Rethinking Function-Space Variational Inference in Bayesian Neural Networks

Tim G. J. Rudner [*1] **Zonghao Chen** [*2] **Yee Whye Teh** [1] **Yarin Gal** [1]

Abstract

Bayesian neural networks (BNNs) define distributions over functions induced by distributions over parameters. However, empirical evidence suggests that performing variational inference directly on the distributions over parameters and not on the induced distributions over functions can make it difficult to find good posteriors. Unfortunately, previous attempts at variational inference over the distribution over functions require approximations that do not scale to high-dimensional data or limit the class of functions to be optimized over. To address these limitations, we propose a scalable and tractable function-space variational inference method for BNNs obtained via linearization of the BNN's posterior predictive distribution. We evaluate the proposed method empirically and show that it leads to competitive predictive accuracy and reliable predictive uncertainty estimates on a range of classification tasks and performs well on a selection of relevant downstream tasks.

1. Introduction

Machine learning models succeed at an increasingly wide range of narrowly defined tasks, but fail without warning when used on inputs that are meaningfully different from the data they were trained on. To deploy machine learning models in safety-critical environments where failures are costly or may endanger human lives, machine learning methods must be reliable and have the ability to fail gracefully. As a promising tool for incorporating fail-safe mechanisms into machine learning systems, predictive uncertainty quantification allows machine learning methods to express their confidence in the correctness of their predictions.

In this paper, we develop a method for obtaining reliable uncertainty estimates in Bayesian neural networks (BNNs). While BNNs have promised to combine the advantages of deep learning and Bayesian inference, existing approaches for approximate inference in large BNNs fall short of this promise and result in approximate posterior predictive distributions that underperform non-Bayesian methods both in terms of predictive accuracy and uncertainty quantification—making them of limited use in practice. We hypothesize that this shortcoming is due to the fact that commonly used parameter-space inference methods make it difficult to define meaningful priors that effectively incorporate prior information about the data into training.

To avoid this limitation, we derive a variational objective defined explicitly in terms of the distributions over *functions* induced by a variational distribution over parameters. The proposed approach diverges from prior work on function-space variational inference in that it obtains a tractable variational objective by approximating the distributions over functions induced by the variational distribution over parameters instead of approximating the stochastic gradients. The resulting variational objective allows defining priors that explicitly encourage high uncertainty away from the training data or that contain prior information about the data and results in significantly improved predictive uncertainty estimates compared to a wide array of state-of-the-art Bayesian and non-Bayesian methods.

2. A Function-Space Perspective on Variational Inference in BNNs

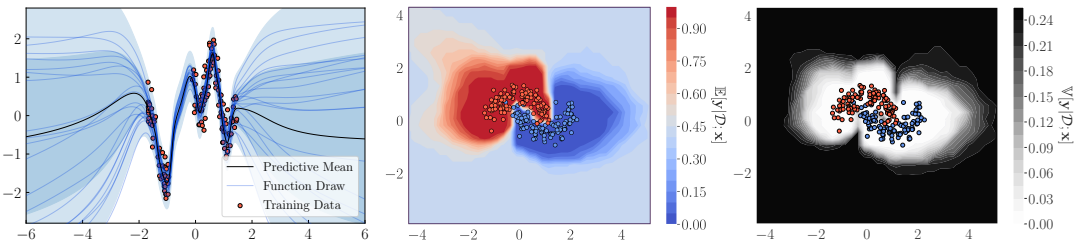
In this section, we present a function-space view of variational inference in BNNs and discuss shortcomings of prior approaches to function-space variational inference (FSVI).

We consider supervised learning tasks on data $\mathcal{D} \doteq \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N = (\mathbf{X}_\mathcal{D}, \mathbf{y})$ with inputs $\mathbf{x}_n \in \mathcal{X} \subseteq \mathbb{R}^D$ and targets $\mathbf{y}_n \in \mathcal{Y}$, where $\mathcal{Y} \subseteq \mathbb{R}^Q$ for regression and $\mathcal{Y} \subseteq \{0, 1\}^Q$ for classification tasks.

Further consider a neural network $f(\cdot\,; \boldsymbol{\Theta})$. For a conditional distribution over targets $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}; f)$, and a prior distribution over parameters $p(\boldsymbol{\theta})$, Bayesian inference provides a mathematical formalism for finding the posterior distribution over parameters given the observed data $p(\boldsymbol{\theta} \mid \mathcal{D})$ (MacKay, 1992; Neal, 1996). However, instead of defining the model in terms of the stochastic parameters $\boldsymbol{\Theta}$,

[*]Equal contribution [1]University of Oxford, Oxford, UK [2]Tsinghua University, Beijing, China. Correspondence to: Tim G. J. Rudner <tim.rudner@cs.ox.ac.uk>.

Presented at the ICML 2021 Workshop on Uncertainty and Robustness in Deep Learning., Copyright 2021 by the author(s).



(a) Predictive Distribution **(b)** Predictive Mean **(c)** Predictive Variance

Figure 1. 1D regression on the *Snelson* dataset and binary classification on the *Two Moons* dataset. The plots show the predictive distribution of a BNN, obtained via function-space variational inference (FSVI) under the local approximation described in Section 3. For a direct comparison to ensembles and MFVI, see Appendix F.

we define it explicitly in terms of the stochastic functions $f(\cdot\,;\boldsymbol{\Theta})$ induced by the stochastic parameters. To make this conceptual difference explicit going forward, we denote the probabilistic model of targets $\mathbf{y}$ given function f at a parameter realization $\boldsymbol{\theta}$ and an evaluation point $\mathbf{x} \in \mathcal{X}$ by $p(\mathbf{y} \mid f(\mathbf{x};\boldsymbol{\theta}))$. Denoting the prior distribution over functions induced by a prior distribution over parameters $p(\boldsymbol{\theta})$ by $p(f(\cdot\,;\boldsymbol{\theta}))$, we can frame the inference problem of finding a posterior distribution over functions, $p(f(\cdot\,;\boldsymbol{\theta}) \mid \mathcal{D})$, variationally as

$$\min_{q(\boldsymbol{\theta}) \in \mathcal{Q}_{\boldsymbol{\theta}}} \mathbb{D}_{\mathrm{KL}}(q(f(\cdot\,;\boldsymbol{\theta})) \,\|\, p(f(\cdot\,;\boldsymbol{\theta}) \mid \mathcal{D})),$$

where $q(f(\cdot\,;\boldsymbol{\theta}))$ is the variational distribution over functions induced by a variational distribution $q(\boldsymbol{\theta})$. As shown by Burt et al. (2021), this variational objective function is well-defined for suitably chosen prior distributions over functions. Specifically, the KL divergence between two distributions over functions generated from different distributions over parameters applied to the same mapping (e.g., the same neural network architecture) is finite if the KL divergence between the distributions over parameters is finite, since by the strong data processing inequality (Polyanskiy & Wu, 2017)

$$\mathbb{D}_{\mathrm{KL}}(q(f(\cdot\,;\boldsymbol{\theta})) \,\|\, p(f(\cdot\,;\boldsymbol{\theta}))) \leq \mathbb{D}_{\mathrm{KL}}(q(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta})). \quad (1)$$

As a result, if $\mathbb{D}_{\mathrm{KL}}(q(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta})) < \infty$, which is the case for finite-dimensional parameter vectors $\boldsymbol{\theta}$ if and only if $q(\boldsymbol{\theta})$ is absolutely continuous with respect to $p(\boldsymbol{\theta})$, then the function-space KL divergence is finite and thus well-defined as a variational objective.

Hence, for a likelihood function defined on a finite set of training targets $\mathbf{y}$ and a suitably defined prior distribution over functions, we can express the variational problem above equivalently as the well-defined maximization problem

$$\max_{q(\boldsymbol{\theta}) \in \mathcal{Q}_{\boldsymbol{\theta}}} \mathcal{F}(q(\boldsymbol{\theta}))) \doteq \max_{q(\boldsymbol{\theta}) \in \mathcal{Q}_{\boldsymbol{\theta}}} \{\mathbb{E}_{q(f(\mathbf{X}_{\mathcal{D}};\boldsymbol{\theta}))}[\log p(\mathbf{y} \mid f(\mathbf{X}_{\mathcal{D}};\boldsymbol{\theta}))] - \mathbb{D}_{\mathrm{KL}}(q(f(\cdot\,;\boldsymbol{\theta})) \,\|\, p(f(\cdot\,;\boldsymbol{\theta})))\}, \quad (2)$$

where $\mathbb{D}_{\mathrm{KL}}(q(f(\cdot\,;\boldsymbol{\theta})) \,\|\, p(f(\cdot\,;\boldsymbol{\theta})))$ is also a KL divergence between distributions over functions.

Unfortunately, if $q(f(\cdot\,;\boldsymbol{\theta}))$ and $p(f(\cdot\,;\boldsymbol{\theta}))$ are variational and prior distributions over functions, evaluating the KL divergence is intractable. To obtain a tractable objective, Sun et al. (2019) show that $\mathbb{D}_{\mathrm{KL}}(q(f(\cdot\,;\boldsymbol{\theta})) \,\|\, p(f(\cdot\,;\boldsymbol{\theta})))$ can be expressed as the supremum of the KL divergence from $q(f(\cdot\,;\boldsymbol{\theta}))$ to $p(f(\cdot\,;\boldsymbol{\theta}))$ over all *finite* sets of evaluation points, $\mathbf{X}_{\mathcal{I}}$, resulting in the objective function

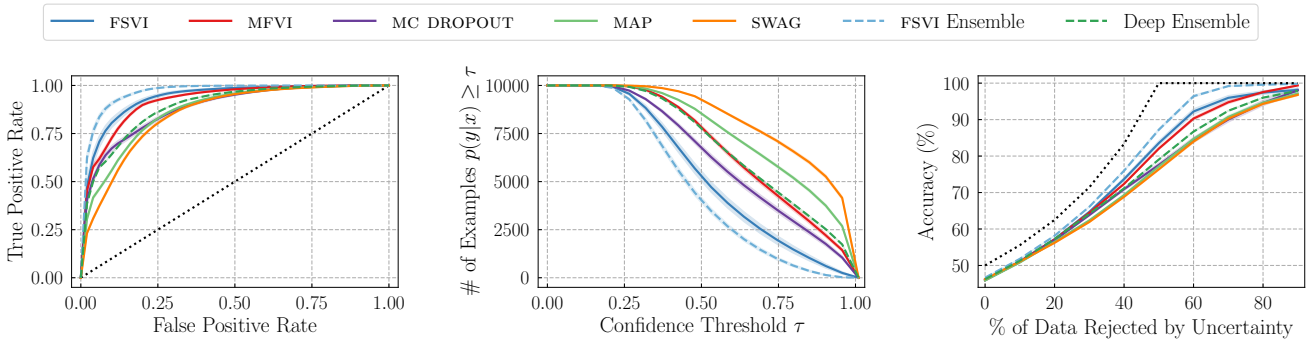
$$\mathcal{F}(q(\boldsymbol{\theta}))) = \mathbb{E}_{q(f(\mathbf{X}_{\mathcal{D}};\boldsymbol{\theta}))}[\log p(\mathbf{y} \mid f(\mathbf{X}_{\mathcal{D}};\boldsymbol{\theta}))] - \sup_{n \in \mathbb{N}, \mathbf{X}_{\mathcal{I}} \in \mathcal{X}^n} \mathbb{D}_{\mathrm{KL}}(q(f(\mathbf{X}_{\mathcal{I}};\boldsymbol{\theta})) \,\|\, p(f_{\mathbf{X}_{\mathcal{I}}})).$$

However, this objective function is still challenging to optimize in practice: The supremum cannot be obtained analytically, searching for it iteratively may lead to undesirable optimization behavior (Sun et al., 2019), and the KL divergence term itself is intractable as well—even for finite $\mathbf{X}_{\mathcal{I}}$. What's more, existing approaches to approximating the KL divergence via approximate gradients do not scale to high input or target dimensions (Sun et al., 2019).

In this paper, we propose a fundamentally different approach to function-space variational inference. Starting from Equation (2), we consider a local approximation to $q(f(\cdot\,;\boldsymbol{\theta}))$ and $p(f(\cdot\,;\boldsymbol{\theta}))$ by linearizing them about their mean parameters. Assuming a Gaussian distribution over the network parameters, this approximation turns $q(f(\cdot\,;\boldsymbol{\theta}))$ and $p(f(\cdot\,;\boldsymbol{\theta}))$ into Gaussian processes. To evaluate the resulting locally accurate KL divergence, we make a prior conditional matching assumption, which results in a tractable KL divergence evaluated at a *finite* number of evaluation points. We present this approximation in more detail next.

3. Function-Space Variational Inference

The primary obstacle to making the objective in Equation (2) tractable is the KL divergence from $q(f(\cdot\,;\boldsymbol{\theta}))$ to $p(f(\cdot\,;\boldsymbol{\theta}))$. In Appendix C, we show that under a set of variational approximations (stated in Appendix B), we can simplify the function-space variational objective to:



(a) ROC **(b)** OoD Confidence **(c)** Rejection Accuracy

Figure 2. Uncertainty evaluation metrics for out-of-distribution prediction. Models were trained on FashionMNIST and MNIST is used as out-of-distribution data. **Left**: Receiver operating characteristic for out-of-distribution detection. Curves closer to the top left are better. **Center**: Model confidence on out-of-distribution inputs. Curves closer to the bottom left are better. **Right**: Predictive uncertainty-based rejection of input samples. Curves closer to the theoretical maximum (denoted by the dotted line) are better. For additional results with other in- and out-of-distribution dataset pairs, see Appendix F. For further model and training details, see Appendix D.

Proposition 1 (Function-Space Variational Inference (FSVI))**.** *For a mapping f, stochastic parameters $\boldsymbol{\Theta}$ with mean $\mathbf{m} \doteq \mathbb{E}[\boldsymbol{\Theta}]$, and Jacobian $\mathcal{J}_{\mathbf{m}}(\cdot) \doteq \frac{\partial f(\cdot\,;\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}|_{\boldsymbol{\Theta}=\mathbf{m}}$, let the linearization of the stochastic function $f(\cdot\,;\boldsymbol{\Theta})$ about $\mathbf{m}$ be given by*

$$f(\cdot\,;\boldsymbol{\Theta}) \approx \tilde{f}(\cdot\,;\boldsymbol{\Theta}) \doteq f(\cdot\,;\mathbf{m}) + \mathcal{J}_{\mathbf{m}}(\cdot)(\boldsymbol{\Theta} - \mathbf{m}).$$

For $\boldsymbol{\Theta}$ distributed according to a mean-field Gaussian variational distribution $q(\boldsymbol{\theta})$ and a diagonal Gaussian prior $p(\boldsymbol{\theta})$, the induced distributions over functions under the linearized mapping $\tilde{f}$ about the distributions' mean parameters and are given by

$$\tilde{p}(\tilde{f}(\mathbf{x};\boldsymbol{\theta})) = \mathcal{N}(f(\mathbf{x};\boldsymbol{\mu}_0), \mathcal{J}_{\boldsymbol{\mu}_0}(\mathbf{x})\boldsymbol{\Sigma}_0\mathcal{J}_{\boldsymbol{\mu}_0}(\mathbf{x}')^\top)$$
$$\tilde{q}(\tilde{f}(\mathbf{x};\boldsymbol{\theta})) = \mathcal{N}(f(\mathbf{x};\boldsymbol{\mu}), \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{x})\boldsymbol{\Sigma}\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{x}')^\top),$$

respectively, and under the approximations in Appendix B, we obtain the function-space variational objective

$$\tilde{\mathcal{F}}(q(\boldsymbol{\theta})) \doteq \mathbb{E}_{q(f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))}[\log p(\mathbf{y} \mid f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta})] - \mathbb{D}_{\mathrm{KL}}(\tilde{q}(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})) \,\|\, \tilde{p}(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))). \tag{3}$$

Proof. See Appendix C. □

Under an additional diagonalization assumption, the final variational objective for a mini-batch $(\mathbf{X}_\mathcal{B}, \mathbf{y}_\mathcal{B})$ can be optimized via stochastic variational inference on

$$\begin{aligned}\bar{\mathcal{F}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \frac{1}{S}\sum_{k=1}^{Q}\sum_{i=1}^{S} \log p([\mathbf{y}_\mathcal{B}]_k \mid [f(\mathbf{X}_\mathcal{B}; h(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\epsilon}^{(i)}))]_k) \\ &\quad - \sum_{j=1}^{|\mathbf{X}_\mathcal{I}|} \frac{1}{2}\left(\log \frac{[K^p_{\mathcal{I}\mathcal{I}}]_{j,k}}{[K^q_{\mathcal{I}\mathcal{I}}]_{j,k}} + \frac{[K^q_{\mathcal{I}\mathcal{I}}]_{j,k}}{[K^p_{\mathcal{I}\mathcal{I}}]_{j,k}} - 1\right) \\ &\quad + \frac{[K^q_{\mathcal{I}\mathcal{I}}]_{j,k} + ([f(\mathbf{X}_\mathcal{I};\boldsymbol{\mu})]_{j,k} - [f(\mathbf{X}_\mathcal{I};\boldsymbol{\mu}_0)]_{j,k})^2}{2[K^p_{\mathcal{I}\mathcal{I}}]_{j,k}},\end{aligned} \tag{4}$$

where $K^\alpha_{\mathcal{I}\mathcal{I}} \doteq \mathrm{Cov}^{\mathrm{diag}}_\alpha(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}), \tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))$ for $\alpha \in \{q, p\}$ and $h(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\epsilon}^{(i)}) \doteq \boldsymbol{\mu} + \boldsymbol{\Sigma} \odot \boldsymbol{\epsilon}^{(i)}$ is a reparameterization of $\boldsymbol{\Theta}$ with $\boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_P)$.

After optimizing this variational objective with respect to the parameters of the variational distribution $q(\boldsymbol{\theta})$, we obtain an approximate posterior predictive distribution

$$\begin{aligned}p(\mathbf{y}_* \mid \mathbf{x}_*) &= \int p(\mathbf{y}_* \mid f(\mathbf{x}_*;\boldsymbol{\theta}))\, q(f(\mathbf{x}_*;\boldsymbol{\theta}))\, df(\mathbf{x}_*;\boldsymbol{\theta}) \\ &\approx \frac{1}{S}\sum_{i=1}^{S} p(\mathbf{y}_* | f(\mathbf{x}_*;\boldsymbol{\Theta}^{(i)})), \quad \boldsymbol{\Theta}^{(i)} \sim q(\boldsymbol{\theta}).\end{aligned}$$

4. Empirical Evaluation

We evaluate FSVI on a wide array of high-dimensional classification tasks prior work (Sun et al., 2019) was unable to scale to. We show that FSVI (sometimes *significantly*) outperforms existing Bayesian and non-Bayesian methods in terms of their in-distribution uncertainty calibration and out-of-distribution predictive uncertainty estimation. For a comprehensive description of the experiment setups, details on the models, training, and validation procedures, see Appendix D. We provide additional visualizations for a larger set of datasets and a larger selection of models in Appendix F.

