5th Symposium on Advances in Approximate Bayesian Inference, 2023 1–12

Indirect Functional Bayesian Neural Networks

Anonymous Authors
Anonymous Institution

Abstract

Bayesian neural networks (BNNs) have made significant contributions in improving the robustness and uncertainty quantification of deep neural networks but suffer from problematic priors for network weights. We propose a new kind of indirect functional BNNs (IFBNN) by building a Wasserstein bridge, which consists of a 2-Wasserstein distance between the approximate posterior and a bridging distribution of network weights, and a 1-Wasserstein distance between the bridging distribution over functions induced by weight distributions and a functional GP prior. It can avoid the potential risks of invalid or infinite functional KL divergence commonly used by most existing functional BNNs. We demonstrate the improved extrapolation and predictive performances of the proposed IFBNN empirically on both synthetic and real-world datasets.

1. Introduction

Bayesian neural networks (BNNs) (Blundell et al., 2015; Gal and Ghahramani, 2016; Gal, 2016) are able to merge the strong predictive capability of deep neural networks (DNNs) with the uncertainty modeling ability of Bayesian inference. Despite its merits, BNNs still suffer from some weight prior issues: i) The widely used Gaussian priors for network parameters are not always applicable due to their possible pathological features, such as prior samples tend to be horizontally linear for deep nets (Duvenaud et al., 2014; Matthews et al., 2018; Tran et al., 2020); ii) The effects of the given priors on posterior inference for weights and further on the resulting distributions over model outputs in function space are unclear and hard to control owing to the complex architecture and nonlinear nature of BNNs (Ma and Hernández-Lobato, 2021; Fortuin et al., 2021; Wild et al., 2022).

To resolve these issues, there have been increasing attention to the analysis of BNNs in function space instead of weight space (Ma et al., 2019; Rudner et al., 2020, 2022). In such formulation, the distributions over function mappings from BNNs are regarded as probability measures in function space induced by the distribution over network weights, that is, stochastic processes. Also, the prior distributions can be constructed in function space more informatively, such as the popular functional priors *Gaussian Processes* (GPs) (see Appendix A). A major advantage of GP priors is that they are able to easily encode prior knowledge about properties of unknown functions (e.g. periodicity and smoothness) through different kernel functions. In order to approximate posterior distributions over functions, existing functional BNNs directly build and minimize the Kullback–Leibler (KL) divergence between BNN posterior and GP prior (Sun et al., 2019). However, such infinite-dimensional KL divergence may be invalid due to the assumption of the existence of Radon-Nikodym derivatives between the prior and the variational approximate posterior (Matthews et al., 2016; Burt et al., 2020), such as the KL divergence between two BNNs with different structures is infinite (Ma and Hernández-Lobato, 2021).

© A. Authors.

In this work, we propose a new kind of indirect functional BNNs based on a Wasserstein bridge. Specifically, the Wasserstein bridge consists of a 2-Wasserstein distance between the approximate posterior and a bridging distribution over network weights, and a 1-Wasserstein distance between the bridging distribution over network functions (induced by the distribution over network weights) and the prior over functions, like GP. Instead of directly regularizing the variational BNN posterior using the KL divergence with GP prior by most functional BNNs, our indirect functional BNNs can avoid the limitations of infinite-dimensional KL divergence through the proposed Wasserstein bridge.

2. Preliminaries

Given a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n = \{\mathcal{X}, \mathcal{Y}\}$, a (deep) neural network defines and learns a function mapping $\mathbf{y} = \mathbf{f}(\mathbf{x}; \mathbf{w})$ between input $\mathbf{x} \in \mathbb{R}^d$ and output $\mathbf{y} \in \mathbb{R}^c$, where $\mathbf{w} \in \mathbb{R}^p$ denotes the network weights. A Bayesian neural network further assigns prior $p_0(\mathbf{w})$ for the network weights and learns its posterior $p(\mathbf{w}|\mathcal{D})$ together with defined likelihood $p(\mathcal{D}|\mathbf{w})$.

Weight-space view for BNNs Since the marginal integration required in solving the posterior is intractable for any practical dimensions, the main goal of the variational inference for BNNs is to fit a tractable approximate posterior $q(\mathbf{w}; \boldsymbol{\theta_q})$ by maximizing the evidence lower bound (ELBO), where $\boldsymbol{\theta_q}$ denotes the distribution parameters. Bayes By Backprop (BBB) proposed by Blundell et al. (2015) is one of the most representative mean-field variational inference algorithms in weight space

$$\mathcal{L}_{q(\mathbf{w};\boldsymbol{\theta_q})} := \mathbb{E}_{q(\mathbf{w};\boldsymbol{\theta_q})}\left[\log p(\mathcal{D} \mid \mathbf{w}; \boldsymbol{\theta_q})\right] - \mathbb{KL}[q(\mathbf{w}; \boldsymbol{\theta_q}) \| p_0(\mathbf{w})], \tag{1}$$

where the common treatment paradigm for $p_0(\mathbf{w})$ is i.i.d. Gaussians.

