha$ with $k \in \{1, \dots, K\}$ the number of priors.

Adaption to our setup: We adopted a minimalistic approach from [3] where we defined two priors. These priors were updated using the same likelihood, resulting in two distrinct posteriors. Subsequently, we calculated the α-level HDR for each model individually (Figure 1: left) and then combined them to obtain an α-level IHDR (Figure 1: right).

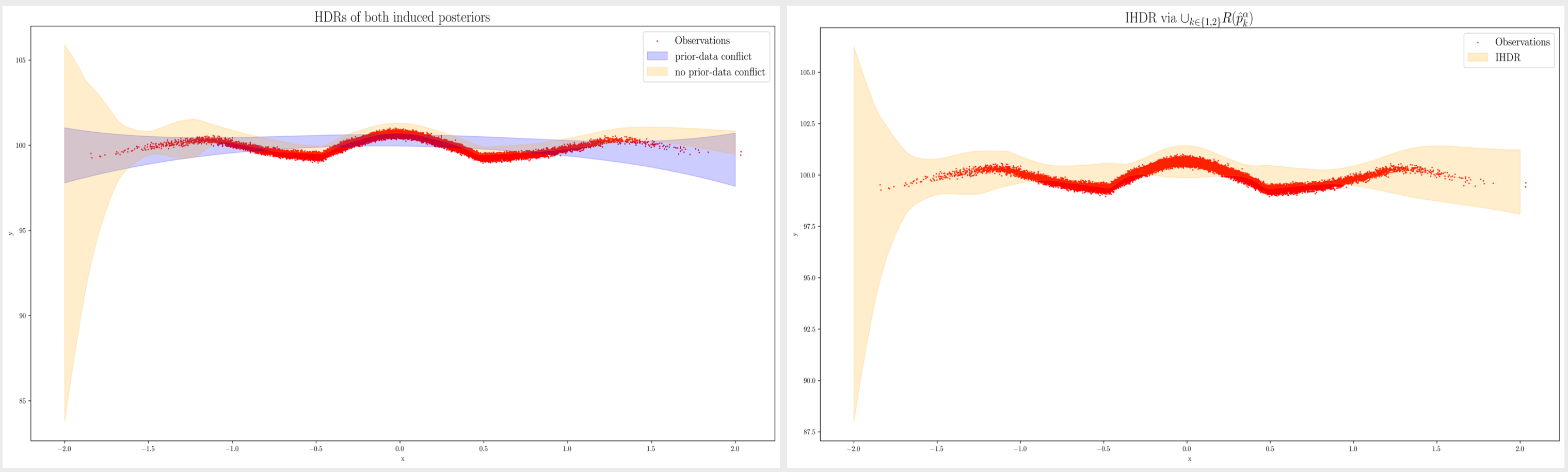

Figure 1: Left: Individual HDRs of both models, with and without prior-data conflict. Right: Combined HDR of both models, where we used the union of both HDRs to model an IHDR.

Figure 1 illustrates that in regions where both models exhibit agreement, uncertainty predictions are tighter. However, in regions where the two models do not align, the IHDR reports a bigger width, indicating a higher level of uncertainty, indeed.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] Thomas Augustin, Gero Walter, and Frank Coolen. Statistical inference. In T. Augustin, F. Coolen, G. de Cooman, and M. Troffaes, editors, *Introduction to Imprecise Probabilities*, pages 135–189. Wiley, 2014.

[3] Michele Caprio, Souradeep Dutta, Kuk Jin Jang, Vivian Lin, Radoslav Ivanov, Oleg Sokolsky, and Insup Lee. Imprecise Bayesian neural networks. *arXiv preprint arXiv:2302.09656*, 2023.

[4] Ethan Goan and Clinton Fookes. Bayesian neural networks: An introduction and survey. In Kerrie L. Mengersen, Pierre Pudlo, and Christian P. Robert, editors, *Case Studies in Applied Bayesian Data Science*, pages 45–87. Springer International Publishing, 2020.