4.1. Out-of-Distribution Uncertainty Estimation

To evaluate the predictive performance of FSVI, we consider a selection of image classification datasets and assess models' predictive uncertainty estimates on out-of-distribution samples. In Figures 2a, 2b, and 14 (Appendix F), we plot metrics for evaluating the ability of different methods identify and make low-confidence predictions on out-of-distribution inputs. The plots show that FSVI does not

Table 1. Comparison of in- and out-of-distribution performance metrics (mean ± standard error over ten random seeds). Best results are printed in boldface. For further details about model architectures and training, see Appendix D. *AUROC for binary in- and out-of-distribution detection on MNIST/NotMNIST. §AUROC for binary in- and out-of-distribution detection on SVHN and out-of-distribution accuracy on corrupted CIFAR-10. [1]Implemented using cited paper's code. [2]Values taken from cited paper.

Dataset	Method	Accuracy ↑	NLL ↓	ECE ↓	OOD-AUROC (M/NM)* ↑
FMNIST	MAP	91.73±0.08	0.288±0.003	0.037±0.001	87.00±0.30 / 74.85±1.31
	MFVI (Blundell et al., 2015)	91.03±0.04	0.354±0.003	0.038±0.001	93.10±0.34 / 88.88±0.74
	MFVI (tempered)	91.38±0.05	0.519±0.005	0.058±0.001	86.30±0.29 / 80.78±0.68
	MFVI (radial) (Farquhar et al., 2020a)[1]	90.31±0.11	0.340±0.001	0.035±0.001	84.40±0.68 / 82.11±1.15
	MC DROPOUT (Gal & Ghahramani, 2016)	90.55±0.04	**0.230**±0.001	0.012±0.001	88.46±0.57 / 80.02±1.04
	DUQ (van Amersfoort et al., 2020)[2]	**92.40**±0.20	–	–	95.50±0.70 / 94.60±1.80
	BNN-GLM (Immer et al., 2020)[2]	92.25±0.10	0.244±0.003	0.012±0.003	95.55±0.60 / –
	FSVI	91.04±0.17	0.256±0.004	0.011±0.002	**96.02**±0.44 / **95.41**±0.59
	FSVI (MAP init)	90.46±0.06	0.299±0.003	**0.009**±0.001	93.40±0.42 / 92.19±0.39
	SWAG (Maddox et al., 2019)[1]	92.56±0.05	0.300±0.000	0.043±0.001	85.18±0.35 / 80.31±0.30
	Deep Ensemble (Lakshminarayanan et al., 2017)	92.49±0.01	0.242±0.001	**0.019**±0.000	89.22±0.09 / 83.17±0.91
	MFVI Ensemble	92.46±0.04	0.294±0.001	0.026±0.000	94.29±0.21 / 90.31±0.37
	FSVI Ensemble	**93.34**±0.06	**0.221**±0.001	0.028±0.001	**97.35**±0.16 / **96.86**±0.22
Dataset	**Method**	**Accuracy ↑**	**NLL ↓**	**ECE ↓**	**OOD-AUROC/Acc.§ ↑**
CIFAR-10	MAP	87.35±0.09	0.491±0.005	0.070±0.001	90.86±0.43 / 74.20±0.60
	MFVI (Blundell et al., 2015)	84.04±0.07	0.372±0.002	**0.016**±0.001	92.62±0.31 / 71.48±0.74
	MFVI (tempered)	86.29±0.08	0.457±0.003	0.049±0.001	91.54±0.57 / 72.02±0.58
	MFVI (radial) (Farquhar et al., 2020a)[1]	83.99±0.18	0.510±0.001	0.048±0.002	86.04±0.18 / 73.54±0.53
	MC DROPOUT (Gal & Ghahramani, 2016)	83.89±0.18	0.412±0.005	0.018±0.002	92.69±0.49 / 69.75±0.82
	BNN-GLM (Immer et al., 2020)[2,3]	81.37±0.15	0.601±0.008	0.084±0.010	84.30±0.02 / –
	FSVI	86.34±0.11	0.499±0.005	0.061±0.001	94.00±0.39 / 73.59±0.66
	FSVI (MAP init)	**88.10**±0.08	**0.330**±0.003	0.021±0.001	**97.71**±0.11 / **79.64**±0.36
	DUQ (van Amersfoort et al., 2020)[2] (ResNet)	94.10±0.2	–	–	92.70±1.30 / –
	VOGN (Osawa et al., 2019)[2] (ResNet)	84.27±0.20	0.477±0.006	0.040±0.002	87.60±0.20 / –
	FSVI (MAP init) (ResNet)	**94.10**±0.04	**0.175**±0.002	**0.014**±0.001	**98.89**±0.23 / **81.38**±0.41
	SWAG (Maddox et al., 2019)[1]	89.73±0.14	0.480±0.001	0.067±0.002	89.79±0.50 / 76.12±0.51
	Deep Ensemble (Lakshminarayanan et al., 2017)	89.28±0.04	0.339±0.003	0.020±0.001	92.00±0.16 / 76.65±0.21
	MFVI Ensemble	89.49±0.07	0.330±0.001	0.049±0.001	94.06±0.20 / 73.67±0.38
	FSVI Ensemble	**90.17**±0.03	**0.314**±0.001	**0.018**±0.001	**96.17**±0.10 / **78.63**±0.40

make overconfident predictions on out-of-distribution inputs (Figures 2b and 14) and is the most successful method at identifying them (Figure 2a).

Table 1 presents additional results for a larger set of models, including AUROC values for identifying NotMNIST out-of-distribution images and predictive accuracy on corrupted CIFAR-10. Across all models, FSVI is the best-performing model on all out-of-distribution experiments. Further results on additional in- and out-of-distribution dataset pairs can be found in Appendix F.

4.2. In-Distribution Performance & Calibration

To evaluate in-distribution predictive performance and calibration, we compute predictive accuracies and expected calibration errors on four datasets: FashionMNIST, CIFAR-10, MNIST, and NotMNIST. The results for the first two are shown in (Table 1) and the results for the latter two are shown in Appendix F. On FashionMNIST, FSVI Ensemble achieves the highest predictive accuracy and lowest negative log-likelihood of all methods, while single-model FSVI performs competitively on both and FSVI (MAP init) achieves the lowest expected calibration error. On CIFAR-10 FSVI (MAP init; ResNet) and FSVI Ensemble outperform other methods on all metrics.

5. Conclusion

We proposed a new approach to variational inference in BNNs, where the parameters are inferred *indirectly* by performing inference over a tractable induced distribution over functions. We showed that FSVI exhibits competitive an in- and out-of-distribution predictive performance, performs well on important downstream tasks (see Appendix F.2) and can be tuned without access to out-of-distribution validation data (see Appendix H.1). We demonstrate the advantages of taking a function-space perspective on variational inference in BNNs and hope this work will inspire further research on function-space variational inference and linearization-based approximations to variational inference.

References

Benjamin, A., Rolnick, D., and Kording, K. Measuring and regularizing networks in function space. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SkMwpiR9Y7.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. Weight uncertainty in neural network. volume 37 of *Proceedings of Machine Learning Research*, pp. 1613–1622, Lille, France, 07–09 Jul 2015. PMLR. URL http://proceedings.mlr.press/v37/blundell15.html.

Burt, D. R., Ober, S. W., Garriga-Alonso, A., and van der Wilk, M. Understanding variational inference in function-space. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2021. URL https://openreview.net/forum?id=7P9y3sRa5Mk.

de G. Matthews, A. G., Hensman, J., Turner, R., and Ghahramani, Z. On sparse variational methods and the kullback-leibler divergence between stochastic processes. volume 51 of *Proceedings of Machine Learning Research*, pp. 231–239, Cadiz, Spain, 09–11 May 2016. PMLR. URL http://proceedings.mlr.press/v51/matthews16.html.

Farquhar, S., Osborne, M. A., and Gal, Y. Radial bayesian neural networks: Beyond discrete support in large-scale bayesian deep learning. In Chiappa, S. and Calandra, R. (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 1352–1362. PMLR, 26–28 Aug 2020a.

Farquhar, S., Smith, L., and Gal, Y. Liberty or depth: Deep bayesian neural nets do not need complex weight posterior approximations. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b. URL https://proceedings.neurips.cc/paper/2020/hash/2dfe1946b3003933b7f8ddd71f24dbb1-Abstract.html.

Foong, A. Y. K., Li, Y., Hernández-Lobato, J. M., and Turner, R. E. 'in-between' uncertainty in bayesian neural networks, 2019.

Foong, A. Y. K., Burt, D. R., Li, Y., and Turner, R. E. On the expressiveness of approximate inference in bayesian neural networks. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/b6dfd41875bc090bd31d0b1740eb5b1b-Abstract.html.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML 2016, pp. 1050–1059, 2016.

Graves, A. Practical variational inference for neural networks. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, pp. 2348–2356, Red Hook, NY, USA, 2011. Curran Associates Inc. ISBN 9781618395993.

Hinton, G. E. and van Camp, D. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the Sixth Annual Conference on Computational Learning Theory*, COLT '93, pp. 5–13, New York, NY, USA, 1993. Association for Computing Machinery. ISBN 0897916115.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. Stochastic variational inference. *Journal of Machine Learning Research*, 14(1):1303–1347, May 2013. ISSN 1532-4435.

Immer, A., Korzepa, M., and Bauer, M. Improving predictions of bayesian neural networks via local linearization, 2020.

Izmailov, P., Maddox, W. J., Kirichenko, P., Garipov, T., Vetrov, D., and Wilson, A. G. Subspace inference for bayesian deep learning. In Adams, R. P. and Gogate, V. (eds.), *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, volume 115 of *Proceedings of Machine Learning Research*, pp. 1169–1179. PMLR, 22–25 Jul 2020. URL http://proceedings.mlr.press/v115/izmailov20a.html.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 8571–8580. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/8076-neural-tangent-kernel-convergence-and-generalization-in-neural-networks.pdf.

Khan, M. E. E., Immer, A., Abedi, E., and Korzepa, M. Approximate inference turns deep networks into gaussian processes. In Wallach, H., Larochelle, H., Beygelzimer, A., d' Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 3094–3104. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/8573-approximate-inference-turns-deep-networks-into-gaussian-processes.pdf.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N. C., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., and Hadsell, R. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114:3521 – 3526, 2017.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6402–6413, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html.

Ma, C., Li, Y., and Hernandez-Lobato, J. M. Variational implicit processes. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4222–4233. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/ma19b.html.

MacKay, D. J. C. A practical bayesian framework for backpropagation networks. *Neural Comput.*, 4(3):448–472, May 1992. ISSN 0899-7667. doi: 10.1162/neco.1992.4.3.448. URL https://doi.org/10.1162/neco.1992.4.3.448.

Maddox, W. J., Izmailov, P., Garipov, T., Vetrov, D. P., and Wilson, A. G. A simple baseline for bayesian uncertainty in deep learning. In *Advances in Neural Information Processing Systems*, pp. 13153–13164, 2019.

Murphy, K. P. *Machine learning : a probabilistic perspective*. MIT Press, Cambridge, Mass. [u.a.], 2013. ISBN 9780262018029 0262018020.

Neal, R. M. Bayesian Learning for Neural Networks. 1996.

Nguyen, C. V., Li, Y., Bui, T. D., and Turner, R. E. Variational continual learning. *International Conference on Learning Representations*, 2018.

Ober, S. W. and Aitchison, L. Global inducing point variational posteriors for bayesian neural networks and deep gaussian processes, 2020.

Osawa, K., Swaroop, S., Khan, M. E. E., Jain, A., Eschenhagen, R., Turner, R. E., and Yokota, R. Practical deep learning with bayesian principles. In Wallach, H., Larochelle, H., Beygelzimer, A., d' Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32, pp. 4287–4299. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/b53477c2821c1bf0da5d40e57b870d35-Paper.pdf.

Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J., Lakshminarayanan, B., and Snoek, J. Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems 32*. 2019.

Pan, P., Swaroop, S., Immer, A., Eschenhagen, R., Turner, R. E., and Khan, M. E. Continual deep learning by functional regularisation of memorable past. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/2f3bbb9730639e9ea48f309d9a79ff01-Abstract.html.

Polyanskiy, Y. and Wu, Y. Strong data-processing inequalities for channels and bayesian networks. In Carlen, E., Madiman, M., and Werner, E. M. (eds.), *Convexity and Concentration*, pp. 211–249, New York, NY, 2017. Springer New York. ISBN 978-1-4939-7005-6.

Schervish, M. J. *Theory of Statistics*. Springer-Verlag, New York, NY, 1995.

Snelson, E. and Ghahramani, Z. Sparse gaussian processes using pseudo-inputs. In Weiss, Y., Schölkopf, B., and Platt, J. C. (eds.), *Advances in Neural Information Processing Systems 18*, pp. 1257–1264. MIT Press, 2006. URL http://papers.nips.cc/paper/2857-sparse-gaussian-processes-using-pseudo-inputs.pdf.

Sun, S., Zhang, G., Shi, J., and Grosse, R. B. Functional variational bayesian neural networks. In *7th International Conference on Learning Representations, ICLR*

2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net, 2019. URL https://openreview.net/forum?id=rkxacs0qY7.

Titsias, M. K. Variational learning of inducing variables in sparse gaussian processes. In van Dyk, D. and Welling, M. (eds.), *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pp. 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL http://proceedings.mlr.press/v5/titsias09a.html.

Titsias, M. K., Schwarz, J., de G. Matthews, A. G., Pascanu, R., and Teh, Y. W. Functional regularisation for continual learning with gaussian processes. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HkxCzeHFDB.

van Amersfoort, J., Smith, L., Teh, Y. W., and Gal, Y. Uncertainty estimation using a single deep deterministic neural network. In *International Conference on Machine Learning*, 2020.

van Amersfoort, J., Smith, L., Jesson, A., Key, O., and Gal, Y. Variational deterministic uncertainty quantification, 2021. URL https://openreview.net/forum?id=8W7LTo_zxdE.

Wainwright, M. J. and Jordan, M. I. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers Inc., Hanover, MA, USA, 2008. ISBN 1601981848.

Wang, Z., Ren, T., Zhu, J., and Zhang, B. Function space particle optimization for bayesian neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BkgtDsCcKQ.

Wenzel, F., Roth, K., Veeling, B., Swiatkowski, J., Tran, L., Mandt, S., Snoek, J., Salimans, T., Jenatton, R., and Nowozin, S. How good is the Bayes posterior in deep neural networks really? In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10248–10259. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/wenzel20a.html.

Supplementary Materials

A. Background

A.1. Bayesian Neural Networks.

Consider a neural network $f(\cdot\,; \boldsymbol{\Theta})$, defined in terms of stochastic parameters $\boldsymbol{\Theta} \in \mathbb{R}^P$. For a conditional distribution over targets $p(\mathbf{y} \,|\, \mathbf{x}, \boldsymbol{\theta}; f)$, and a prior distribution over parameters $p(\boldsymbol{\theta})$, Bayesian inference provides a mathematical formalism for finding the posterior distribution over parameters given the observed data $p(\boldsymbol{\theta} \,|\, \mathcal{D})$ (MacKay, 1992; Neal, 1996). However, since neural networks are non-linear in their parameters, exact inference over the stochastic network parameters is analytically intractable. Variational inference is an approximate method that seeks to avoid this intractability by framing posterior inference as a variational optimization problem. Following this approach, we can obtain a Bayesian neural network (BNN) defined in terms of a variational distribution over parameters $q(\boldsymbol{\theta})$ by solving the maximization problem

$$\min_{q(\boldsymbol{\theta}) \in \mathcal{Q}_{\boldsymbol{\theta}}} \mathbb{D}_{\mathrm{KL}}(q(\boldsymbol{\theta}) \parallel p(\boldsymbol{\theta} \,|\, \mathcal{D})) \iff \max_{q(\boldsymbol{\theta}) \in \mathcal{Q}_{\boldsymbol{\theta}}} \left\{ \mathbb{E}_{q(\boldsymbol{\theta})}[\log p(\mathbf{y} \,|\, \mathbf{X}_{\mathcal{D}}, \boldsymbol{\theta}; f)] - \mathbb{D}_{\mathrm{KL}}(q(\boldsymbol{\theta}) \parallel p(\boldsymbol{\theta})) \right\},$$

where $\mathcal{Q}_{\boldsymbol{\theta}}$ is a variational family of distributions. If $\mathcal{Q}_{\boldsymbol{\theta}}$ is the family of mean-field Gaussian distributions and the prior distribution over parameters $p(\boldsymbol{\theta})$ is a diagonal Gaussian as well, the resulting variational objective is amenable to stochastic variational inference and can be optimized using gradient-based methods. (Blundell et al., 2015; Graves, 2011; Hinton & van Camp, 1993; Hoffman et al., 2013; Wainwright & Jordan, 2008). Henceforth, we will refer to BNN inference methods that make these variational assumptions as parameter-space mean-field variational inference (MFVI).

While MFVI is compatible with techniques used in modern deep learning and can be scaled to large networks, prior work has identified several empirical issues with parameter-space MFVI in BNNs. For example, parameter-space MFVI has been shown to suffer from a decreasing signal-to-noise ratio in the expected log-likelihood gradients and requires ad-hoc fixes to stabilize training, such as modifying the variational objective via temperature scaling (Wenzel et al., 2020) and changes to the gradient estimator (Farquhar et al., 2020a).

A.2. Related Work

There is a growing body of work on function-space approaches to inference in BNNs, deep learning, and applications such as continual learning (Benjamin et al., 2019; Burt et al., 2021; Jacot et al., 2018; Khan et al., 2019; Pan et al., 2020; Sun et al., 2019; Titsias et al., 2020).

Previously proposed methods for FSVI in BNNs were based on approximate gradient estimators and either replaced the supremum in the objective derived in Sun et al. (2019) with an expectation or did not define an explicit variational objective at all (Wang et al., 2019). More recent work has attempted to circumvent the intractability of the variational objective in Equation (2) by proposing alternative objectives derived from the weight space-function space duality in BNNs and Bayesian linear models (Ma et al., 2019; Ober & Aitchison, 2020). In a similar vein, Immer et al. (2020) follow Khan et al. (2019) to explore the distribution over functions induced by different BNN models and show that approximating BNN posterior distribution via the Laplace and Generalized-Gauss-Newton approximation corresponds to changing the original mapping, $f(\cdot\,; \boldsymbol{\Theta})$, to the linearization in Approximation 4. Unlike in our approach, they do not use the linearization to train a BNN from scratch, but instead apply the linearization around the maximum a-posteriori (MAP) parameters obtained by training a deterministic neural network with Tikhonov regularization (Murphy, 2013). Their result provides an alternative view of the linearization used in our method, namely that it corresponds to the posterior predictive distribution of the un-linearized BNN under the Laplace and Generalized-Gauss-Newton approximations.