Function-space view for BNNs Suppose $p_0(\mathbf{f})$ is a functional prior for BNNs defined on a probability space $(\Omega, \mathcal{F}, P)$ (separable metric and complete Polish space). The objective of functional BNNs is to maximize the functional ELBO to infer the approximate posterior processes $q(\mathbf{f}; \boldsymbol{\theta_q})$ over functions induced by distributions over network weights

$$\mathcal{L}_{q(\mathbf{f};\boldsymbol{\theta_q})} := \mathbb{E}_{q(\mathbf{f};\boldsymbol{\theta_q})}\left[\log p(\mathcal{D} \mid \mathbf{f}; \boldsymbol{\theta_q})\right] - \mathbb{KL}[q(\mathbf{f}; \boldsymbol{\theta_q}) \| p_0(\mathbf{f})], \tag{2}$$

where $\mathbb{KL}[q(\mathbf{f}; \boldsymbol{\theta_q}) \| p_0(\mathbf{f})]$ is an infinite-dimensional KL divergence and its estimation is difficult but achievable because there is a link with its marginal KL divergence on measurement set as demonstrated by Theorem 1 in (Sun et al., 2019) that is $\mathbb{KL}[P \| Q] = \sup_{n \in \mathcal{N}, X \in \mathcal{X}^n} \mathbb{KL}[P_X \| Q_X]$ where P and Q are two stochastic processes, but this KL divergence is sometimes problematic as discussed in the Introduction.

3. Indirect functional Bayesian neural networks

In this section, we propose Wasserstein distance-based functional Bayesian neural networks. We firstly transform the GP functional prior to a weight space prior by matching it with a bridging distribution that is used to replace the i.i.d Gaussian prior (Section 3.1), and then an integrated strategy is proposed to jointly optimize the bridging distribution and the variational posterior (Section 3.2).

3.1. GP-induced functional BNNs

For function mapping $\mathbf{f}: \mathbb{R}^d \to \mathbb{R}^c$ of BNNs, let $p(\mathbf{f}_{nn}; \boldsymbol{\theta_p})$ be a BNN-based bridging distribution over functions induced by the distribution over weights $p(\mathbf{w}; \boldsymbol{\theta_p})$, which also depends on certain network architecture (e.g., depth, width, activation). The GP prior is denoted by $p_0(\mathbf{f}_{gp}) \sim \mathcal{GP}(\mathbf{m}, \mathbf{K})$. Since it is challenging to directly use GP prior to regularize the BNN posterior (Wild et al., 2022), we firstly transform the GP prior to a BNN-based bridging distribution through optimizing the 1-Wasserstein distance (See Appendix A for more details) between the bridging distribution $p(\mathbf{f}_{nn}; \boldsymbol{\theta_p})$ and GP prior $p_0(\mathbf{f}_{gp})$ (Tran et al., 2022). Considering the infinite-dimensional nature of stochastic processes, we will use their corresponding (random) marginal distributions, specifically, solving the 1-Wasserstein distance (in its dual form) between $p(\mathbf{f}_{nn}; \boldsymbol{\theta_p})$ and $p_0(\mathbf{f}_{gp})$ on finite randomly sampled measurement points $\mathbf{X}_{\mathcal{M}} \overset{\text{det}}{=} [\mathbf{x}_1, \ldots, \mathbf{x}_M]^{\mathrm{T}}$. The specific form is as follows

$$W_1\left(p(\mathbf{f}_{nn}; \boldsymbol{\theta_p}), p_0(\mathbf{f}_{gp})\right) = \mathbb{E}_{\mathbf{X}_{\mathcal{M}}}\left[\mathbb{E}_p \phi(\mathbf{f}_{nn}^{\mathcal{M}}) - \mathbb{E}_{p_0} \phi(\mathbf{f}_{gp}^{\mathcal{M}})\right] \tag{3}$$

where $\mathbf{f}_{nn}^{\mathcal{M}}$ and $\mathbf{f}_{gp}^{\mathcal{M}}$ are corresponding function values evaluated at $\mathbf{X}_{\mathcal{M}}$, respectively. It can be seen that this approximated computation procedure is based entirely on sampling, so it can still be performed smoothly even without the closed form of $p(\mathbf{f}_{nn}; \boldsymbol{\theta_p})$. We can obtain a GP-induced optimal bridging distribution over weights $p(\mathbf{w}; \boldsymbol{\theta_p}^*)$ where $\boldsymbol{\theta_p}^* = \arg\min_{\theta_p} W_1\left(p(\mathbf{f}_{nn}; \boldsymbol{\theta_p}), p_0(\mathbf{f}_{gp})\right)$.

With the $p(\mathbf{w}; \boldsymbol{\theta_p}^*)$ in hand, we can conduct a more reasonable posterior variational inference based on the 2-Wasserstein distance between the approximate posterior $q(\mathbf{w}; \boldsymbol{\theta_q})$ and $p(\mathbf{w}; \boldsymbol{\theta_p}^*)$ with the ELBO denoted by

$$\mathcal{L}_{q(\mathbf{w}; \boldsymbol{\theta_q})} := \mathbb{E}_{q(\mathbf{w}; \boldsymbol{\theta_q})}\left[\log p(\mathcal{D} \mid \mathbf{w}; \boldsymbol{\theta_q})\right] - \lambda W_2(q(\mathbf{w}; \boldsymbol{\theta_q}), p(\mathbf{w}; \boldsymbol{\theta_p}^*)), \tag{4}$$

where λ is a hyperparameter and $W_2(q(\mathbf{w}; \boldsymbol{\theta_q}), p(\mathbf{w}; \boldsymbol{\theta_p}^*))$ has an analytical solution for two Gaussians as

$$W_2(q(\mathbf{w}; \boldsymbol{\theta_q}), p(\mathbf{w}; \boldsymbol{\theta_p}^*)) = \|m_q - m_p\|_2^2 + \operatorname{trace}\left(C_q + C_p - 2\left(C_q^{1/2} C_p C_q^{1/2}\right)^{1/2}\right), \tag{5}$$

where $\boldsymbol{\theta_q} := \{m_q, C_q\}, \boldsymbol{\theta_p}^* := \{m_p^*, C_p^*\}$ are respective mean and covariance matrices. We call this improved variational inference approach based on the matching between bridging distribution and GP prior as the GP-induced FBNNs (GPi-FBNN).