[5] Alex Graves. Practical variational inference for neural networks. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011.

[6] Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, 110(3):457–506, mar 2021.

[7] Laurent Valentin Jospin, Hamid Laga, Farid Boussaid, Wray Buntine, and Mohammed Bennamoun. Hands-on Bayesian neural networks—a tutorial for deep learning users. *IEEE Computational Intelligence Magazine*, 17(2):29–48, may 2022.

[8] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

[9] Alexander Marquard. Eine empirische Analyse von Prior-Daten Konflikten in Bayesianischen neuronalen Netzen. Bachelor's Thesis, Ludwig-Maximilians-Universität München, 2023.

[10] Gero Walter and Thomas Augustin. Imprecision and prior-data conflict in generalized bayesian inference. *Journal of Statistical Theory and Practice*, 3(1):255–271, 2009.

Experimental Setup

Motivation: In standard BNNs, it is common to assume a prior distribution of $N(0, \alpha I)$ for the weights [7]. This choice is due to the fact that there is a lack of reasonable a priori assumptions about the true values of the weights, thereby resulting in potential prior-data conflicts. To evaluate the extent to which these conflicts influence the results in BNNs, we conducted our analysis.
In our study [9] we focused on analyzing the epistemic uncertainty, specifically assuming that σ^2 is known and constant.

Setup: To analyze the impact of prior-data conflicts we constructed a synthetic dataset with the following functional relationship

$$y_i = \frac{cos(x_i)}{|x_i| + 1} + 100 + \epsilon_i.$$

Here, $\epsilon_i \sim N(0, 0.25)$. This dataset was used to train a traditional neural network, comprising three input neurons, two hidden layers with 100 neurons each, and an output layer with a single neuron. ReLU Activation was applied between the hidden layers. The weights of this traditional neural network were then employed to generate a new dataset:

$$\tilde{y_i} = f(x_i) + \tilde{\epsilon_i}$$

with $\epsilon_i \sim N(0, 0.01)$. This dataset possesses two important characteristics: firstly, the true functional relationship is known (i.e., a neural network with four layers and ReLU Activations), which minimizes model uncertainty [6]; and secondly, the true values (i.e., the weights) are known a priori.