Burt et al. (2021) consider the function-space variational objective in Equation (2), and highlight advantages and limitations of employing the KL divergence in this setting. They show that minimizing the KL divergence between a wide class of parametric distributions, such as those induced by a finite-width BNN, and the posterior induced by a (non-degenerate) BNN results in an ill-defined objective. We note that this result does not immediately affect the results presented in Section 3, unless the *un-linearized* prior distribution over function were chosen to be a non-degenerate BNN. A complementary line of research showed that posterior predictive distributions of shallow BNNs with mean-field variational distributions have a limited ability to represent complex covariance structures in function space (Foong et al., 2019; 2020), but that deep BNNs do not suffer from this limitation Farquhar et al. (2020b).

B. Approximations

To obtain a tractable distribution over functions and a tractable KL divergence, we make two variational assumptions and two approximations to obtain a tractable estimator for the function-space variational objective.

We consider the distribution over parameters that gives rise to the distribution over functions $q(f(\cdot\,;\boldsymbol{\theta}))$ and assume that the variational distribution over functions is induced by a mean-field Gaussian distribution over parameters:

Approximation 1 (Gaussian Mean-Field Variational Distribution over Parameters)**.** *Assume a factorized Gaussian variational distribution over parameters, $q(\boldsymbol{\theta}) \doteq \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Define a variational distribution over functions $q(f(\cdot\,;\boldsymbol{\theta}))$ as the distribution induced by the variational distribution over parameters $q(\boldsymbol{\theta})$ under the mapping f.*

We further make an assumption about how the distribution over functions $q(f(\cdot\,;\boldsymbol{\theta})) = q(f(\mathbf{X}_*;\boldsymbol{\theta}), f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))$ factorizes, where $\mathbf{X}_\mathcal{I}$ is a finite set of so-called inducing points and $\mathbf{X}_* \doteq \mathcal{X}\backslash\mathbf{X}_\mathcal{I}$ is an infinite set of evaluation points containing all points in the data space except for $\mathbf{X}_\mathcal{I}$. Specifically, we assume prior conditional matching, that is:

Approximation 2 (Prior Conditional Matching (Titsias, 2009; de G. Matthews et al., 2016))**.** *Let the variational distribution over functions factorize as*

$$q(f(\mathbf{X}_*;\boldsymbol{\theta}), f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})) \doteq p(f(\mathbf{X}_*) \mid f(\mathbf{X}_\mathcal{I}))q(f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})),$$

where $p(f(\mathbf{X}_*) \mid f(\mathbf{X}_\mathcal{I}))$ is the conditional prior distribution over functions under the mapping f and some prior distribution over parameters $p(\boldsymbol{\theta})$.

Prior conditional matching has previously been proposed in Gaussian processes, where the prior conditional distribution over functions, $p(f(\mathbf{X}_*) \mid f(\mathbf{X}_\mathcal{I}))$, is a conditional Gaussian distribution. In contrast, in BNNs, we do not have access to an explicit parameterization of $p(f(\mathbf{X}_*) \mid f(\mathbf{X}_\mathcal{I}))$.

Remark 1. *The prior conditional matching assumption places a constraint on the marginal distributions of $q(f(\cdot\,;\boldsymbol{\theta}))$, namely that for any given $\mathbf{X}_\mathcal{I}$ any $\mathbf{X} \subset \mathbf{X}_*$ and $\mathbf{X}^c \doteq \mathbf{X}_*\backslash\mathbf{X}$,*

$$q(f(\mathbf{X};\boldsymbol{\theta})) = \int p(f(\mathbf{X}\,;\boldsymbol{\theta}), f(\mathbf{X}^c\,;\boldsymbol{\theta}) \mid f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))\, q(f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))\, df(\mathbf{X}^c;\boldsymbol{\theta})\, df(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}).^{1} \tag{B.1}$$

However, in a BNN this constraint would require being able to compute the marginal distribution

$$q(f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta})) = \int p(f(\mathbf{X}_\mathcal{D}\,;\boldsymbol{\theta}), f(\mathbf{X}^c\,;\boldsymbol{\theta}) \mid f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))\, q(f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))\, df(\mathbf{X}^c;\boldsymbol{\theta})\, df(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}) \tag{B.2}$$

for training input points $\mathbf{X}_\mathcal{D}$, which is not possible without further approximations or parametric assumptions about the variational distribution.

To apply prior conditional matching to the function-space variational objective and maintain marginal consistency, we make the following approximation:

Approximation 3 (Marginal Consistency on Observed Data)**.** *Assume that*

$$q(f(\mathbf{X}_\mathcal{D}\,;\boldsymbol{\theta})) = \int p(f(\mathbf{X}_\mathcal{D}\,;\boldsymbol{\theta}), f(\mathbf{X}^c\,;\boldsymbol{\theta}) \mid f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))\, q(f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))\, df(\mathbf{X}^c;\boldsymbol{\theta})\, df(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}).$$

Remark 2. *While we cannot enforce this constraint explicitly, as we do not have access to $p(f(\mathbf{X}_\mathcal{D}\,;\boldsymbol{\theta}), f(\mathbf{X}^c\,;\boldsymbol{\theta}) \mid f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))$ and, even if we did, evaluating the integral analytically would be infeasible, we use a sampling scheme for the inducing inputs $\mathbf{X}_\mathcal{X}$ that encourages the variational distribution over functions $q(f(\cdot\,;\boldsymbol{\theta}))$ to approximately obey the marginal consistency constraint. In doing so, we diverge from the literature on variational inference in Gaussian process models, which treats $\mathbf{X}_\mathcal{I}$ as variational parameters to optimize over. Instead, we treat the inducing inputs $\mathbf{X}_\mathcal{I}$ (and their selection) as part of the function-space variational objective estimator and use them to approximately enforce the marginal consistency constraint.*

Finally, we consider a linearization of the mapping f, which we will use to obtain a tractable estimator of the function-space KL divergence:

[1] We note that technically, the integrals are not well-defined, since there is no infinitely dimensional Lebesgue measure and use this notation for ease of exposition. For a detailed discussion of prior conditional matching and marginalization of infinite-dimensional measures, see de G. Matthews et al. (2016).

Approximation 4 (Linearization about Parameters)**.** *For mapping a f, stochastic parameters $\boldsymbol{\Theta}$ with mean $\mathbf{m} \doteq \mathbb{E}[\boldsymbol{\Theta}]$, and Jacobian $\mathcal{J}_{\mathbf{m}}(\cdot) \doteq \frac{\partial f(\cdot\,;\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}|_{\boldsymbol{\Theta}=\mathbf{m}}$, define the linearization of the stochastic function $f(\cdot\,;\boldsymbol{\Theta})$ about $\mathbf{m}$ by*

$$f(\cdot\,;\boldsymbol{\Theta}) \approx \tilde{f}(\cdot\,;\boldsymbol{\Theta}) \doteq f(\cdot\,;\mathbf{m}) + \mathcal{J}_{\mathbf{m}}(\cdot)(\boldsymbol{\Theta} - \mathbf{m}).$$

Due to local linearity, the approximation $\tilde{f}(\cdot\,;\boldsymbol{\Theta})$ will be accurate for realizations $\boldsymbol{\theta}$ close to $\boldsymbol{\mu}$, and hence, the distribution over the linearized stochastic function $\tilde{f}(\cdot\,;\boldsymbol{\Theta})$ (induced by a distribution over the parameters $\boldsymbol{\Theta}$), denoted by $\tilde{q}(\tilde{f}(\cdot\,;\boldsymbol{\theta}))$ will be close to the distribution over $f(\cdot\,;\boldsymbol{\Theta})$ for small variance parameters $\boldsymbol{\Sigma}$.

C. Proofs & Derivations

C.1. Linearization under Linearized Function Mapping

Lemma 1 (Distribution under Linearized Mapping)**.** *Consider $\boldsymbol{\Theta} \sim g(\boldsymbol{\theta})$, $f(\mathbf{X};\boldsymbol{\Theta})$, and $\tilde{f}(\mathbf{X};\boldsymbol{\Theta})$ as defined above. For a multivariate Gaussian distribution $g(\boldsymbol{\theta})$ with mean $\boldsymbol{\mu} \doteq \mathbb{E}[\boldsymbol{\Theta}]$ and diagonal covariance $\boldsymbol{\Sigma} \doteq \mathbb{V}[\boldsymbol{\Theta}]$, the mean and variance of the distribution over the linearized mapping $\tilde{g}(\tilde{f}(\mathbf{X};\boldsymbol{\theta}))$ are given by*

$$\mathbb{E}_{\tilde{q}(\tilde{f}(\mathbf{X};\boldsymbol{\theta}))}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})] = f(\mathbf{X};\boldsymbol{\mu}) \quad \textit{and} \quad \mathbb{V}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})] = \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\boldsymbol{\Sigma}\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top,$$

and the distribution $\tilde{g}$ over $\tilde{f}(\mathbf{X};\boldsymbol{\theta})$ is given by

$$\tilde{g}(\tilde{f}(\mathbf{X});\boldsymbol{\theta}) = \mathcal{N}(f(\mathbf{X};\boldsymbol{\mu}), \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\boldsymbol{\Sigma}\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top). \tag{C.3}$$

Proof. Since $\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta} \,|\, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, and $\tilde{f}(\mathbf{X};\boldsymbol{\Theta}) = f(\mathbf{X};\boldsymbol{\mu}) + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})(\boldsymbol{\Theta} - \boldsymbol{\mu})$ is a linear transformation of $\boldsymbol{\Theta}$, $\tilde{f}(\mathbf{X};\boldsymbol{\Theta})$ is a multivariate Gaussian distribution

$$\tilde{g}(\tilde{f}(\mathbf{X});\boldsymbol{\theta}) = \mathcal{N}(m(\mathbf{X}), S(\mathbf{X},\mathbf{X}')) \tag{C.4}$$

with some predictive mean $m(\mathbf{X})$ and predictive covariance $S(\mathbf{X},\mathbf{X}')$. To find $\tilde{g}(\tilde{f}(\mathbf{X};\boldsymbol{\theta}))$, we need to find the predictive mean $m(\mathbf{X})$ and the predictive covariance $S(\mathbf{X},\mathbf{X}')$, which, by definition, we can write as:

$$m(\mathbf{X}) = \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})] \tag{C.5}$$

and

$$\begin{aligned} S(\mathbf{X},\mathbf{X}') &= \mathrm{Cov}(\tilde{f}(\mathbf{X};\boldsymbol{\theta}), \tilde{f}(\mathbf{X}';\boldsymbol{\theta})) && \text{(C.6)}\\ &= \mathbb{E}[(\tilde{f}(\mathbf{X};\boldsymbol{\theta}) - \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})])\,(\tilde{f}(\mathbf{X}';\boldsymbol{\theta}) - \mathbb{E}[\tilde{f}(\mathbf{X}';\boldsymbol{\theta})])^\top]. && \text{(C.7)} \end{aligned}$$

To see that $m(\mathbf{X}) = \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})] = f(\mathbf{X};\boldsymbol{\mu})$, note that, by linearity of expectation, we have

$$\begin{aligned} m(\mathbf{X}) &= \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})] && \text{(C.8)}\\ &= \mathbb{E}[f(\mathbf{X};\boldsymbol{\mu}) + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})(\boldsymbol{\Theta} - \boldsymbol{\mu})] && \text{(C.9)}\\ &= f(\mathbf{X};\boldsymbol{\mu}) + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})(\mathbb{E}[\boldsymbol{\Theta}] - \boldsymbol{\mu}) && \text{(C.10)}\\ &= f(\mathbf{X};\boldsymbol{\mu}). && \text{(C.11)} \end{aligned}$$

To see that $S(\mathbf{X},\mathbf{X}') = \mathrm{Cov}(\tilde{f}(\mathbf{X};\boldsymbol{\theta}), \tilde{f}(\mathbf{X}';\boldsymbol{\theta})) = \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\boldsymbol{\Sigma}\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top$, note that in general, $\mathrm{Cov}(\mathbf{X},\mathbf{X}) = \mathbb{E}[\mathbf{X}\mathbf{X}^\top] + \mathbb{E}[\mathbf{X}]\mathbb{E}[\mathbf{X}]^\top$, and hence,

$$\mathrm{Cov}(\tilde{f}(\mathbf{X};\boldsymbol{\theta}), \tilde{f}(\mathbf{X}';\boldsymbol{\theta})) = \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})\tilde{f}(\mathbf{X}';\boldsymbol{\theta})^\top] - \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})]\mathbb{E}[\tilde{f}(\mathbf{X}';\boldsymbol{\theta})]^\top. \tag{C.12}$$

We already know that $\mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})] = f_{\boldsymbol{\mu}}(\mathbf{X})$, so we only need to find $\mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})\tilde{f}(\mathbf{X}';\boldsymbol{\theta})^\top]$:

$$
\begin{aligned}
&\mathbb{E}_{q(\boldsymbol{\theta})}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})\tilde{f}(\mathbf{X}';\boldsymbol{\theta})^\top] && \text{(C.13)}\\
&\quad= \mathbb{E}_{q(\boldsymbol{\theta})}[(f(\mathbf{X};\boldsymbol{\mu}) + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})(\boldsymbol{\Theta}-\boldsymbol{\mu}))(f(\mathbf{X}';\boldsymbol{\mu}) + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')(\boldsymbol{\Theta}-\boldsymbol{\mu}))^\top] \\
&\quad= \mathbb{E}_{q(\boldsymbol{\theta})}[f(\mathbf{X};\boldsymbol{\mu})f(\mathbf{X}';\boldsymbol{\mu})^\top + (\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})(\boldsymbol{\Theta}-\boldsymbol{\mu}))(\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')(\boldsymbol{\Theta}-\boldsymbol{\mu}))^\top && \text{(C.14)}\\
&\qquad\qquad + f(\mathbf{X};\boldsymbol{\mu})(\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')(\boldsymbol{\Theta}-\boldsymbol{\mu}))^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})(\boldsymbol{\Theta}-\boldsymbol{\mu})f(\mathbf{X}';\boldsymbol{\mu})^\top] \\
&\quad= \mathbb{E}_{q(\boldsymbol{\theta})}[f(\mathbf{X};\boldsymbol{\mu})f(\mathbf{X}';\boldsymbol{\mu})^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})(\boldsymbol{\Theta}-\boldsymbol{\mu})(\boldsymbol{\Theta}-\boldsymbol{\mu})^\top\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top && \text{(C.15)}\\
&\qquad\qquad + f(\mathbf{X};\boldsymbol{\mu})(\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')(\boldsymbol{\Theta}-\boldsymbol{\mu}))^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})(\boldsymbol{\Theta}-\boldsymbol{\mu})f(\mathbf{X}';\boldsymbol{\mu})^\top] \\
&\quad= f(\mathbf{X};\boldsymbol{\mu})f(\mathbf{X}';\boldsymbol{\mu})^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\mathbb{E}_{q(\boldsymbol{\theta})}[(\boldsymbol{\Theta}-\boldsymbol{\mu})(\boldsymbol{\Theta}-\boldsymbol{\mu})^\top]\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top \\
&\qquad\qquad + f(\mathbf{X};\boldsymbol{\mu})(\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')\underbrace{(\mathbb{E}_{q(\boldsymbol{\theta})}[\boldsymbol{\Theta}]-\boldsymbol{\mu})}_{=0})^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\underbrace{(\mathbb{E}_{q(\boldsymbol{\theta})}[\boldsymbol{\Theta}]-\boldsymbol{\mu})}_{=0}f(\mathbf{X}';\boldsymbol{\mu})^\top, && \text{(C.16)}
\end{aligned}
$$

where the last line follows from the definition of $q(\boldsymbol{\theta})$. By definition of the variance, we then obtain

$$
\begin{aligned}
\mathbb{E}_{q(\boldsymbol{\theta})}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})\tilde{f}(\mathbf{X}';\boldsymbol{\theta})^\top] &= f(\mathbf{X};\boldsymbol{\mu})f(\mathbf{X}';\boldsymbol{\mu})^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\mathbb{E}_{q(\boldsymbol{\theta})}[(\boldsymbol{\Theta}-\boldsymbol{\mu})(\boldsymbol{\Theta}-\boldsymbol{\mu})^\top]\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top && \text{(C.17)}\\
&= f(\mathbf{X};\boldsymbol{\mu})f(\mathbf{X};\boldsymbol{\mu})^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\,\mathbb{V}[\boldsymbol{\theta}]\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top. && \text{(C.18)}
\end{aligned}
$$

With this result, we obtain the covariance function

$$
\begin{aligned}
S(\mathbf{X},\mathbf{X}') &= \mathrm{Cov}(\tilde{f}(\mathbf{X};\boldsymbol{\theta}),\tilde{f}(\mathbf{X}';\boldsymbol{\theta})) && \text{(C.19)}\\
&= \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})\tilde{f}(\mathbf{X}';\boldsymbol{\theta})^\top] - \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})]\mathbb{E}[\tilde{f}(\mathbf{X}';\boldsymbol{\theta})]^\top && \text{(C.20)}\\
&= \mathbb{E}[\tilde{f}(\mathbf{X};\boldsymbol{\theta})\tilde{f}(\mathbf{X}';\boldsymbol{\theta})^\top] - f(\mathbf{X};\boldsymbol{\mu})f(\mathbf{X};\boldsymbol{\mu})^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\,\mathbb{V}[\boldsymbol{\Theta}]\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top && \text{(C.21)}\\
&= f(\mathbf{X};\boldsymbol{\theta})f(\mathbf{X}';\boldsymbol{\theta})^\top - f(\mathbf{X};\boldsymbol{\mu})f(\mathbf{X};\boldsymbol{\mu})^\top + \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\,\mathbb{V}[\boldsymbol{\Theta}]\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top && \text{(C.22)}\\
&= \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\,\mathbb{V}[\boldsymbol{\Theta}]\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top. && \text{(C.23)}
\end{aligned}
$$