3.2. Indirect functional BNNs based on Wasserstein bridge

The two-step optimization of Equation (3) and Equation (4) in above GPi-FBNN may lead to suboptimal of $p(\mathbf{w}; \boldsymbol{\theta_p}^*)$ and high variance due to the isotropy of 1-Wasserstein distance. Hence, we further consider treating $\boldsymbol{\theta_p} = \{\mu, \sigma\}$ based on the Gaussian reparameterization trick in bridging distribution as parameters need to be optimized together with variational posterior parameters.

Based on the bridging distribution, we propose a new kind of indirect functional BNNs (IFBNN) by building a Wasserstein bridge with the generalized loss as

$$\mathcal{L} := -\mathbb{E}\left[\log p(\mathcal{D} \mid \mathbf{w}; \boldsymbol{\theta_p}, \boldsymbol{\theta_q})\right] + \underbrace{\lambda_1 W_2(q(\mathbf{w}; \boldsymbol{\theta_q}), p(\mathbf{w}; \boldsymbol{\theta_p})) + \lambda_2 W_1\left(p(\mathbf{f}_{nn}; \boldsymbol{\theta_p}), p_0(\mathbf{f}_{gp})\right)}_{\textbf{Wasserstein bridge}}, \tag{6}$$

Algorithm 1: Indirect Functional Bayesian Neural Networks (IFBNN)

Input: Dataset $\mathcal{D}$, GP prior $p_0(\mathbf{f}_{gp})$;
Initialization: $\boldsymbol{\theta_p} \sim \mathcal{N}(0,1)$, $\boldsymbol{\theta_q} \sim \mathcal{N}(0,1)$;
while $\boldsymbol{\theta_p}$, $\boldsymbol{\theta_q}$ *not converged* **do**
  draw measurement set $\mathbf{X}_{\mathcal{M}}$ randomly;
  draw GP functions $\mathbf{f}_{gp}^{\mathcal{M}} \sim p_0(\mathbf{f}_{gp})$ at $\mathbf{X}_{\mathcal{M}}$;
  draw bridging distribution functions $\mathbf{f}_{nn}^{\mathcal{M}} \sim p(\mathbf{f}_{nn}; \boldsymbol{\theta_p})$ at $\mathbf{X}_{\mathcal{M}}$;
  calculate $\log p(\mathcal{D} \mid \mathbf{w}; \boldsymbol{\theta_p}, \boldsymbol{\theta_q})$, $W_2(q(\mathbf{w}; \boldsymbol{\theta_q}), p(\mathbf{w}; \boldsymbol{\theta_p}))$, $W_1(p(\mathbf{f}_{nn}; \boldsymbol{\theta_p}), p_0(\mathbf{f}_{gp}))$ using Equation (3) and Equation (5);
  $\boldsymbol{\theta_p}$, $\boldsymbol{\theta_q} \leftarrow$ Optimizer($\boldsymbol{\theta_p}$, $\boldsymbol{\theta_q}$, $\mathcal{L}$);
end