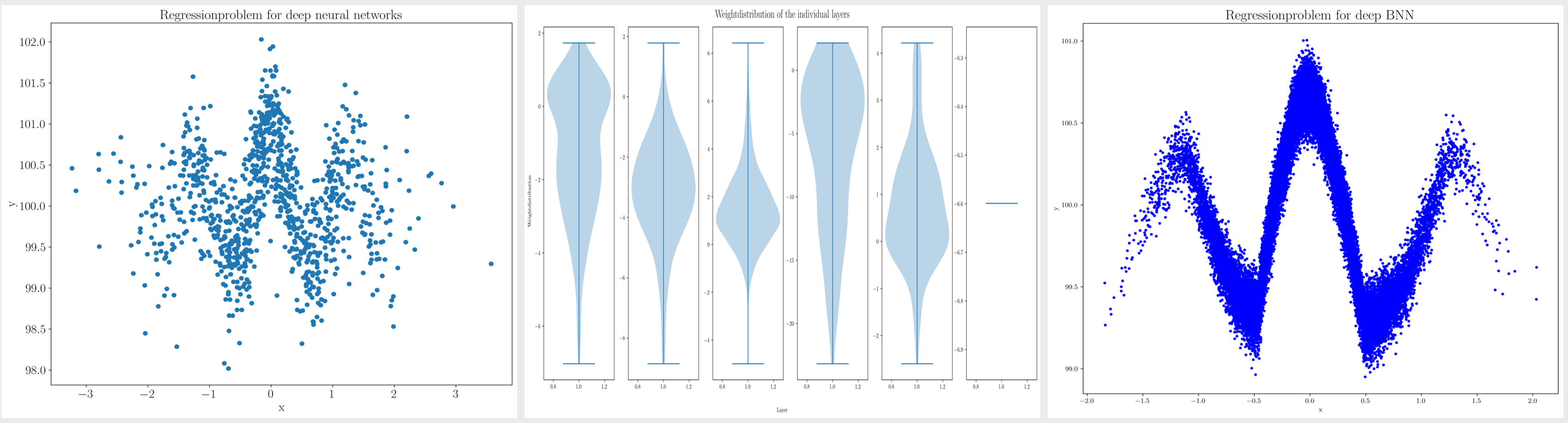

Figure 2: Illustration of the three stages involved in generating the synthetic dataset. Left: The initial dataset to train a traditional neural network. Middle: Visualization of the weights in the traditional neural network. Each layer exhibits values that are significantly different from 0. The figures are paired, representing the weights of neurons and biases from one layer to the next. Right: The generated dataset, utilizing the traditional neural network as ground truth.

Structural Assumptions:

- Model posterior distributions for $\theta = (w, b)$: $q(\theta_i^{(l)}|\mu_i^l, \sigma_i^{(l)})$ where $\theta_i^{(l)}$ is the i-th weight in the l-th layer
- Assume a factorizable multivariate gaussian $q(\theta^{(l)}|\mu^{(l)}, \sigma^{(l)^2}) = \prod_{i=1}^{n_l} q(\theta_i|\mu_i^{(l)}, \sigma_i^{(l)^2})$ where n_l denotes the number of weights in the l-th layer
- Assume a gaussian likelihood: $\tilde{y}_i|X_i \sim N(f(X_i|\theta), 0.01)$
- Assume a $N(0, I)$ prior for θ implying a prior-data conflict and $N(w_{trad}^{(l)}, I)$ as reference prior with the exact weights known from the NN

Training: We used a batch size of 1.000 samples, with a total of 30.000 observations. Given that we trained approcimately 21.000 parameters, having 30.000 observations seemed reasonable to ensure that the prior choice still influences the inference process. Furthermore, we scaled the KL-Divergence between the variational posterior and the prior by $\frac{1}{30}$, as advocated by [5].
Scaling the KL-Divergence has the following implications:

- Using a high scaling-factor (e.g. $\frac{1}{\#\text{batches}}$, as in [5]) gives a greater importance towards the complexity cost
- Using a low scaling factor (e.g. $\frac{1}{\#\text{total observations}}$) prioritizes the likelihood

Lastly, we employed the Reparametrization Trick, [8], which is the standard procedure for backpropagation in Tensorflow, [1].

Results

Inference: The following figures depict the MC samples from the BNNs. We utilized two different priors – one centered around the true values and another centered around 0.