Finally, $\mathbb{V}[\boldsymbol{\Theta}] = \boldsymbol{\Sigma}$ yields

$$
S(\mathbf{X},\mathbf{X}') = \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X})\boldsymbol{\Sigma}\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{X}')^\top, \tag{C.24}
$$

where $\boldsymbol{\Sigma}$ is a diagonal matrix. This concludes the proof. □

C.2. Function-Space Variational Objective

This proof follows steps from de G. Matthews et al. (2016). Consider measures $\hat{P}$ and P both of which define distributions over some function f, indexed by an infinite index set X. Let $\mathcal{D}$ be a dataset and let $\mathbf{X}_\mathcal{D}$ denote a set of inputs and $\mathbf{y}_\mathcal{D}$ a set of targets. Consider the measure-theoretic version of Bayes' Theorem (Schervish, 1995):

$$
\frac{d\hat{P}}{dP}(f) = \frac{p_X(Y \mid f)}{p(Y)}, \tag{C.25}
$$

where $p_X(Y \mid f)$ is the likelihood and $p(Y) = \int_{\mathbb{R}^X} p_X(Y \mid f)dP(f)$ is the marginal likelihood. We assume that the likelihood function is evaluated at a finite subset of the index set X. Denote by $\pi_C : \mathbb{R}^X \to \mathbb{R}^C$ a projection function that takes a function and returns the same function, evaluated at a finite set of points C, so we can write

$$
\frac{d\hat{P}}{dP}(f) = \frac{d\hat{P}_{\mathbf{X}_\mathcal{D}}}{dP_{\mathbf{X}_\mathcal{D}}}(\pi_{\mathbf{X}_\mathcal{D}}(f)) = \frac{p(\mathbf{y}_\mathcal{D} \mid \pi_{\mathbf{X}_\mathcal{D}}(f))}{p(\mathbf{y}_\mathcal{D})}, \tag{C.26}
$$

and similarly, the marginal likelihood becomes $p(\mathbf{y}_\mathcal{D}) = \int_{\mathbf{X}_\mathcal{D}} p(\mathbf{y}_\mathcal{D} \mid f_{\mathbf{X}_\mathcal{D}})dP_{\mathbf{X}_\mathcal{D}}(f_{\mathbf{X}_\mathcal{D}})$. Now, considering the measure-theoretic version of the KL divergence between an approximating stochastic process Q and a posterior stochastic process $\hat{P}$, we can write

$$
\mathbb{D}_{\mathrm{KL}}(Q \,\|\, \hat{P}) = \int_{\mathbb{R}^X} \log \frac{dQ}{d\hat{P}}(f)dQ(f) - \int_{\mathbb{R}^X} \log \frac{d\hat{P}}{dP}(f)dQ(f), \tag{C.27}
$$

where P is some prior stochastic process. Now, considering the second term, we can apply the measure-theoretic Bayes' Theorem to obtain

$$\int_{\mathbb{R}^X} \log \frac{d\hat{P}}{dP}(f) dQ(f) = \int_{\mathbb{R}^{\mathbf{X}_\mathcal{D}}} \log \frac{d\hat{P}_{\mathbf{X}_\mathcal{D}}}{dP_{\mathbf{X}_\mathcal{D}}}(f_{\mathbf{X}_\mathcal{D}}) dQ_{\mathbf{X}_\mathcal{D}}(f_{\mathbf{X}_\mathcal{D}}) \tag{C.28}$$

$$= \mathbb{E}_{Q_{\mathbf{X}_\mathcal{D}}}\left[\log p\left(\mathbf{y}_\mathcal{D} \mid f_{\mathbf{X}_\mathcal{D}}\right)\right] - \log p(\mathbf{y}_\mathcal{D}), \tag{C.29}$$

giving us

$$\mathbb{D}_{\mathrm{KL}}(Q \,\|\, \hat{P}) = \int_{\mathbb{R}^X} \log \frac{dQ}{dP}(f) dQ(f) - \mathbb{E}_{Q_{\mathbf{X}_\mathcal{D}}}\left[\log p\left(\mathbf{y}_\mathcal{D} \mid f_{\mathbf{X}_\mathcal{D}}\right)\right] + \log p(\mathbf{y}_\mathcal{D}). \tag{C.30}$$

Rearranging, we can get

$$p(\mathbf{y}_\mathcal{D}) = \mathbb{E}_{Q_{\mathbf{X}_\mathcal{D}}}\left[\log p\left(\mathbf{y}_\mathcal{D} \mid f_{\mathbf{X}_\mathcal{D}}\right)\right] - \int_{\mathbb{R}^X} \log \frac{dQ}{dP}(f) dQ(f) + \mathbb{D}_{\mathrm{KL}}(Q \,\|\, P)] \tag{C.31}$$

$$\geq \mathbb{E}_{Q_{\mathbf{X}_\mathcal{D}}}\left[\log p\left(\mathbf{y}_\mathcal{D} \mid f_{\mathbf{X}_\mathcal{D}}\right)\right] - \int_{\mathbb{R}^X} \log \frac{dQ}{dP}(f) dQ(f). \tag{C.32}$$

By the measure-theoretic definition of the KL divergence, we can thus write

$$p(\mathbf{y}_\mathcal{D}) \geq \mathbb{E}_{Q_{\mathbf{X}_\mathcal{D}}}\left[\log p\left(\mathbf{y}_\mathcal{D} \mid f_{\mathbf{X}_\mathcal{D}}\right)\right] - \int_{\mathbb{R}^X} \log \frac{dQ}{dP}(f) dQ(f) \tag{C.33}$$

$$= \mathbb{E}_{Q_{\mathbf{X}_\mathcal{D}}}\left[\log p\left(\mathbf{y}_\mathcal{D} \mid f_{\mathbf{X}_\mathcal{D}}\right)\right] - \mathbb{D}_{\mathrm{KL}}(Q \,\|\, P), \tag{C.34}$$

which corresponds to the expression for the function-space variational objective in Section 2.

Proposition 1 (Function-Space Variational Inference (FSVI))**.** *For a mapping f, stochastic parameters $\boldsymbol{\Theta}$ with mean $\mathbf{m} \doteq \mathbb{E}[\boldsymbol{\Theta}]$, and Jacobian $\mathcal{J}_\mathbf{m}(\cdot) \doteq \frac{\partial f(\cdot\,;\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}|_{\boldsymbol{\Theta}=\mathbf{m}}$, let the linearization of the stochastic function $f(\cdot\,;\boldsymbol{\Theta})$ about $\mathbf{m}$ be given by*

$$f(\cdot\,;\boldsymbol{\Theta}) \approx \tilde{f}(\cdot\,;\boldsymbol{\Theta}) \doteq f(\cdot\,;\mathbf{m}) + \mathcal{J}_\mathbf{m}(\cdot)(\boldsymbol{\Theta} - \mathbf{m}).$$

For $\boldsymbol{\Theta}$ distributed according to a mean-field Gaussian variational distribution $q(\boldsymbol{\theta})$ and a diagonal Gaussian prior $p(\boldsymbol{\theta})$, the induced distributions over functions under the linearized mapping $\tilde{f}$ about the distributions' mean parameters are given by

$$\tilde{p}(\tilde{f}(\mathbf{x};\boldsymbol{\theta})) = \mathcal{N}(f(\mathbf{x};\boldsymbol{\mu}_0), \mathcal{J}_{\boldsymbol{\mu}_0}(\mathbf{x})\boldsymbol{\Sigma}_0\mathcal{J}_{\boldsymbol{\mu}_0}(\mathbf{x}')^\top) \qquad \textit{and} \qquad \tilde{q}(\tilde{f}(\mathbf{x};\boldsymbol{\theta})) = \mathcal{N}(f(\mathbf{x};\boldsymbol{\mu}), \mathcal{J}_{\boldsymbol{\mu}}(\mathbf{x})\boldsymbol{\Sigma}\mathcal{J}_{\boldsymbol{\mu}}(\mathbf{x}')^\top),$$

respectively, and under the approximations in Appendix B, we obtain the function-space variational objective

$$\tilde{\mathcal{F}}(q(\boldsymbol{\theta})) \doteq \mathbb{E}_{q(f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))}[\log p(\mathbf{y} \mid f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta})] - \mathbb{D}_{\mathrm{KL}}(\tilde{q}(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})) \,\|\, \tilde{p}(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))). \tag{C.35}$$

Proof. Let $\mathbf{X}_*$ and $\mathbf{X}_\mathcal{I}$ be as defined, and approximate the distributions over functions induced by $p(\boldsymbol{\theta})$ and $q(\boldsymbol{\theta})$ under the non-linearized mapping f by $\tilde{p}(\tilde{f}(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))$ and $\tilde{q}(\tilde{f}(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))$ as derived in Lemma 1. Consider the function-space variational objective

$$\mathcal{F}(q(\boldsymbol{\theta}))) \doteq \mathbb{E}_{q(f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))}[\log p(\mathbf{y} \mid f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))] - \mathbb{D}_{\mathrm{KL}}(q(f(\cdot\,;\boldsymbol{\theta})) \,\|\, p(f(\cdot\,;\boldsymbol{\theta}))), \tag{C.36}$$

which contains a KL divergence between two distributions over functions.

To obtain a tractable estimator, we write the objective as

$$\mathcal{F}(q(\boldsymbol{\theta})) \doteq \mathbb{E}_{q(f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))}[\log p(\mathbf{y} \mid f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))] - \mathbb{D}_{\mathrm{KL}}(q(f(\mathbf{X}_*;\boldsymbol{\theta}), f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})) \,\|\, p(f(\mathbf{X}_*;\boldsymbol{\theta}), f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))), \tag{C.37}$$

and note that under the prior conditional matching assumption (see Appendix B) the variational and prior distributions over functions in the KL divergence are

$$q(f(\mathbf{X}_*;\boldsymbol{\theta}), f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})) = p(f(\mathbf{X}_*;\boldsymbol{\theta}) \mid f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))\, q(f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))$$
$$p(f(\mathbf{X}_*;\boldsymbol{\theta}), f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})) = p(f(\mathbf{X}_*;\boldsymbol{\theta}) \mid f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))\, p(f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})).$$

and the $p(f(\mathbf{X}_*;\boldsymbol{\theta}) \mid f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))$ terms in the numerator and denominator of the KL divergence cancel out so that the KL divergence simplifies to $\mathbb{D}_{\mathrm{KL}}(q(f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}) \,\|\, p(f(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))$.

To obtain an analytically tractable estimator of the KL divergence, we consider the linearized mapping $\tilde{f}$, under which, by Lemma 1, the distributions $\tilde{q}(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})$ and $\tilde{p}(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})$ in the KL divergence will both be multivariate Gaussian distributions. Since a KL divergence between two multivariate Gaussian distribution can be expressed analytically, the resulting estimator of the function-space variational objective can be estimated via Monte Carlo sampling of $q(f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))$ and $\mathbf{X}_\mathcal{I}$ to estimate the expected log-likelihood and the KL divergence, respectively. The resulting approximation to the function-space variational objective is then given by

$$\tilde{\mathcal{F}}(q(\boldsymbol{\theta})) \doteq \mathbb{E}_{q(f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))}[\log p(\mathbf{y} \mid f(\mathbf{X}_\mathcal{D};\boldsymbol{\theta}))] - \mathbb{D}_{\mathrm{KL}}(\tilde{q}(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta})) \,\|\, \tilde{p}(\tilde{f}(\mathbf{X}_\mathcal{I};\boldsymbol{\theta}))), \tag{C.38}$$

where $\tilde{p}(\tilde{f}(\cdot;\boldsymbol{\theta}))$ is the local approximation to a prior distribution over functions induced by a diagonal Gaussian prior over the network parameters. This concludes the proof. □

D. Model, Algorithmic & Experiment Details

D.1. Hyperparameter Selection

For FSVI, we used a holdout validation set (10% of the training set) to conduct a hyperparameter search over the prior variance, the number of inducing inputs used to evaluate the KL divergence, the inducing input sampling methods, and the number of Monte Carlo samples used to evaluate the expected log-likelihood. The full results can be found in Appendix H. We selected the set of hyperparameters that yielded the lowest validation log-likelihood for all experiments. We state the hyperparameters selected for the different datasets below. We used 5 Monte Carlo samples and 10 inducing inputs per gradient step for all of our experiments.

For other methods, we used a holdout validation set of the same size and selected the best-performing hyperparameters. We used implementations provided by the authors of MFVI (radial) and SWAG. All other methods were implemented from scratch unless stated otherwise in Table 2.

D.2. FashionMNIST vs. MNIST/NotMNIST

We train all model on the FashionMNIST dataset and evaluate the models' predictive uncertainty performance on out-of-distribution data on the MNIST dataset. Both datasets consist of images of size 28×28 pixels. The FashionMNIST dataset is normalized to have zero mean and a standard deviation of one. The MNIST dataset is normalized with the same transformation, that is, using the same mean and standard deviation used for the in-distribution data. We chose FashionMNIST/MNIST instead of MNIST/notMNIST because the latter is notably easier than the former.

In this experiment, a network architecture with two convolutional layers of 32 and 64 3×3 filters and a fully-connected final layer of 128 hidden units is used. A max pooling operation is placed after each convolutional layer and ReLU activations are used. We do not use batch normalization. All models are trained for 30 epochs with a mini-batch size of 200 and using the Adam optimizer with a learning rate 5×10^{-4}.

For FSVI, we used a prior variance of $\boldsymbol{\Sigma}_0 = 10$ and sampled 50% of the inducing inputs for each gradient step from the mini-batch and the other 50% according to the method described in Appendix D.7.

D.3. CIFAR-10/SVHN

We train all model on the CIFAR-10 dataset and evaluate the models' predictive uncertainty performance on out-of-distribution data on the SVHN dataset. Both datasets consist of images of size $32 \times 32 \times 3$, with RBG channels. The CIFAR-10 dataset is normalized to have zero mean and a standard deviation of one. The SVHN dataset is normalized with the same transformation, that is, using the same mean and standard deviation used for the in-distribution data. The training data is augmented with random horizontal flips (with a probability of 0.5) and random crops (4 zero pixels on all sides).

In this experiment, a network architecture with six convolutional layers of 32, 32, 64, 64, 128, 128 3×3 filters and a fully-connected final layer of 128 hidden units is used. A max pooling operation is placed after the second, fourth, and sixth convolutional layer and ReLU activations are used. We do not use batch normalization. All models are trained for 50 epochs and using the Adam optimizer with a mini-batch size of 200 and a learning rate 5×10^{-4}.

For FSVI, we used a prior variance of $\mathbf{\Sigma}_0 = 0.1$ and sampled the inducing inputs for each gradient step according to the method described in Appendix D.7.

D.4. Two Moons

In this experiment, a network architecture with with two fully-connected layer with 30 hidden units each is used. We train all models with a learning rate of 10^{-3}.

For FSVI, we used a prior variance of $\mathbf{\Sigma}_0 = 10$ and sampled inducing input randomly from $[-10, 10]^2$.

D.5. 1D Regression Problems

In this experiment, we use a model consisting of two fully-connected layers with 100 hidden units each and Tanh activations. We train all models with a learning rate of 10^{-3}.

For FSVI, we used a prior variance of $\mathbf{\Sigma}_0 = 10$ and sampled inducing input randomly from $[-10, 10]$.

D.6. Further Implementation Details

We use the Adam optimizer with default settings of $\beta_1 = 0.9$, $\beta_2 = 0.99$ and $\epsilon = 10^{-8}$ for all experiments. The deterministic neural networks that were used for the ensemble were trained with a weight decay of λ = 1e-1. MFVI (tempered) was trained with a KL scaling factor of 0.1 to obtain a cold posterior.

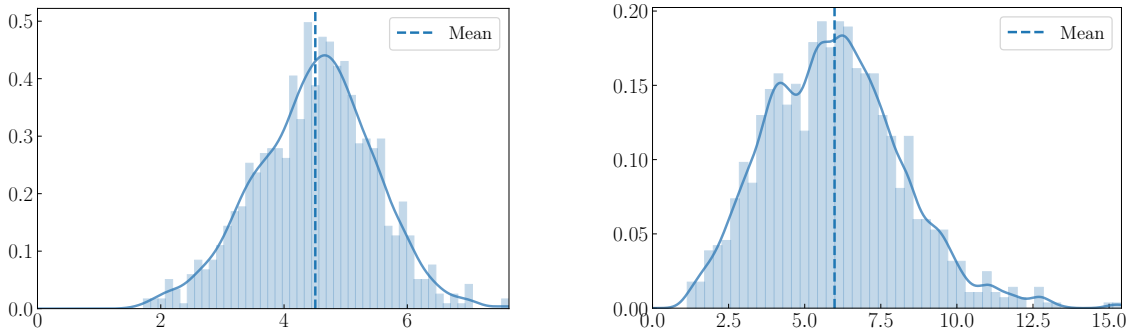
D.7. Selection of Inducing Inputs.

Ideally, the inducing set would cover the entire input space, making the prior conditional matching assumption unnecessary. However, as sampling an infinite set and evaluating the KL divergence on it is not possible both algorithmically and computationally, we instead use prior conditional matching in conjunction with an iterative sampling procedure.

To have (increasing) coverage of regions in the input space, we select a random set of inducing inputs $\mathbf{X}_\mathcal{I}$ from an infinite set $\mathcal{I} \subseteq \mathbb{R}^D$ at every gradient step. For tasks with image inputs, we construct $\mathcal{I}$ by generating images with monochromatic channels by randomly sampling a value from the empirical pixel value distribution of each channel for images in the training set. For regression tasks with a D-dimensional input space, we construct $\mathcal{I}$ by randomly sampling from a D-dimensional uniform distribution with lower and upper bounds set to the empirical lower and upper bounds of the training data. For further details on the effect of different sampling schemes on the posterior predictive distribution's performance, see Appendix F.

E. Validation of Approximating Assumptions

E.1. Validation of Linearization Assumption: FashionMNIST



(a) $\mathbb{D}_{\mathrm{KL}}(q(f(\mathbf{X}_\mathcal{E};\boldsymbol{\theta})) \,\|\, \tilde{q}(\tilde{f}(\mathbf{X}_\mathcal{E};\boldsymbol{\theta})))$ **(b)** $\mathbb{D}_{\mathrm{KL}}(\tilde{q}(\tilde{f}(\mathbf{X}_\mathcal{E};\boldsymbol{\theta})) \,\|\, q(f(\mathbf{X}_\mathcal{E};\boldsymbol{\theta})))$

Figure 3. KL divergences between distributions induced by $q(\boldsymbol{\theta})$ under linearized and non-linearized mappings evaluated on 1,000 data points $\mathbf{X}_\mathcal{E}$ sampled from the test set. The KL divergence is estimated using kernel density estimation over output dimensions for $q(f(\mathbf{X}_\mathcal{E};\boldsymbol{\theta}))$.