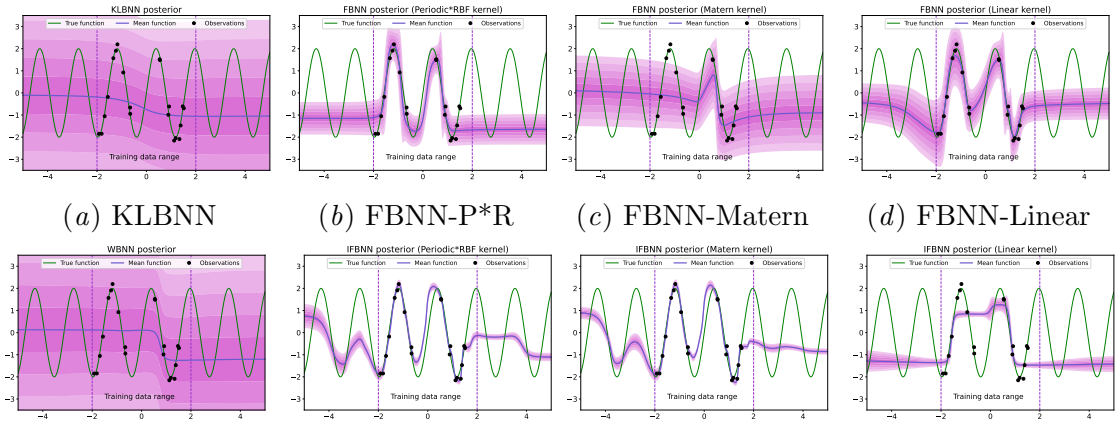
where $\boldsymbol{\theta_q}$ and $\boldsymbol{\theta_p}$ are respective stochastic parameters of approximate variational posterior and bridging distribution which would be optimized jointly; λ_1 and λ_2 are two hyperparameters. Different from most functional BNNs which perform variational inference directly on approximate posterior over functions, the proposed IFBNN treats bridging distribution over network functions induced by distributions over weights as an intermediate variable to link the BNN posterior and GP prior. It could obtain approximate posterior of weights indirectly. Specifically, it performs distribution matching between $p(\mathbf{f}_{nn}; \boldsymbol{\theta_p})$ induced by $p(\mathbf{w}; \boldsymbol{\theta_p})$ and a GP prior through 1-Wasserstein distance and minimizing 2-Wasserstein distance between variational posterior $q(\mathbf{w}; \boldsymbol{\theta_q})$ and $p(\mathbf{w}; \boldsymbol{\theta_p})$ simultaneously. The main advantages of IFBNN are that: i) Compared with the common i.i.d Gaussian assumption, it can place a more interpretable and informative prior by a GP; ii) It can obtain approximate posterior of weights through the Wasserstein bridge, which can also circumvent the limitation of functional KL divergence in most posterior inference processes of existing functional BNNs. Note that Wild et al. (2022) also proposed a generalized variational inference in function space based on the 2-Wasserstein distance, but they only use a GP with deep neural network-based mean function as the variational posterior rather than a BNN, which eases the computation but limits its ability on uncertainty modeling. The pseudocode of IFBNN is presented in Algorithm 1.

4. Experimental evaluation

Following the literature, we evaluate the extrapolation and predictive ability of our proposed IFBNN on two toy examples and benchmark UCI regression tasks.

4.1. Extrapolation illustrative examples

Learning periodic structure Consider an illustrative periodic function: $y = 2{*}\sin(4x) + \epsilon$ with noise $\epsilon \sim \mathcal{N}(0, 0.01)$. 20 training points are randomly sampled from the interval $[-2, -0.5] \cup [0.5, 2]$. GPs with RBF kernel multiplied by a periodic kernel are able to well characterize the periodic properties of this kind of function. We compare our method with two weight-space variational inference approaches: BBB (Blundell et al., 2015) with KL divergence denoted by KLBNN and a 2-Wasserstein distance alternative version called



(a) KLBNN (b) FBNN-P*R (c) FBNN-Matern (d) FBNN-Linear

(e) WBNN (f) IFBNN-P*R (g) IFBNN-Matern (h) IFBNN-Linear

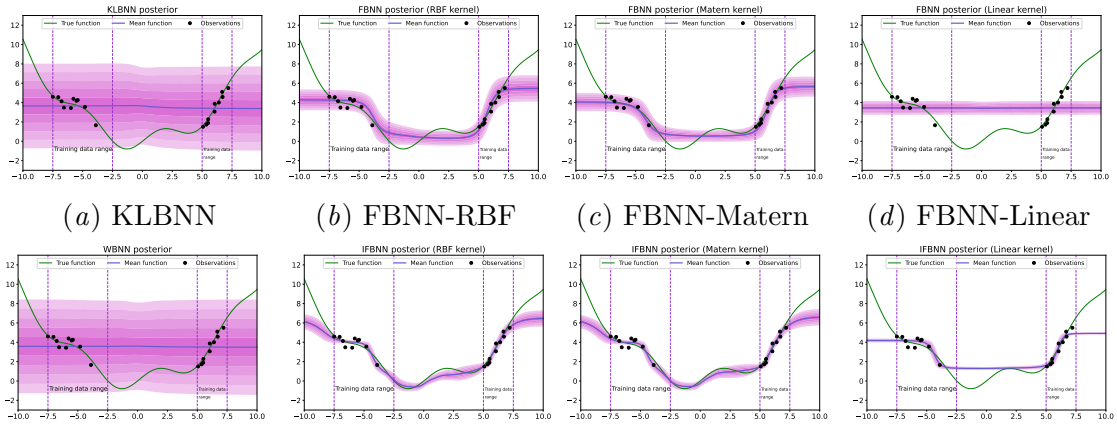
Figure 1: Learning periodic structure. The green line is the ground true function and blue lines correspond to mean approximate posterior predictions. Black dots denote 20 training points, and we also use 40 inducing points for the sampling of random marginal measurement points in IFBNN and FBNN. Shadow areas represent the predictive standard deviations. The leftmost column shows two weight-space BNNs, and the other three columns are the results of functional approaches under GP priors with three different kernels. For more details, see Appendix B.

WBNN. We also compare with functional BNNs (FBNN) (Sun et al., 2019). Meanwhile, for both IFBNN and FBNN, we use Matern kernel and linear kernel (not suitable for modeling periodicity). As shown in Figure 1, the two parametric methods fail to fit the target function. For IFBNN and FBNN with different GP prior kernels, the objective structure capture results are significantly different. (f) and (g) show that IFBNN matching GP priors incorporated periodic information is able to recover the periodic pattern in both observation and non-observation areas. And due to the linear kernel, (h) on the other hand, reflects a certain linear trend, which indicates that the functional prior matching term of IFBNN can play a significant role in the inference process. In contrast, FBNN fits less well under the corresponding kernel functions, and is less responsive to different kernel information and to uncertainty in observations and non-observations. We also provide results from GPi-FBNN in Appendix C.