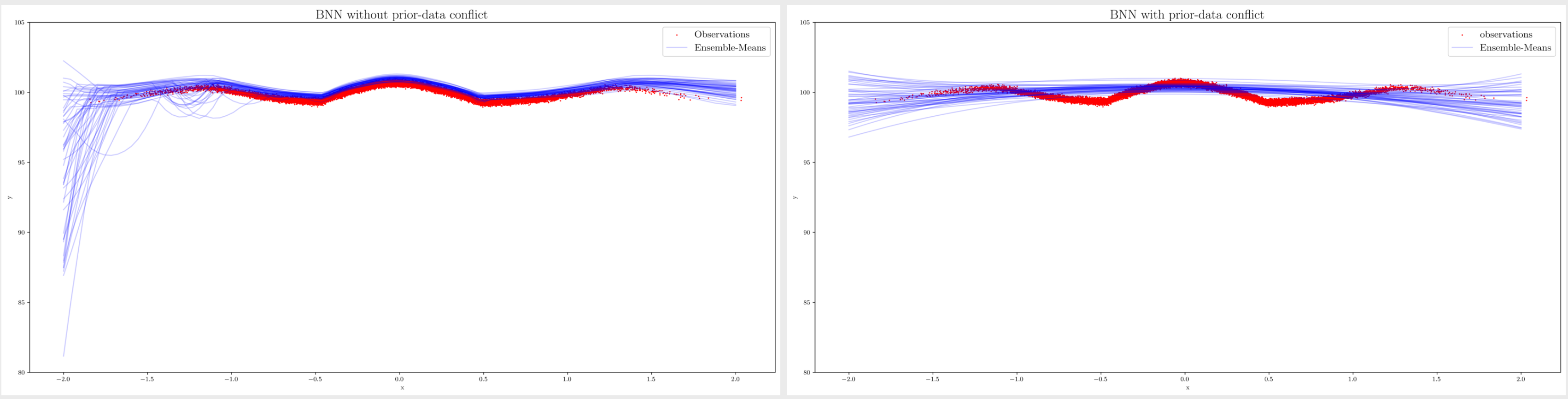

Figure 3: Left: The predictions of the reference BNN which employs priors centered around the true values known from the NN. Right: The predictions of the BNN with a prior centered around 0.

Looking at the posterior variance yields similar results, independently from the prior choice.

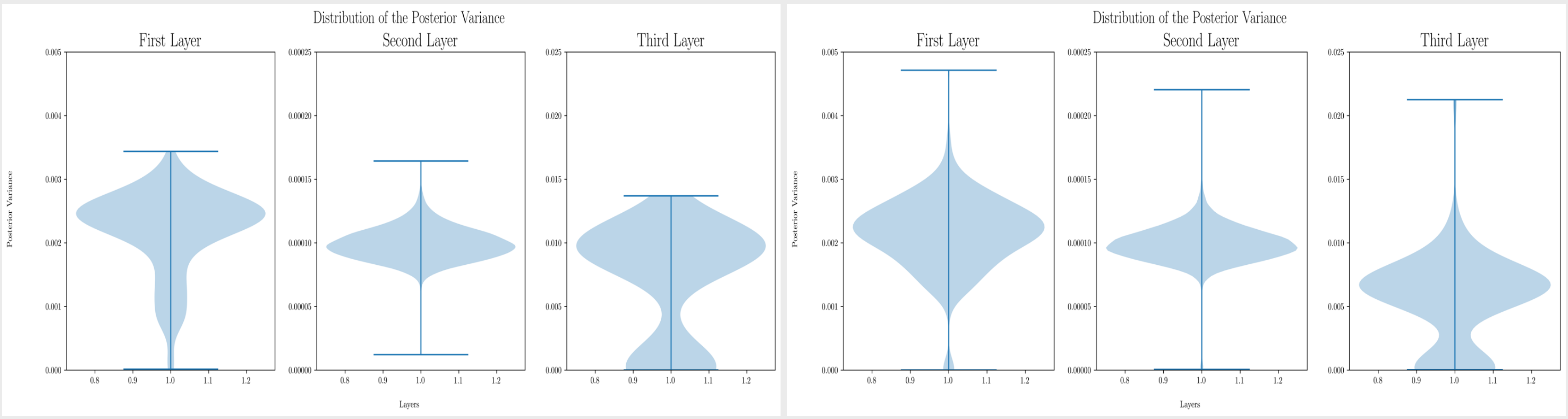

Figure 4: Left: Distribution of the posterior variance $Var(\theta|X, y)$ for the model without prior-data conflict for each layer. Right: Distribution of the posterior variance for the model with prior-data conflict.

Summary: The results depicted in Figure 3 highlight that BNNs with prior-data conflict struggle to capture the underlying structure, while BNNs with the correct prior demonstrate a good fit to the data, despite the limited number of observations. This finding contrasts with the estimated posterior variances, as shown in Figure 4, where both models exhibit similar uncertainty estimations. **Apparently, the BNNs are not able to adequately represent the uncertainty induced by the prior-data conflict in our simulation.**