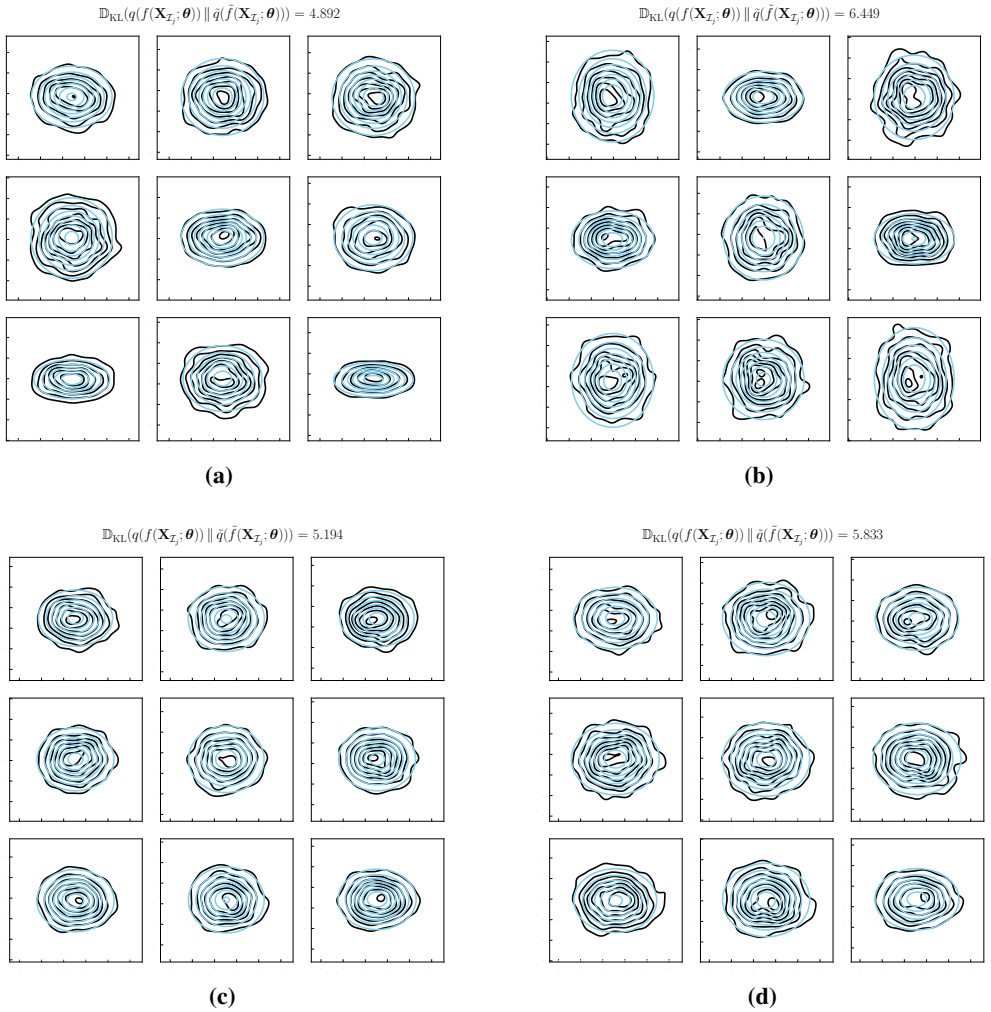
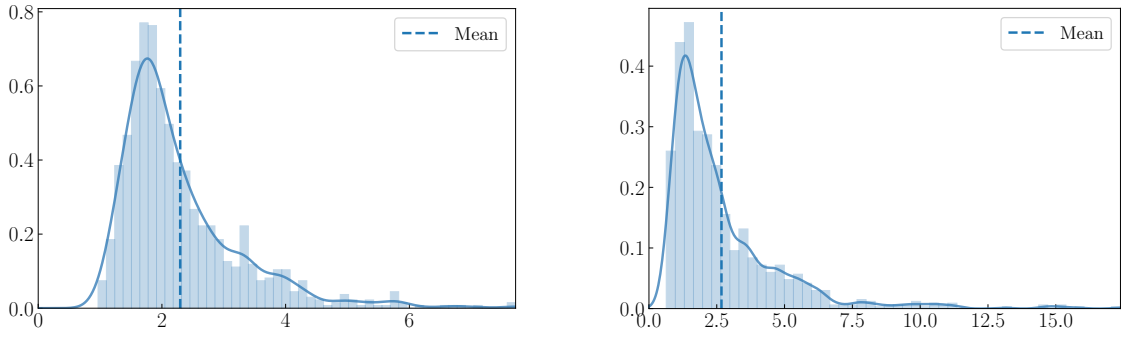


Figure 4. Distributions over functions under linearized and non-linearized mappings for a model trained on the FashionMNIST dataset. Each plot shows the covariance over stochastic functions (logits) between the first output dimension (corresponding to the first class) and all other output dimensions for a given input point sampled from the test set. The first output dimension is on the x-axis. The BNN's distribution over functions is shown in black, and the distribution over linearized functions is shown in light-blue. The title of each set of plots show the estimated KL divergence from the distribution over non-linearized functions to the distribution over linearized functions.

E.1.1. Validation of Linearization Assumption: CIFAR-10



(a) $\mathbb{D}_{\mathrm{KL}}(q(f(\mathbf{X}_{\mathcal{E}};\boldsymbol{\theta})) \,\|\, \tilde{q}(\tilde{f}(\mathbf{X}_{\mathcal{E}};\boldsymbol{\theta})))$ **(b)** $\mathbb{D}_{\mathrm{KL}}(\tilde{q}(\tilde{f}(\mathbf{X}_{\mathcal{E}};\boldsymbol{\theta})) \,\|\, q(f(\mathbf{X}_{\mathcal{E}};\boldsymbol{\theta})))$

Figure 5. KL divergences between distributions induced by $q(\boldsymbol{\theta})$ under linearized and non-linearized mappings evaluated on 1,000 data points $\mathbf{X}_{\mathcal{E}}$ sampled from the test set. The KL divergence is estimated using kernel density estimation over output dimensions for $q(f(\mathbf{X}_{\mathcal{E}};\boldsymbol{\theta}))$.

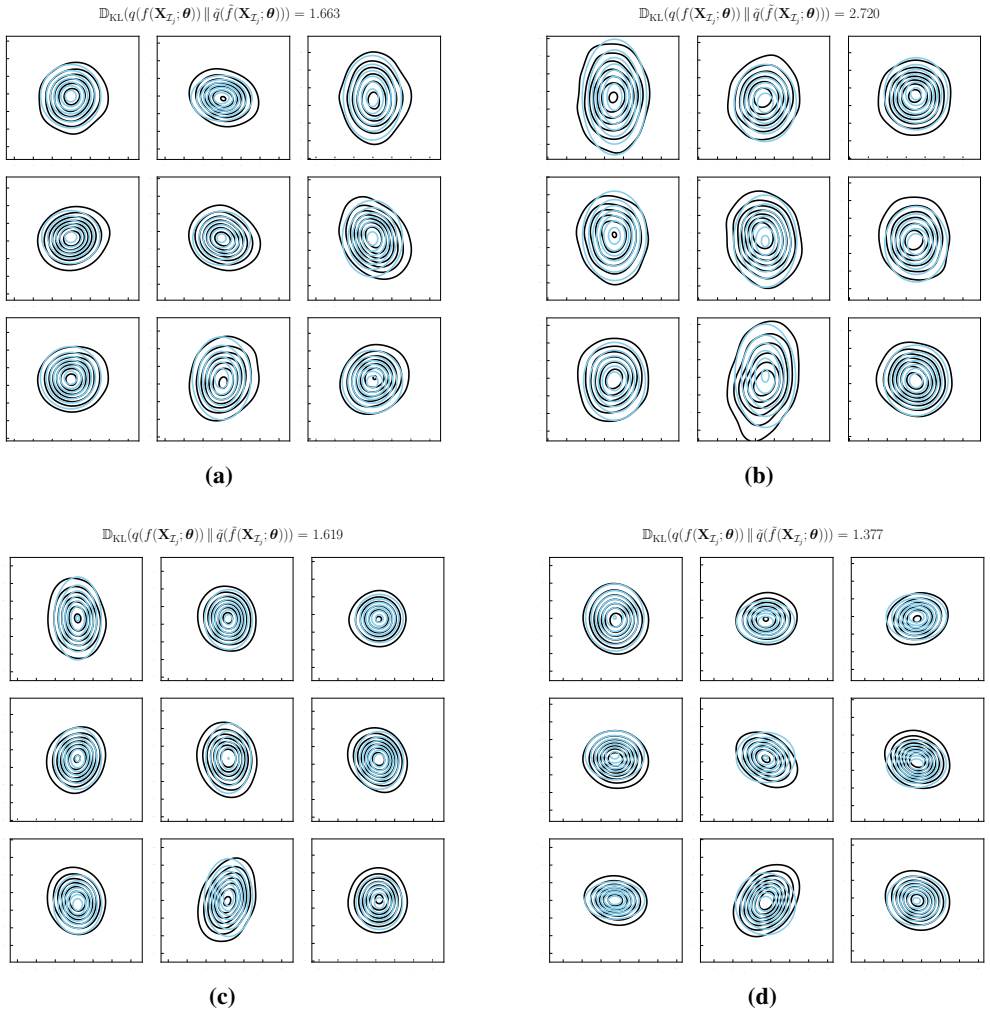


Figure 6. Distributions over functions under linearized and non-linearized mappings for a model trained on the CIFAR-10 dataset. Each plot shows the covariance over stochastic functions (logits) between the first output dimension (corresponding to the first class) and all other output dimensions for a given input point sampled from the test set. The first output dimension is on the x-axis. The BNN's distribution over functions is shown in black, and the distribution over linearized functions is shown in light-blue. The title of each set of plots show the estimated KL divergence from the distribution over non-linearized functions to the distribution over linearized functions.

F. Further Empirical Results

Table 2. Comparison of in- and out-of-distribution performance metrics (mean ± standard error over ten random seeds). Best results are printed in boldface. Deep Ensembles are constructed from six models, all ensembles of BNNs are constructed from three models. For further details about model architectures and training, see Appendix D. *AUROC for binary in- and out-of-distribution detection on MNIST/NotMNIST. §AUROC for binary in- and out-of-distribution detection on SVHN and out-of-distribution accuracy on corrupted CIFAR-10. †AUROC for binary in- and out-of-distribution detection on NotMNIST/FashionMNIST. [1]Implemented using cited paper's code. [2]Values taken from cited paper.

Dataset	Method	Accuracy ↑	NLL ↓	ECE ↓	OOD-AUROC (M/NM)* ↑
FMNIST	MAP	91.73±0.08	0.288±0.003	0.037±0.001	87.00±0.30 / 74.85±1.31
	MFVI (Blundell et al., 2015)	91.03±0.04	0.354±0.003	0.038±0.001	93.10±0.34 / 88.88±0.74
	MFVI (tempered)	91.38±0.05	0.519±0.005	0.058±0.001	86.30±0.29 / 80.78±0.68
	MFVI (radial) (Farquhar et al., 2020a)[1]	90.31±0.11	0.340±0.001	0.035±0.001	84.40±0.68 / 82.11±1.15
	MFVI (five sample)	90.85±0.04	0.298±0.002	0.026±0.000	93.82±0.18 / 86.90±0.38
	MC DROPOUT (Gal & Ghahramani, 2016)	90.55±0.04	**0.230**±0.001	0.012±0.001	88.46±0.57 / 80.02±1.04
	DUQ (van Amersfoort et al., 2020)[2]	**92.40**±0.20	–	–	95.50±0.70 / 94.60±1.80
	BNN-GLM (Immer et al., 2020)[2]	92.25±0.10	0.244±0.003	0.012±0.003	95.55±0.60 / –
	FSVI	91.04±0.17	0.256±0.004	0.011±0.002	**96.02**±0.44 / **95.41**±0.59
	FSVI (MAP init)	90.46±0.06	0.299±0.003	**0.009**±0.001	93.40±0.42 / 92.19±0.39
	SWAG (Maddox et al., 2019)[1]	92.56±0.05	0.300±0.000	0.043±0.001	85.18±0.35 / 80.31±0.30
	Deep Ensemble (Lakshminarayanan et al., 2017)	92.49±0.01	0.242±0.001	**0.019**±0.000	89.22±0.09 / 83.17±0.91
	MC DROPOUT Ensemble	92.30±0.03	0.221±0.000	0.019±0.001	90.17±0.29 / 79.70±0.76
	MFVI Ensemble	92.46±0.04	0.294±0.001	0.026±0.000	94.29±0.21 / 90.31±0.37
	MFVI (tempered) Ensemble	92.21±0.03	0.398±0.002	0.040±0.001	89.46±0.26 / 82.19±0.29
	FSVI Ensemble	**93.34**±0.06	**0.221**±0.001	0.028±0.001	**97.35**±0.16 / **96.86**±0.22
	FSVI (MAP init) Ensemble	92.45±0.05	0.242±0.001	0.019±0.001	96.06±0.23 / 94.89±0.20
Dataset	**Method**	**Accuracy ↑**	**NLL ↓**	**ECE ↓**	**OOD-AUROC/Acc.§ ↑**
CIFAR-10	MAP	87.35±0.09	0.491±0.005	0.070±0.001	90.86±0.43 / 74.20±0.60
	MFVI (Blundell et al., 2015)	84.04±0.07	0.372±0.002	**0.016**±0.001	92.62±0.31 / 71.48±0.74
	MFVI (tempered)	86.29±0.08	0.457±0.003	0.049±0.001	91.54±0.57 / 72.02±0.58
	MFVI (radial) (Farquhar et al., 2020a)[1]	83.99±0.18	0.510±0.001	0.048±0.002	86.04±0.18 / 73.54±0.53
	MC DROPOUT (Gal & Ghahramani, 2016)	83.89±0.18	0.412±0.005	0.018±0.002	92.69±0.49 / 69.75±0.82
	BNN-GLM (Immer et al., 2020)[2,3]	81.37±0.15	0.601±0.008	0.084±0.010	84.30±0.02 / –
	FSVI	86.34±0.11	0.499±0.005	0.061±0.001	94.00±0.39 / 73.59±0.66
	FSVI (MAP init)	**88.10**±0.08	**0.330**±0.003	0.021±0.001	**97.71**±0.11 / **79.64**±0.36
	DUQ (van Amersfoort et al., 2020)[2] (ResNet)	94.10±0.2	–	–	92.70±1.30 / –
	VOGN (Osawa et al., 2019)[2] (ResNet)	84.27±0.20	0.477±0.006	0.040±0.002	87.60±0.20 / –
	FSVI (MAP init) (ResNet)	**94.10**±0.04	**0.175**±0.002	0.014±0.001	**98.89**±0.23 / **81.38**±0.41
	SWAG (Maddox et al., 2019)[1]	89.73±0.14	0.480±0.001	0.067±0.002	89.79±0.50 / 76.12±0.51
	Deep Ensemble (Lakshminarayanan et al., 2017)	89.28±0.04	0.339±0.003	0.020±0.001	92.00±0.16 / 76.65±0.21
	MC DROPOUT Ensemble	88.02±0.09	0.371±0.002	0.056±0.001	91.92±0.14 / 72.89±0.57
	MFVI Ensemble	89.49±0.07	0.330±0.001	0.049±0.001	94.06±0.20 / 73.67±0.38
	MFVI (tempered) Ensemble	89.78±0.04	0.321±0.001	0.014±0.001	92.07±0.40 / 75.07±0.26
	FSVI Ensemble	90.17±0.03	0.314 ±0.001	0.018±0.001	96.17±0.10 / 78.63±0.40
	FSVI (MAP init) Ensemble	**90.29**±0.09	**0.296**±0.002	**0.009**±0.001	**98.21**±0.06 / **81.02**±0.28
Dataset	**Method**	**Accuracy ↑**	**NLL ↓**	**ECE ↓**	**OOD-AUROC (NM/FM)† ↑**
MNIST	MAP	97.84±0.04	0.069±0.001	0.003±0.000	88.13±1.02 / 94.90±0.57
	MFVI (Blundell et al., 2015)	97.28±0.03	0.060±0.001	0.006±0.000	93.74±0.91 / **96.73**±0.24
	MFVI (tempered)	97.74±0.03	0.088±0.002	0.011±0.000	91.00±1.08 / 93.67±0.45
	MC DROPOUT (Gal & Ghahramani, 2016)	97.48±0.04	0.068±0.001	0.010±0.001	88.63±1.25 / 96.11±0.17
	FSVI	97.47±0.11	0.087±0.003	0.009±0.000	**98.99**±0.33 / 96.67±0.33
	Deep Ensemble (Lakshminarayanan et al., 2017)	98.41±0.01	0.054±0.000	0.011±0.000	95.02±0.20 / 97.00±0.13
	MC DROPOUT Ensemble	98.32±0.03	0.061±0.001	0.016±0.000	93.34±0.37 / 96.88±0.07
	MFVI Ensemble	98.46±0.02	0.052±0.000	0.007±0.000	97.27±0.23 / 96.50±0.17
	MFVI (tempered) Ensemble	98.32±0.02	0.056±0.001	0.003±0.000	96.39±0.29 / 95.82±0.12
	FSVI Ensemble	98.16±0.01	0.060±0.001	0.004±0.000	**99.71**±0.04 / **97.91**±0.10

F.1. Uncertainty Desiderata & Rotated MNIST

In order for a model to be considered reliable, we would like it (i) to exhibit low predictive uncertainty on training data and high predictive uncertainty on out-of-distribution inputs, (ii) to have the ability to use its predictive uncertainty estimates to distinguish in- from out-of-distribution data, and (iii) if possible, maintain high predictive accuracy even under distribution shift (Ovadia et al., 2019). A model that satisfies all of these desiderata would be able to alert a human in the loop or be referred to a domain expert when it encounters data points on which it has particularly high predictive uncertainty. This ability increases a model's level of robustness to making poor but confident predictions and is particularly relevant in safety-critical settings, such as medical diagnosis.

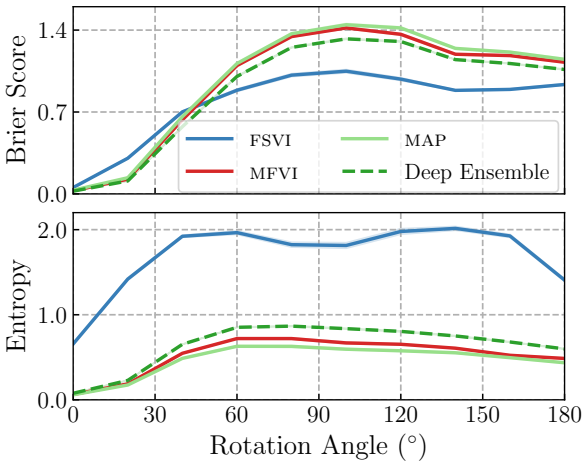


Figure 7. Predictive uncertainty and accuracy on rotated MNIST. Models with reliable uncertainty estimates would exhibit higher predictive uncertainty the more the digits are rotated. Ideally, such models would maintain high predictive accuracy (low Brier score).

To illustrate these desiderata, we consider the rotated MNIST task (Ovadia et al., 2019) where the goal is to maintain a high level of predictive accuracy (measured in terms of Brier scores, which is more sensitive to poorly calibrated predictions than a model's accuracy) while exhibiting high predictive uncertainty. Figure 7 shows Brier scores (lower is better) and predictive entropy (higher means more uncertain) of four different models. Rotating the MNIST digits gradually shifts the data distributions, we would expect Brier scores to increase (worse predictive accuracy) on the increasingly rotated digits. A good model with reliable predictive entropy estimates would only experience a small decrease under distribution shift while exhibiting a large increase in predictive uncertainty. As can be seen in the plot, the Brier scores of FSVI decreases the least, while FSVI's uncertainty is significantly higher than other models'.

F.2. Downstream Tasks: Continual Learning & Expert Referral

We consider two important downstream tasks: expert referral and continual learning.