Learning gap structure Consider a toy polynomial function $y(x) = \sin(x) + 0.1x^2 + \epsilon$ with segmented interval observations, where $\epsilon \sim \mathcal{N}(0, 0.5)$. 20 observations are randomly sampled from piecewise interval $[-7.5, -2.5] \cup [5, 7.5]$. From Figure 2, it is obvious to see that IFBNN has stronger capacity to recover the key characteristics of the truly function based on prior knowledge from appropriate RBF kernel and Matern kernel. Moreover, the posterior predictive variance is significantly larger in three separate non-observed regions compared to data sampled intervals, which illustrates the superior uncertainty quantification ability of our method.

4.2. Evaluation of predictive performance

We also evaluate our method on benchmark UCI regression tasks compared with: KLBNN, WBNN, FBNN and Gaussian Wasserstein Inference (GWI) (Wild et al., 2022). Table 1



(a) KLBNN (b) FBNN-RBF (c) FBNN-Matern (d) FBNN-Linear

(e) WBNN (f) IFBNN-RBF (g) IFBNN-Matern (h) IFBNN-Linear

Figure 2: Learning gap structure. For prior GP, we consider three kernel functions: RBF, Matern and Linear. The marginal measurement set for IFBNN and FBNN is randomly sampled from all observations together with 40 inducing points randomly sampled from [10, 10].

shows the average results of root mean square error (RMSE). All three functional BNNs or functional inference methods provide better results than weight-space approaches, which could reflect the contributions of functional BNNs. And our IFBNN outperforms all other functional methods.

Table 1: The table shows the average RMSE on several UCI regression tasks. We split each dataset randomly into 90% of training data and 10% of test data. This process is repeated 10 times to ensure validity. In each iteration, we randomly sample 40 points from training data to form marginal measurement set. See Appendix C for results about test negative log-likelihood (NLL).

Dataset	IFBNN	GWI	FBNN	WBNN	KLBNN
Yacht	**1.249±0.090**	2.198 ± 0.083	1.523 ± 0.075	2.328 ± 0.091	2.131 ± 0.085
Boston	**1.439±0.087**	1.742 ± 0.046	1.683 ± 0.122	2.306 ± 0.102	1.919 ± 0.074
Concrete	**1.072±0.068**	1.297 ± 0.053	1.274 ± 0.049	2.131 ± 0.068	1.784 ± 0.063
Wine	**1.209±0.054**	1.680 ± 0.064	1.528 ± 0.053	2.253 ± 0.071	1.857 ± 0.069
Kin8nm	**1.119±0.019**	1.188 ± 0.015	1.447 ± 0.069	2.134 ± 0.029	1.787 ± 0.027
Protein	**1.158±0.008**	1.333 ± 0.007	1.503 ± 0.025	2.188 ± 0.012	1.795 ± 0.010

5. Conclusion

We have proposed an indirect functional Bayesian neural network (IFBNN) based on a novel Wasserstein bridge. Unlike most existing functional BNNs, in the processes to approximate variational posterior, IFBNN could avoid the limitations of possible problematic KL divergence between distributions over functions and place a more reasonable prior over network weights benefit from the Wasserstein bridge. We have demonstrated improved performances of our IFBNN on several benchmark tasks.

References

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pages 1613–1622. PMLR, 2015.

David R Burt, Sebastian W Ober, Adrià Garriga-Alonso, and Mark van der Wilk. Understanding variational inference in function-space. *arXiv preprint arXiv:2011.09421*, 2020.

David Duvenaud, Oren Rippel, Ryan Adams, and Zoubin Ghahramani. Avoiding pathologies in very deep networks. In *Artificial Intelligence and Statistics*, pages 202–210. PMLR, 2014.

Vincent Fortuin, Adrià Garriga-Alonso, Sebastian W Ober, Florian Wenzel, Gunnar Rätsch, Richard E Turner, Mark van der Wilk, and Laurence Aitchison. Bayesian neural network priors revisited. *arXiv preprint arXiv:2102.06571*, 2021.

Yarin Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.

Yarin Gal and Zoubin Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. *Advances in Neural Information Processing Systems*, 29, 2016.

Chao Ma and José Miguel Hernández-Lobato. Functional variational inference based on stochastic process generators. *Advances in Neural Information Processing Systems*, 34, 2021.

Chao Ma, Yingzhen Li, and José Miguel Hernández-Lobato. Variational implicit processes. In *International Conference on Machine Learning*, pages 4222–4233. PMLR, 2019.

Alexander G de G Matthews, James Hensman, Richard Turner, and Zoubin Ghahramani. On sparse variational methods and the kullback-leibler divergence between stochastic processes. In *Artificial Intelligence and Statistics*, pages 231–239. PMLR, 2016.

Alexander G de G Matthews, Mark Rowland, Jiri Hron, Richard E Turner, and Zoubin Ghahramani. Gaussian process behaviour in wide deep neural networks. In *International Conference on Learning Representations*, 2018.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Tim GJ Rudner, Zonghao Chen, and Yarin Gal. Rethinking function-space variational inference in bayesian neural networks. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2020.

Tim GJ Rudner, Zonghao Chen, Yee Whye Teh, and Yarin Gal. Tractable function-space variational inference in bayesian neural networks. *Advances in Neural Information Processing Systems*, 35, 2022.

Shengyang Sun, Guodong Zhang, Jiaxin Shi, and Roger Grosse. Functional variational bayesian neural networks. In *International Conference on Learning Representations*, 2019.

Ba-Hien Tran, Dimitrios Milios, Simone Rossi, and Maurizio Filippone. Functional priors for bayesian neural networks through wasserstein distance minimization to gaussian processes. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2020.

Ba-Hien Tran, Simone Rossi, Dimitrios Milios, and Maurizio Filippone. All you need is a good functional prior for bayesian deep learning. *Journal of Machine Learning Research*, 23(74):1–56, 2022.

Veit David Wild, Robert Hu, and Dino Sejdinovic. Generalized variational inference in function spaces: Gaussian measures meet bayesian deep learning. *Advances in Neural Information Processing Systems*, 35, 2022.

Appendix A. Further Background

Gaussian Processes The measurable mapping $g : \Omega \times \mathcal{T} \to \mathcal{R}$ defined on probability space $(\Omega, \mathcal{G}, P)$ with compact index set $\mathcal{T}$ is a Gaussian process (GP) if and only if random vector $g(T) = ((g(t_1), g(t_2), ..., g(t_n))$ is multivariate Gaussian for marginals over any finite index sets $T = \{t_i\}_{i=1}^n \subset \mathcal{T}$. A GP is entirely governed by its mean function $m(t) = \mathbb{E}[g(t)]$ and covariance (kernel) function $k(t, t') = \mathbb{E}[(g(t) - m(t))(g(t') - g(t'))]$ denoted by $g \sim \mathcal{GP}(m, k)$, $t, t' \in \mathcal{T}$ (Rasmussen and Williams, 2006).

Wasserstein distance The p-Wasserstein is defined as

$$W_p(\mu, \nu) = \left(\inf_{\gamma \in \Gamma(\mu,\nu)} \int_{\mathcal{P}\times\mathcal{P}} \|x - y\|^p \, \mathrm{d}\gamma(x, y) \right)^{1/p}, \tag{7}$$

where $(\mathcal{P}, \|\cdot\|)$ is a Polish space, μ, $\nu \in (\mathcal{P}, \|\cdot\|)$ are two probability measures, $\Gamma(\mu, \nu)$ is the set of joint measures or coupling γ with marginals μ and ν on $\mathcal{P} \times \mathcal{P}$. Wild et al. (2022) proposed a functional variational inference method based on the 2-Wasserstein distance between two Gaussian measures. Tran et al. (2022) proposed an algorithm to match a functional prior with a target GP prior using the dual representation of 1-Wasserstein distance as

$$W_1(\mu, \nu) = \sup_{\|\phi\| \leq 1} \mathbb{E}_{x\sim\mu}\phi(x) - \mathbb{E}_{y\sim\nu}\phi(y), \tag{8}$$

where $\phi(\cdot) : \mathcal{P} \to \mathbb{R}$ is a 1-Lipschitz continuous function.

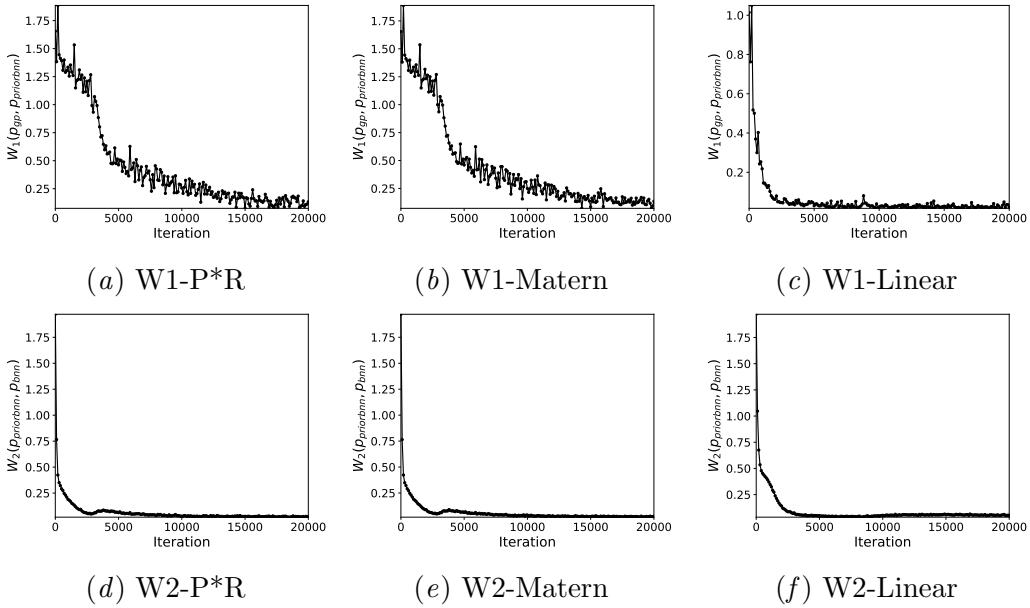
Periodic kernel * RBF kernel

$$k_{\text{Periodic * RBF}}\left(\mathbf{x}, \mathbf{x}'\right) = \exp\left(-2\frac{\sin^2\left(\frac{\pi}{p}\left(|\mathbf{x} - \mathbf{x}'|\right)\right)}{l_p}\right) * \sigma_r^2 \exp\left(-\frac{(\mathbf{x} - \mathbf{x}')^2}{2l_r^2}\right) \tag{9}$$

where p is the period length parameter, σ_r is the scaling factor, l_p and l_r are lengthscale parameters.