For uncertainty-based expert referral, the goal is to use a model's predictive uncertainty to identify data samples from a pool or in- and out-of-distribution samples to refer to an expert for review in order to maximize the predictive accuracy on the remaining samples. This type of downstream task is important in safety-critical settings such as medical diagnostics. Figure 2c shows that FSVI outperforms related methods in selecting data points to maximize predictive accuracy on the remaining samples.

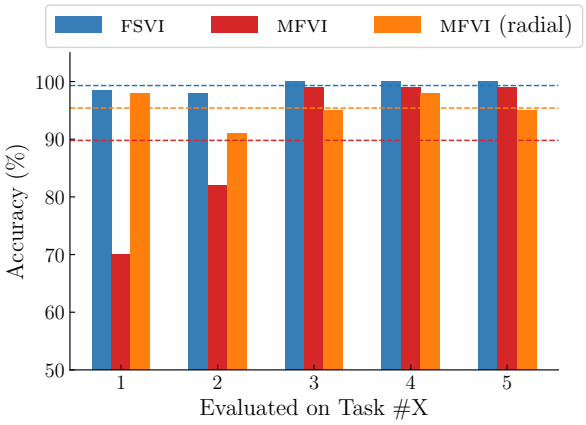
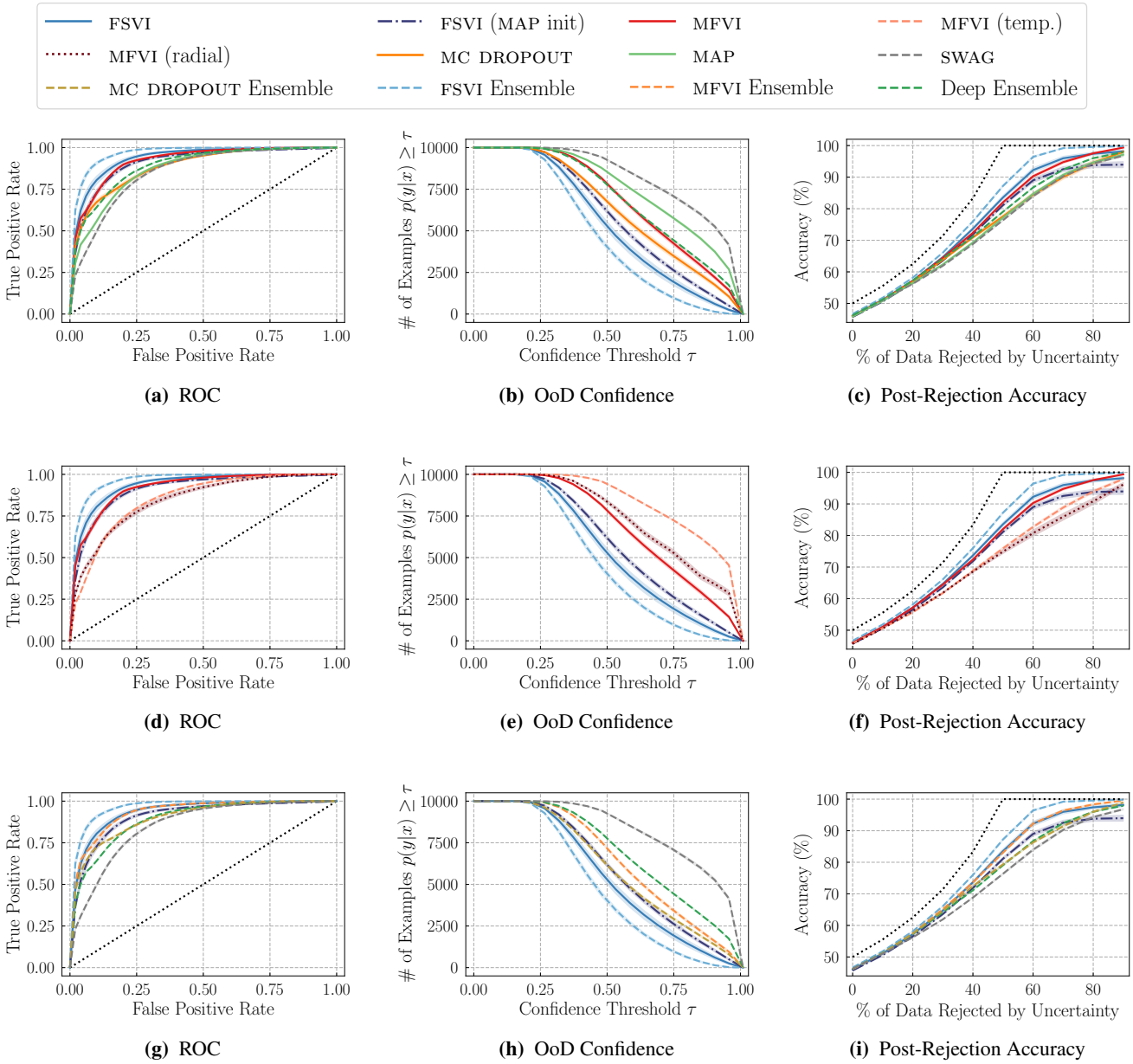


Figure 8. Predictive accuracy on five different tasks. Models are trained on five FashionMNIST tasks in sequence using the posterior distribution over functions from the previous task for FSVI and the posterior distribution over parameters from the previous task for MFVI. Both parameter-space inference methods use variational continual learning (VCL; (Nguyen et al., 2018)).

Continual learning is the problem of learning several tasks sequentially while discarding the dataset after each task (Kirkpatrick et al., 2017). To do this effectively and not forget previous tasks, it is crucial to incorporate prior information into learning. We follow Farquhar et al. (2020a) and compare our method to a widely used parameter-space method for continual learning, variational continual learning (VCL; (Nguyen et al., 2018)), combined with MFVI and MFVI (radial), respectively. As can be seen in Figure 8, FSVI performs significantly better at incorporating prior information learned on previous tasks as evidenced by the consistently high accuracy even on tasks learned in the past (here: tasks 1–4). A full description of the experiment setup can be found in Appendix D.

F.3. Out-of-Distribution Performance FashionMNIST/MNIST



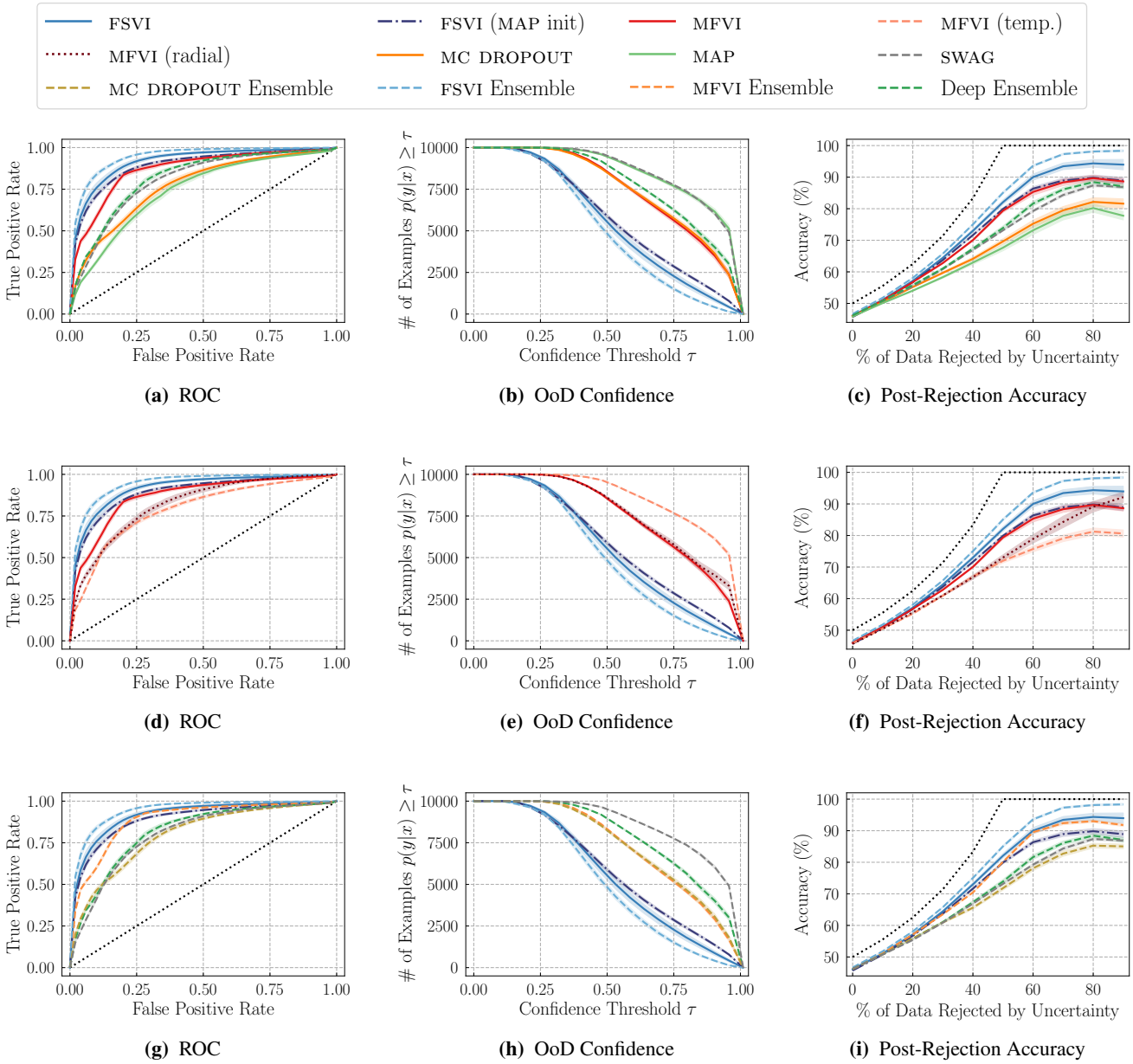
(a) ROC **(b)** OoD Confidence **(c)** Post-Rejection Accuracy

(d) ROC **(e)** OoD Confidence **(f)** Post-Rejection Accuracy

(g) ROC **(h)** OoD Confidence **(i)** Post-Rejection Accuracy

Figure 9. Uncertainty evaluation metrics for out-of-distribution prediction. Models were trained on FashionMNIST and MNIST is used as out-of-distribution data. Shading denotes the standard error, computed over ten random seeds. **Left**: Receiver operating characteristic for out-of-distribution detection. Curves closer to the top left are better. **Center**: Model confidence on out-of-distribution inputs. Curves closer to the bottom left are better. **Right**: Accuracy after rejecting X% of evaluation samples with the highest predictive uncertainty. Curves closer to the theoretical maximum (denoted by the dotted line) are better. The figures show that FSVI consistently outperforms related methods in terms of classifying in- and out-of-distribution datapoints (left), generating low-confidence predictions on out-of-distribution data (center), and using predictive uncertainty to identify points where the model's predictions would be incorrect (right).

F.4. Out-of-Distribution Performance FashionMNIST/NotMNIST



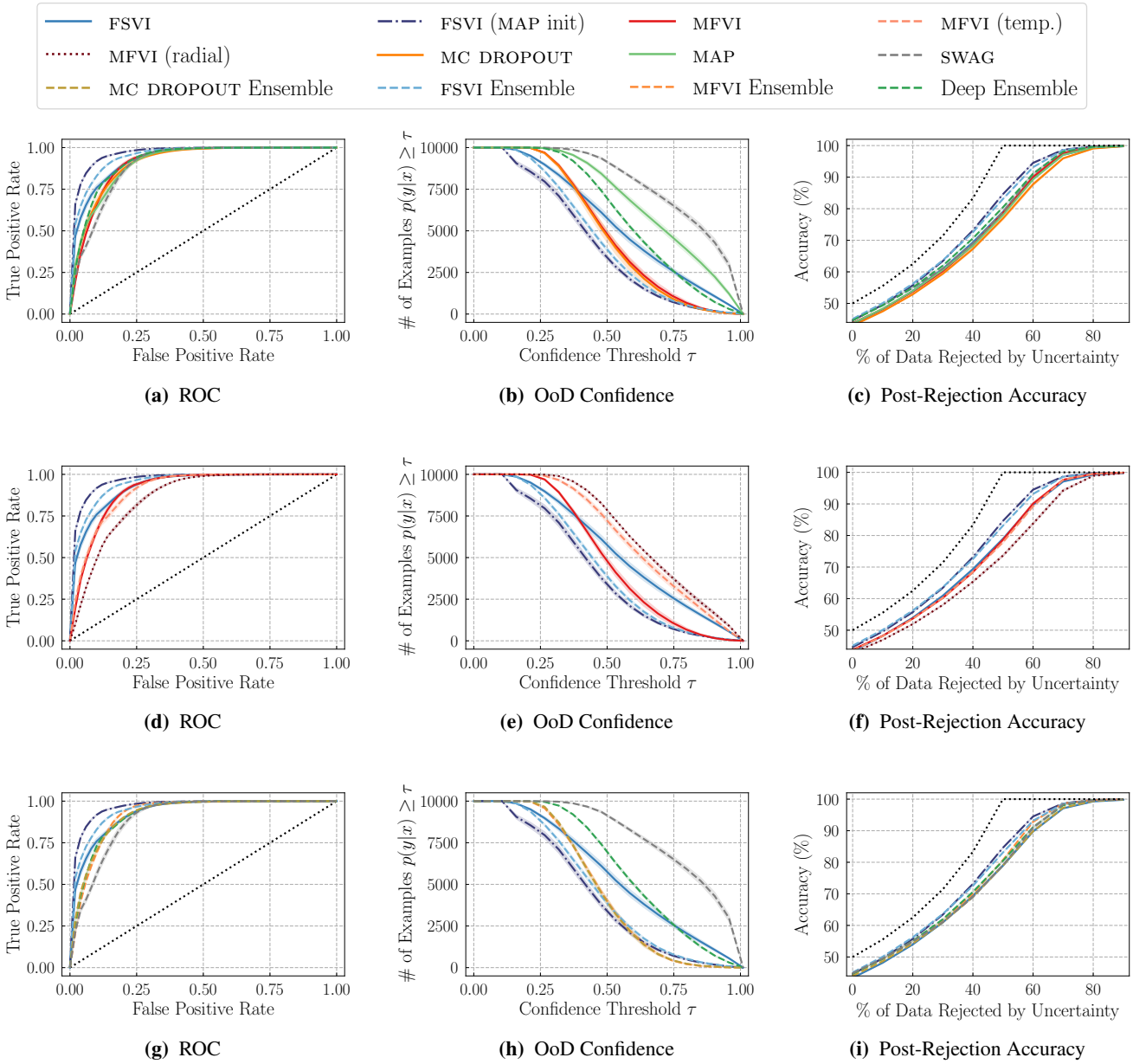
(a) ROC **(b)** OoD Confidence **(c)** Post-Rejection Accuracy

(d) ROC **(e)** OoD Confidence **(f)** Post-Rejection Accuracy

(g) ROC **(h)** OoD Confidence **(i)** Post-Rejection Accuracy

Figure 10. Uncertainty evaluation metrics for out-of-distribution prediction. Models were trained on FashionMNIST and NotMNIST is used as out-of-distribution data. Shading denotes the standard error, computed over ten random seeds. **Left**: Receiver operating characteristic for out-of-distribution detection. Curves closer to the top left are better. **Center**: Model confidence on out-of-distribution inputs. Curves closer to the bottom left are better. **Right**: Accuracy after rejecting X% of evaluation samples with the highest predictive uncertainty. Curves closer to the theoretical maximum (denoted by the dotted line) are better. The figures show that FSVI consistently outperforms related methods in terms of classifying in- and out-of-distribution datapoints (left), generating low-confidence predictions on out-of-distribution data (center), and using predictive uncertainty to identify points where the model's predictions would be incorrect (right).

F.5. Out-of-Distribution Performance CIFAR-10/SVHN



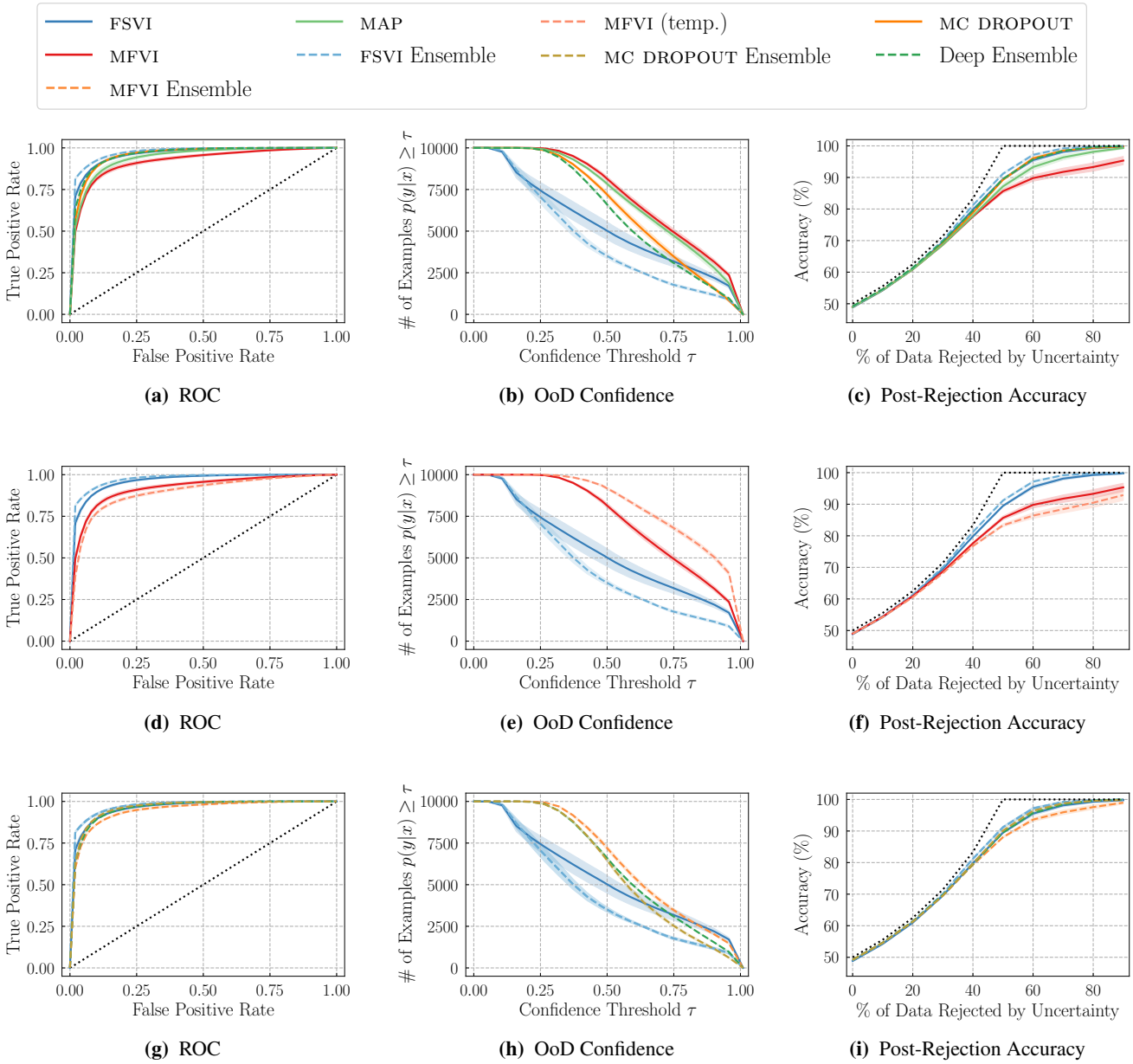
(a) ROC **(b)** OoD Confidence **(c)** Post-Rejection Accuracy

(d) ROC **(e)** OoD Confidence **(f)** Post-Rejection Accuracy

(g) ROC **(h)** OoD Confidence **(i)** Post-Rejection Accuracy

Figure 11. Uncertainty evaluation metrics for out-of-distribution prediction. Models were trained on CIFAR-10 and SVHN is used as out-of-distribution data. Shading denotes the standard error, computed over ten random seeds. **Left**: Receiver operating characteristic for out-of-distribution detection. Curves closer to the top left are better. **Center**: Model confidence on out-of-distribution inputs. Curves closer to the bottom left are better. **Right**: Accuracy after rejecting X% of evaluation samples with the highest predictive uncertainty. Curves closer to the theoretical maximum (denoted by the dotted line) are better. The figures show that FSVI consistently outperforms related methods in terms of classifying in- and out-of-distribution datapoints (left), generating low-confidence predictions on out-of-distribution data (center), and using predictive uncertainty to identify points where the model's predictions would be incorrect (right).