Appendix B. Experimental Details

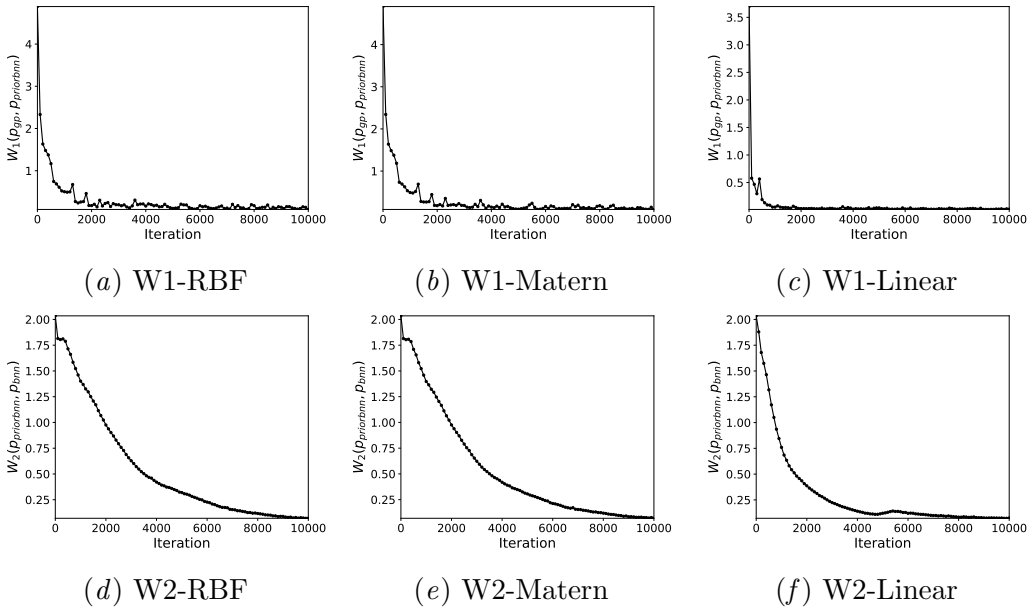
For the periodic experiment, we used a fully-connected networks with two hidden layers of 500 hidden units and trained for 20000 epochs for all models. For the gap toy example, we used BNNs with two hidden layers each with 100 hidden units and iterated 10000 epochs. For UCI regression, we used 2 hidden layer of 10 hidden units BNNs and trained for 2000 epochs. In all experiments, we first pre-train GP hyperparameters for 100 epochs on the uniformly sampled test data set for methods which use GP priors. Finite measurement points for functional approaches are randomly sampled from training data and inducing points sampled from the domain where one is to make predictions in toy examples. For all experiments we used tanh activation and Adam optimizer. The convergence processes of 1-Wasserstein distance and 2-Wasserstein of IFBNN in two toy examples are shown in Figure 3 and Figure 4.



(a) W1-P*R (b) W1-Matern (c) W1-Linear

(d) W2-P*R (e) W2-Matern (f) W2-Linear

Figure 3: Convergence of Wasserstein distances in the training of IFBNN for periodic example.



(a) W1-RBF (b) W1-Matern (c) W1-Linear

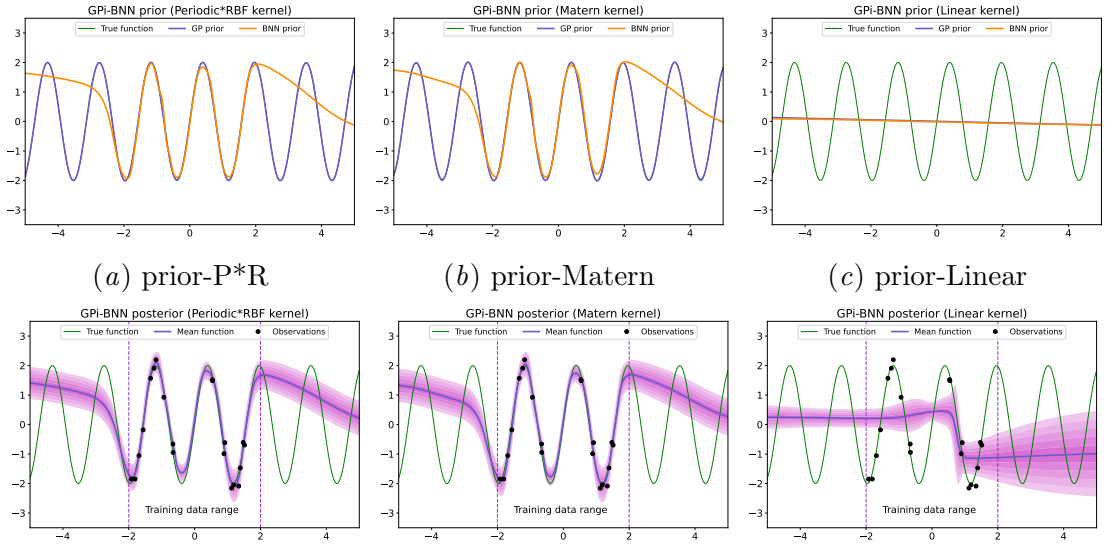
(d) W2-RBF (e) W2-Matern (f) W2-Linear

Figure 4: Convergence of Wasserstein distances in the training of IFBNN for polynomial example.

Appendix C. Further Results

C.1. Toy Results of GPi-FBNN

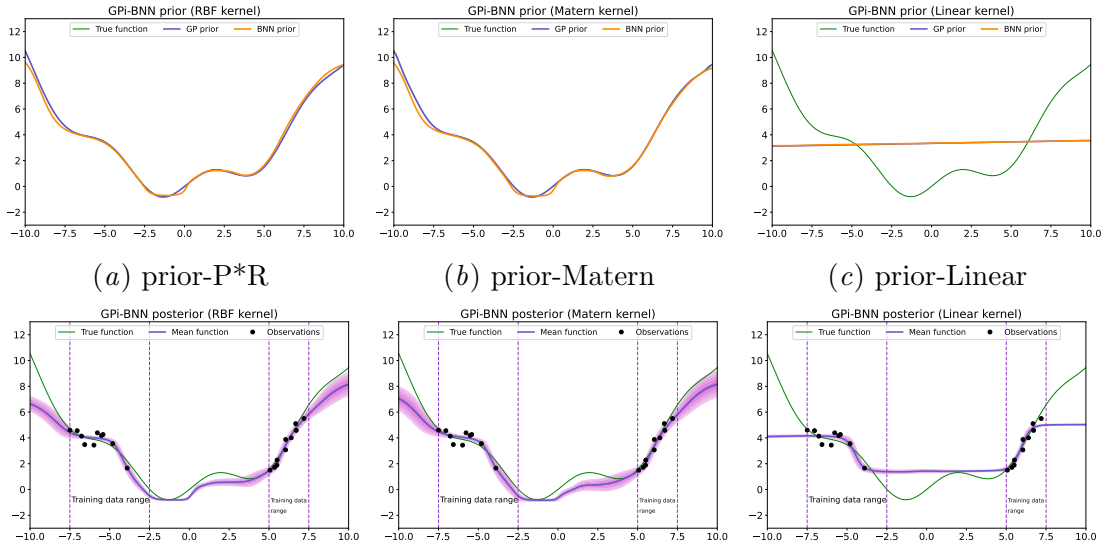
Figure 5 presents periodic structure learning results of GPi-FBNN based on three GP priors with Periodic*RBF kernel, Matern kernel and linear kernel, respectively. GP priors are pre-



(a) prior-P*R (b) prior-Matern (c) prior-Linear

(d) posterior-P*R (e) posterior-Matern (f) posterior-Linear

Figure 5: Periodic structure learning results of GPi-FBNN based on three GP priors with different kernel for three columns. Top row are the results for functional distribution matching. Green line is the ground true function, blue line is the pre-trained GP prior, yellow line is the optimal bridging distribution. Bottom row are approximate posteriors based on the optimal functional priors via variational inference over parameter space.



(a) prior-P*R (b) prior-Matern (c) prior-Linear

(d) posterior-P*R (e) posterior-Matern (f) posterior-Linear

Figure 6: Gap structure learning results of GPi-FBNN . The experimental setup and the arrangement of the results are the same as the periodic example described above.

trained on 1000 uniformly sampled test points. Marginal measurement points are sampled from 20 training data and 40 randomly sampled inducing points. It can be seen that with appropriate Periodic*RBF kernel and Matern kernel, optimal bridging distribution and

Table 2: The table shows the average test NLL on several UCI regression tasks. We split each dataset randomly into 90% of training data and 10% of test data. This process is repeated 10 times to ensure validity. In each iteration, we randomly sample 40 points from training data to form marginal measurement set.

Dataset	IFBNN	GWI	FBNN	WBNN	KLBNN
Yacht	**-1.249±1.214**	0.112 ± 0.757	-0.770 ± 0.869	2.856 ± 0.186	2.512 ± 0.161
Boston	0.324 ± 0.304	-1.043 ± 0.681	**-1.193±0.763**	2.656 ± 0.179	2.066 ± 0.115
Concrete	-0.394 ± 0.335	-0.684 ± 0.492	**-1.001±0.520**	2.838 ± 0.152	2.614 ± 0.166
Wine	**0.259±0.151**	0.700 ± 0.159	0.524 ± 0.137	2.843 ± 0.147	2.148 ± 0.125
Kin8nm	-1.014 ± 0.139	**-2.604±0.237**	-2.445 ± 0.622	2.823 ± 0.066	2.614 ± 0.071
Protein	**-2.126±0.340**	-1.575 ± 0.229	-1.486 ± 0.238	2.744 ± 0.026	2.222 ± 0.020

further obtained approximate posterior are well-fitted and are able to recover the periodic trend in the data region. However, it cannot capture the periodic features away data points as IFBNN dose, which further illustrates the advantage of IFBNN where jointly optimize bridging distribution and variational posterior. We also provide gap structure learning results of GPi-FBNN in Figure 6.

C.2. Further results for UCI Regressions

Table 2 shows the average test NLL on UCI regression tasks. IFBNN still shows competitive performance compared to other weight-space and functional methods.