F.6. Out-of-Distribution Performance MNIST/FashionMNIST



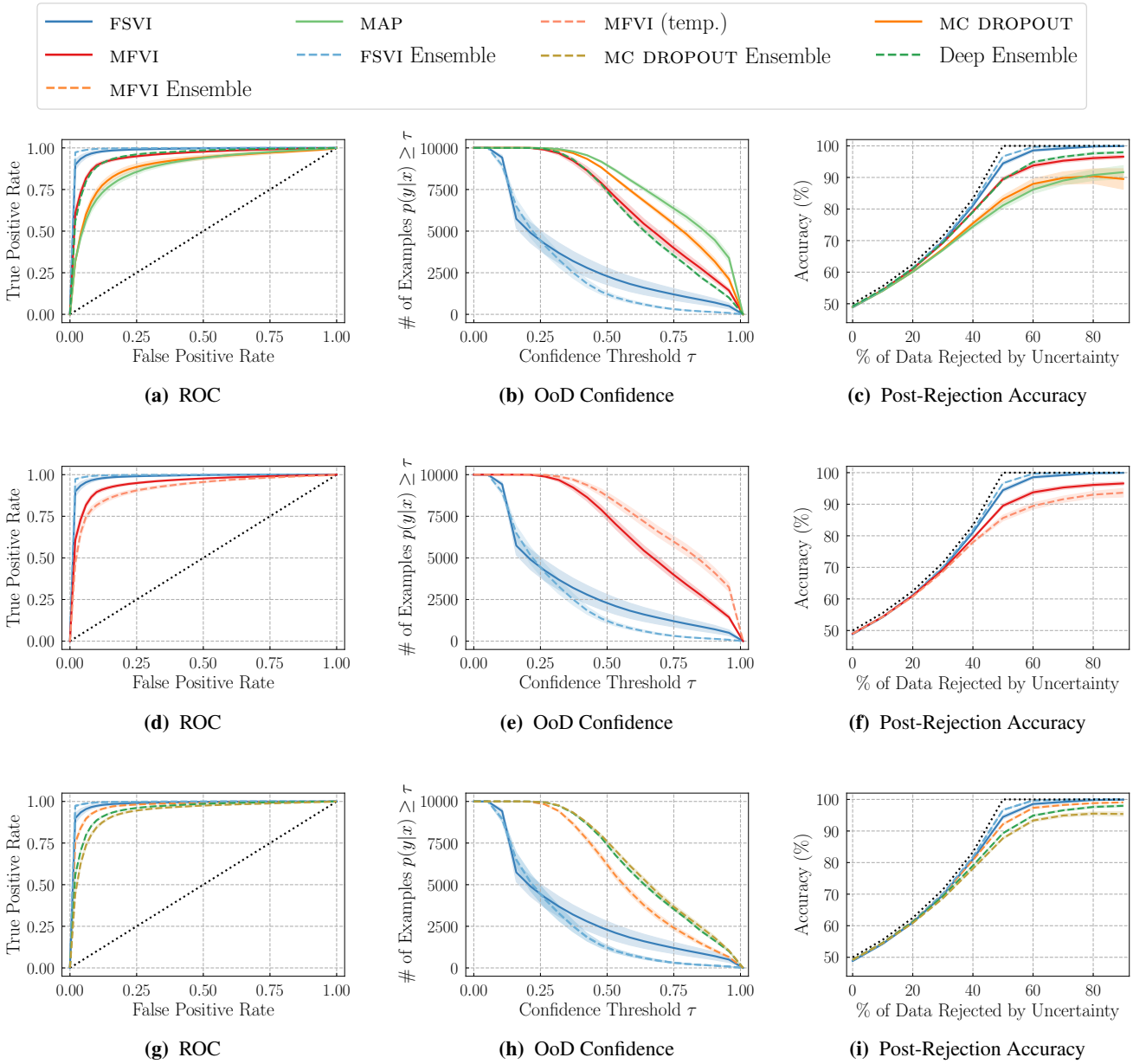
(a) ROC **(b)** OoD Confidence **(c)** Post-Rejection Accuracy

(d) ROC **(e)** OoD Confidence **(f)** Post-Rejection Accuracy

(g) ROC **(h)** OoD Confidence **(i)** Post-Rejection Accuracy

Figure 12. Uncertainty evaluation metrics for out-of-distribution prediction. Models were trained on MNIST and FashionMNIST is used as out-of-distribution data. Shading denotes the standard error, computed over ten random seeds. **Left**: Receiver operating characteristic for out-of-distribution detection. Curves closer to the top left are better. **Center**: Model confidence on out-of-distribution inputs. Curves closer to the bottom left are better. **Right**: Accuracy after rejecting X% of evaluation samples with the highest predictive uncertainty. Curves closer to the theoretical maximum (denoted by the dotted line) are better. The figures show that FSVI consistently outperforms related methods in terms of classifying in- and out-of-distribution datapoints (left), generating low-confidence predictions on out-of-distribution data (center), and using predictive uncertainty to identify points where the model's predictions would be incorrect (right).

F.7. Out-of-Distribution Performance MNIST/NotMNIST



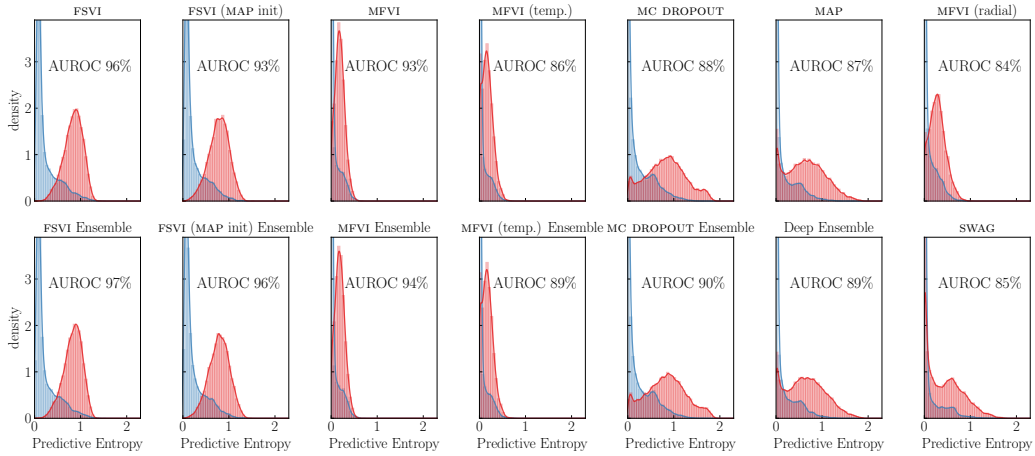
(a) ROC **(b)** OoD Confidence **(c)** Post-Rejection Accuracy

(d) ROC **(e)** OoD Confidence **(f)** Post-Rejection Accuracy

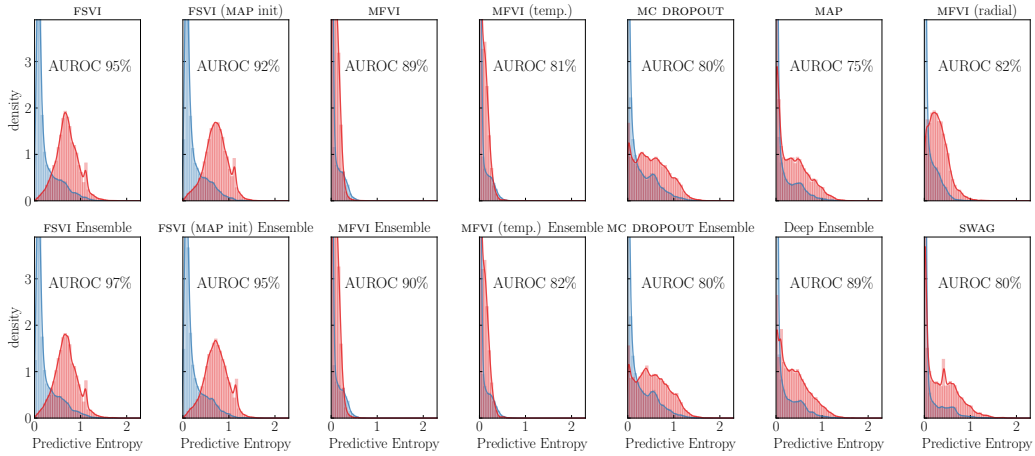
(g) ROC **(h)** OoD Confidence **(i)** Post-Rejection Accuracy

Figure 13. Uncertainty evaluation metrics for out-of-distribution prediction. Models were trained on MNIST and NotMNIST is used as out-of-distribution data. Shading denotes the standard error, computed over ten random seeds. **Left**: Receiver operating characteristic for out-of-distribution detection. Curves closer to the top left are better. **Center**: Model confidence on out-of-distribution inputs. Curves closer to the bottom left are better. **Right**: Accuracy after rejecting X% of evaluation samples with the highest predictive uncertainty. Curves closer to the theoretical maximum (denoted by the dotted line) are better. The figures show that FSVI consistently outperforms related methods in terms of classifying in- and out-of-distribution datapoints (left), generating low-confidence predictions on out-of-distribution data (center), and using predictive uncertainty to identify points where the model's predictions would be incorrect (right).

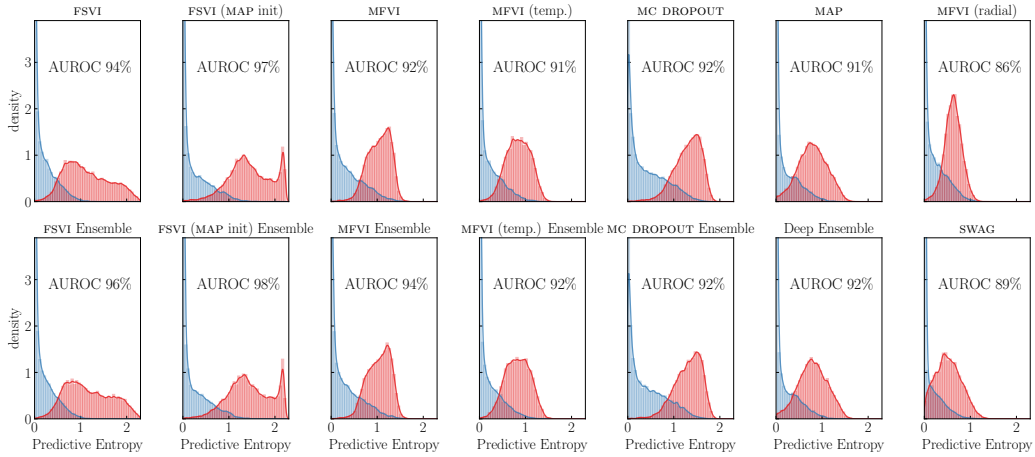
F.8. Predictive Entropy on In- and Out-of-Distribution Inputs



(a) In-Distribution: FashionMNIST, **Out-of-Distribution**: MNIST

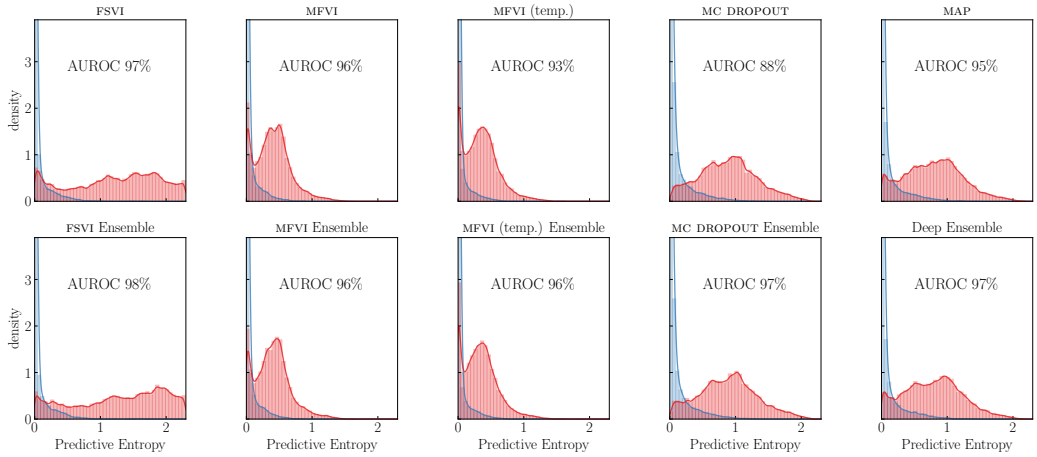


(b) In-Distribution: FashionMNIST, **Out-of-Distribution**: NotMNIST

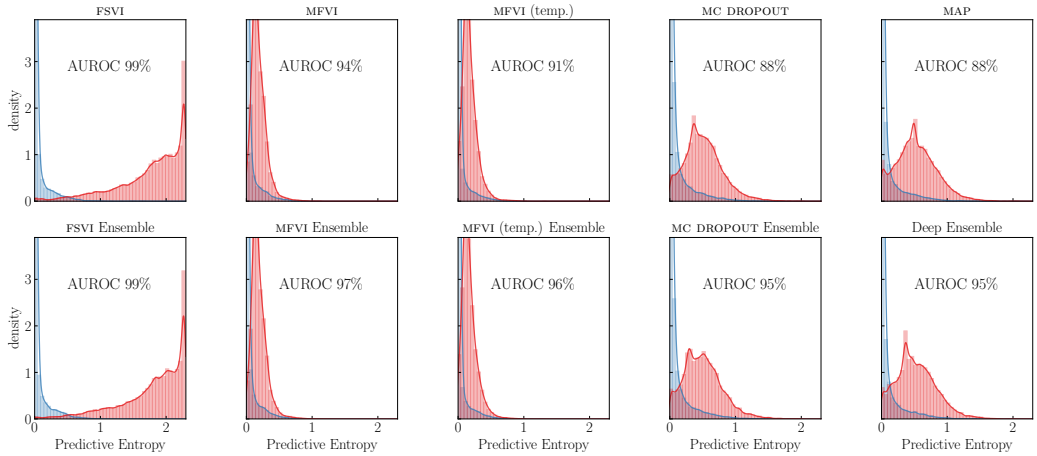


(c) In-Distribution:CIFAR-10, **Out-of-Distribution**: SVHN

Figure 14. Histograms of predictive entropy estimates. Low predictive entropy corresponds to low predictive uncertainty and high predictive entropy corresponds to high predictive uncertainty. Models with reliable predictive uncertainty estimates would exhibit low predictive entropy on in-distribution inputs (i.e., a predictive entropy distribution concentrated near zero) and high predictive entropy on out-of-distribution inputs (i.e., a predictive entropy distribution concentrated away from zero) with little overlap between the distributions.



(a) In-Distribution: MNIST, **Out-of-Distribution**: FashionMNIST

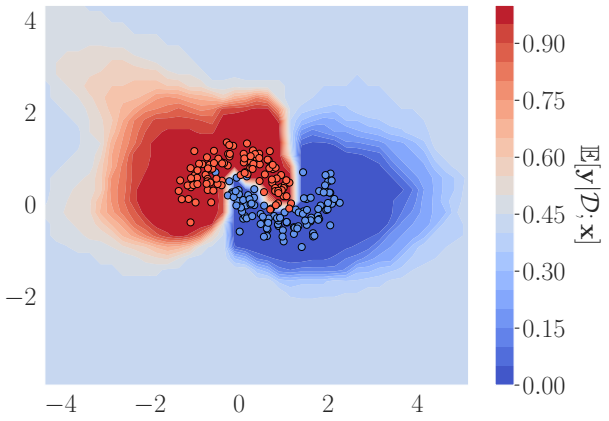


(b) In-Distribution: MNIST, **Out-of-Distribution**: NotMNIST

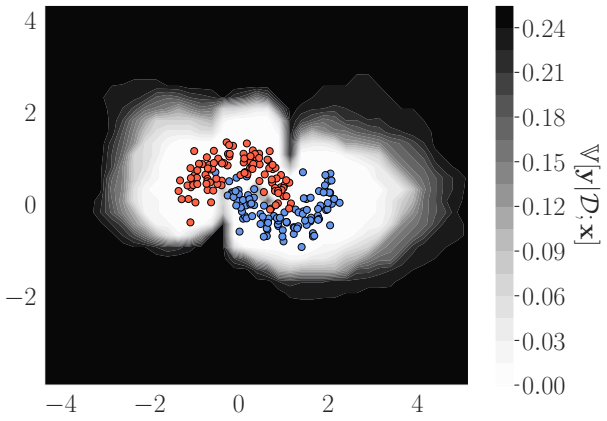
Figure 15. Histograms of predictive entropy estimates. Low predictive entropy corresponds to low predictive uncertainty and high predictive entropy corresponds to high predictive uncertainty. Models with reliable predictive uncertainty estimates would exhibit low predictive entropy on in-distribution inputs (i.e., a predictive entropy distribution concentrated near zero) and high predictive entropy on out-of-distribution inputs (i.e., a predictive entropy distribution concentrated away from zero) with little overlap between the distributions.

G. Illustrative Examples

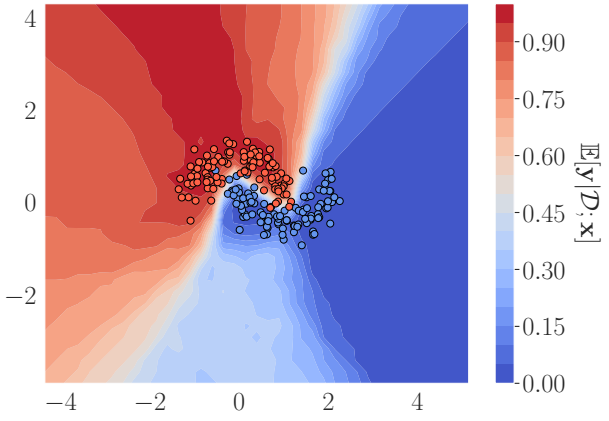
G.1. *Two Moons* Classification Dataset



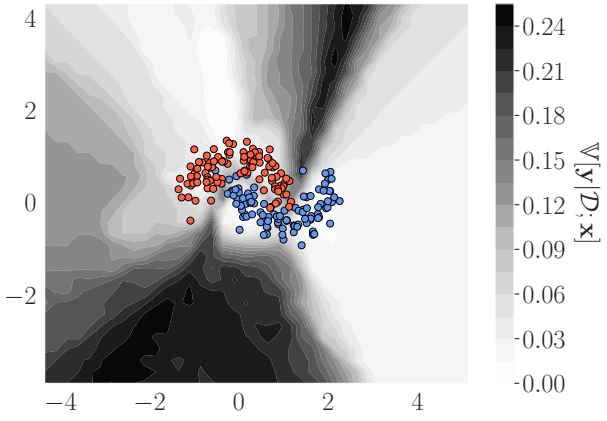
(a) FSVI: Posterior Predictive Mean



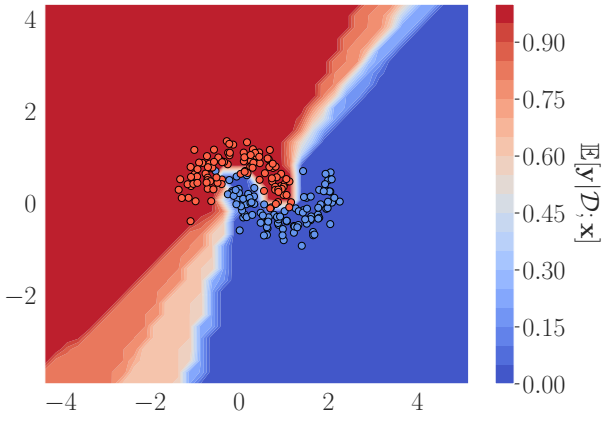
(b) FSVI: Posterior Predictive Variance



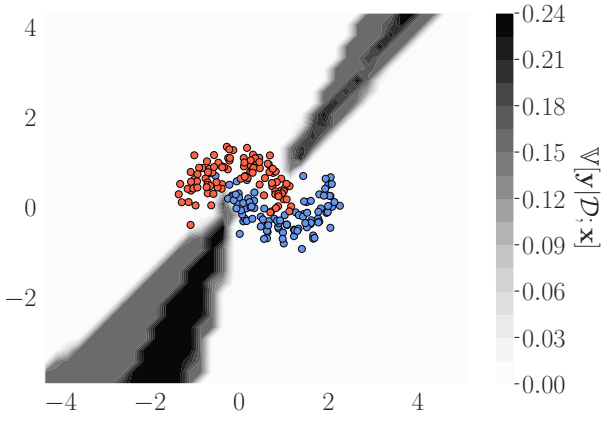
(c) MFVI: Posterior Predictive Mean



(d) MFVI: Posterior Predictive Variance



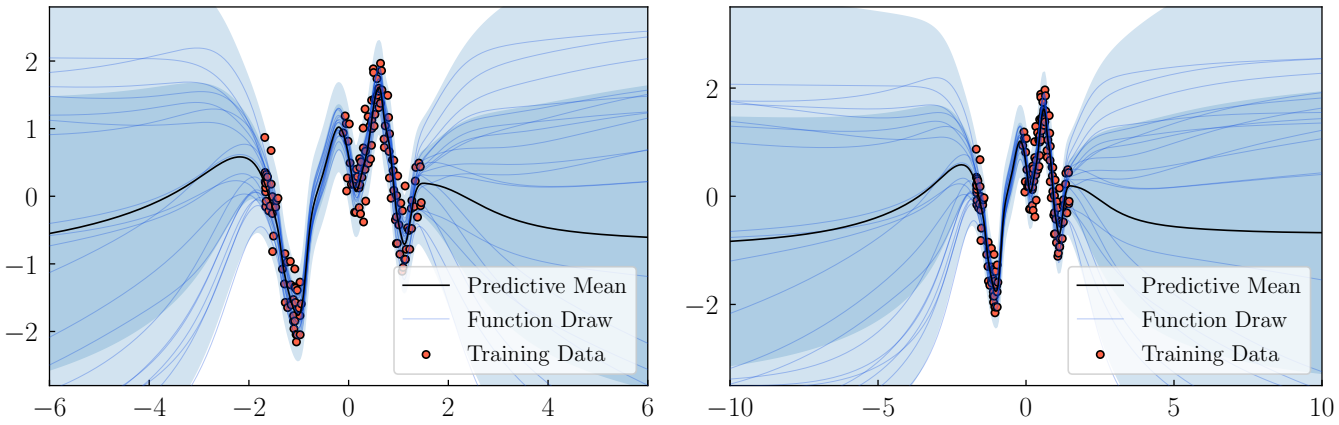
(e) MAP Ensemble: Predictive Mean



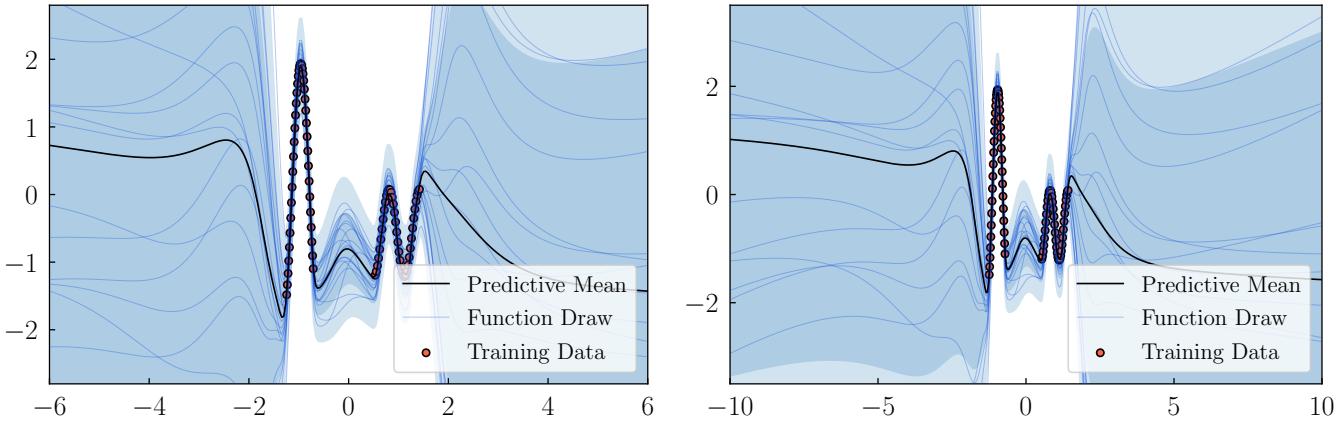
(f) MAP Ensemble: Predictive Variance

Figure 16. Binary classification on the *Two Moons* dataset. The plots show the posterior predictive mean and variance of a BNN trained via FSVI (Figure 16a and Figure 16b), of a BNN trained via MFVI (Figure 16c and Figure 16d), and an ensemble of MAP models (Figure 16e and Figure 16f). The predictive means represent the expected class probabilities and the predictive variance the model's epistemic uncertainty over the class probabilities. With FSVI, the predictive distribution is able to faithfully capture the geometry of the data manifold and exhibits high uncertainty over the class probabilities in areas of the data space of which the data is not informative. In contrast, neither MFVI, nor MAP ensembles are unable to accurately capture the geometry of the data manifold only exhibit high uncertainty around the decision boundary.

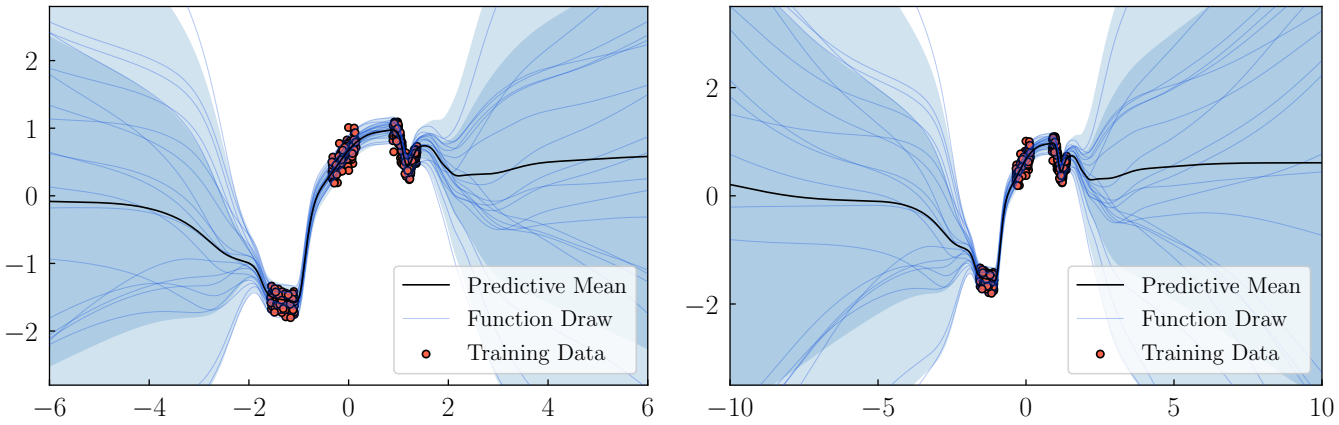
G.2. Illustrative Examples: 1D Regression



(a) "Snelson" Dataset (Snelson & Ghahramani (2006))



(b) "OAT-1D" Dataset (van Amersfoort et al. (2021))



(c) "Subspace Inference" Dataset (Izmailov et al. (2020))

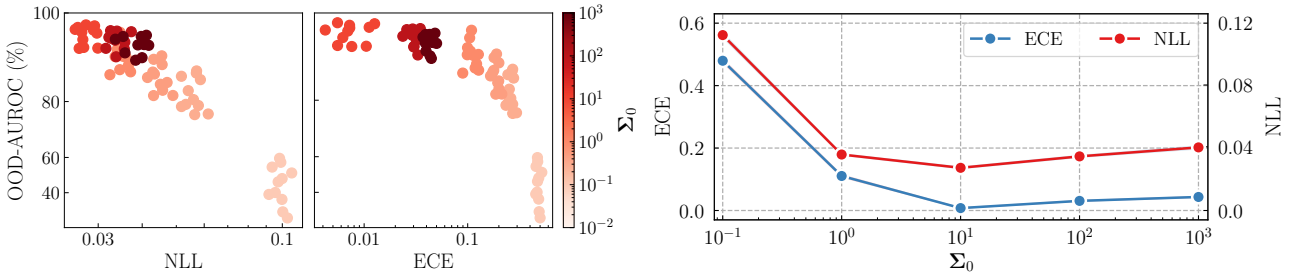
Figure 17. 1D Regression with FSVI on a selection of datasets used to demonstrate desirable predictive uncertainty estimates in prior works. The left column is zoomed in.

H. Ablation Studies

We performed a comprehensive set of ablation studies to understand how different model and variational parameters affect the resulting predictive distributions. Specifically, we used a validation holdout set (10% of the training data) to assess the effect of the prior covariance, the number of inducing samples used per gradient steps, the inducing input selection method, and the number of Monte Carlo samples for evaluating the expected log-likelihood on in- and out-of-distribution performance metrics. All results below are obtained from ten random seeds.

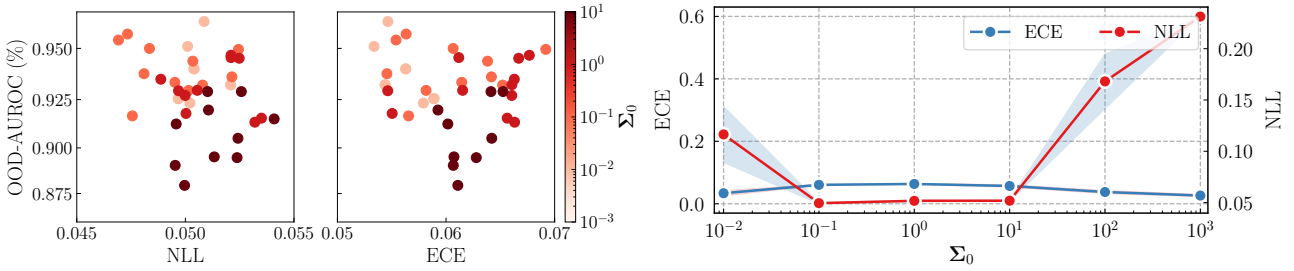
H.1. Ablation Study on the Effect of Different Function-Space Priors

To better understand FSVI, we investigate what hyperparameter choice leads to good predictive performance and reliable out-of-distribution predictive uncertainty. Figure 18 and Figure 19 show plots that demonstrate the link between different choices of prior variance and the resulting OOD-AUROC, test ECE, and test log-likelihood for FashionMNIST. As can be seen in Figure 18b, test ECE is lowest for a prior variance of $\mathbf{\Sigma}_0 = 10$, while Figure 18a show that OOD-AUROC, test ECE, and test log-likelihood are highly negatively correlated. This insight is useful, since it means that in real-world settings, where there are no real out-of-distribution validation sets, one can choose the prior variance that minimize test ECE and test negative log-likelihood. We follow this approach to select the hyperparameters for FSVI in Table 2. For further ablations and details on hyperparameter selection, see Appendix F.



(a) Effect of Prior Variance $\mathbf{\Sigma}_0$ **(b)** ECE and NLL vs. $\mathbf{\Sigma}_0$

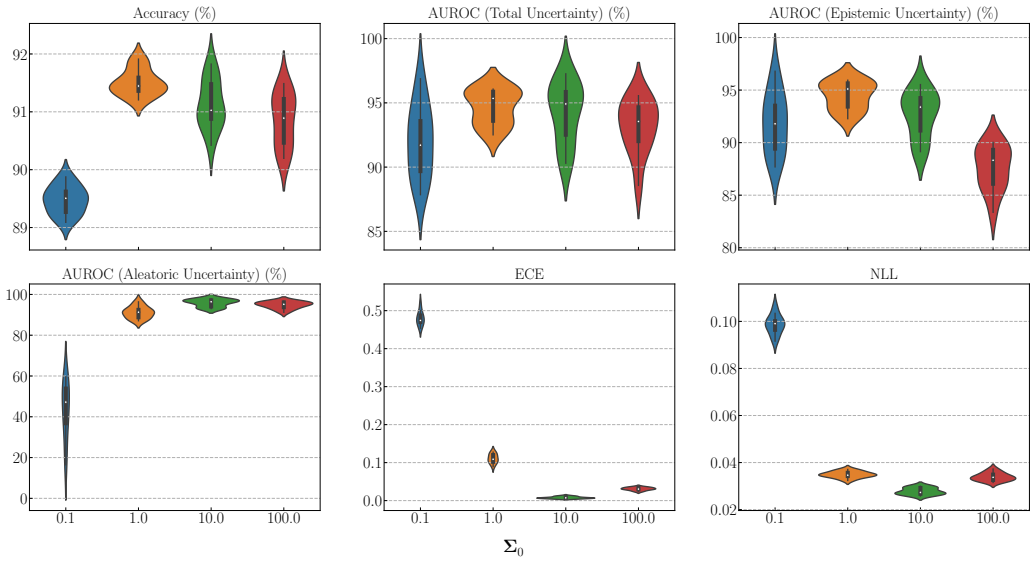
Figure 18. For BNNs trained via FSVI on FashionMNIST, the figures show the relationship between OOD-AUROC (on SVHN), the negative log-likelihood (NLL), expected calibration error (ECE), and the prior variance. OOD-AUROC is very negatively correlated with both NLL and ECE. All metrics are optimal at a prior covariance of $\mathbf{\Sigma}_0 = 10$. For further results, see Appendix F.



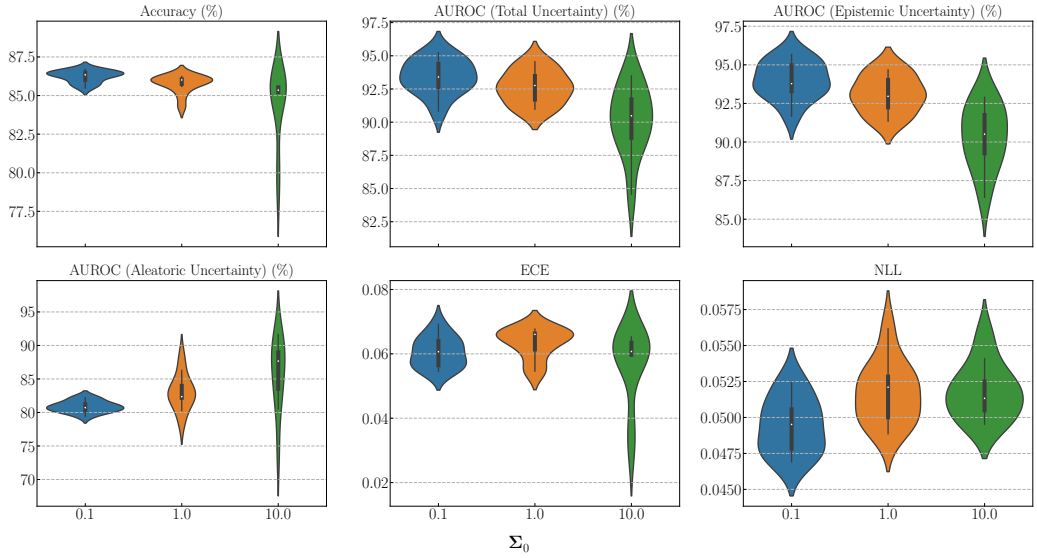
(a) Effect of Prior Variance $\mathbf{\Sigma}_0$ **(b)** ECE and NLL vs. $\mathbf{\Sigma}_0$

Figure 19. For BNNs trained via FSVI on CIFAR-10, the figures show the relationship between OOD-AUROC (on SVHN), the negative log-likelihood (NLL), expected calibration error (ECE), and the prior variance. OOD-AUROC is very negatively correlated with both NLL and ECE. All metrics are optimal at a prior covariance of $\mathbf{\Sigma}_0 = 10$. For further results, see Appendix F.

H.2. Prior Variance



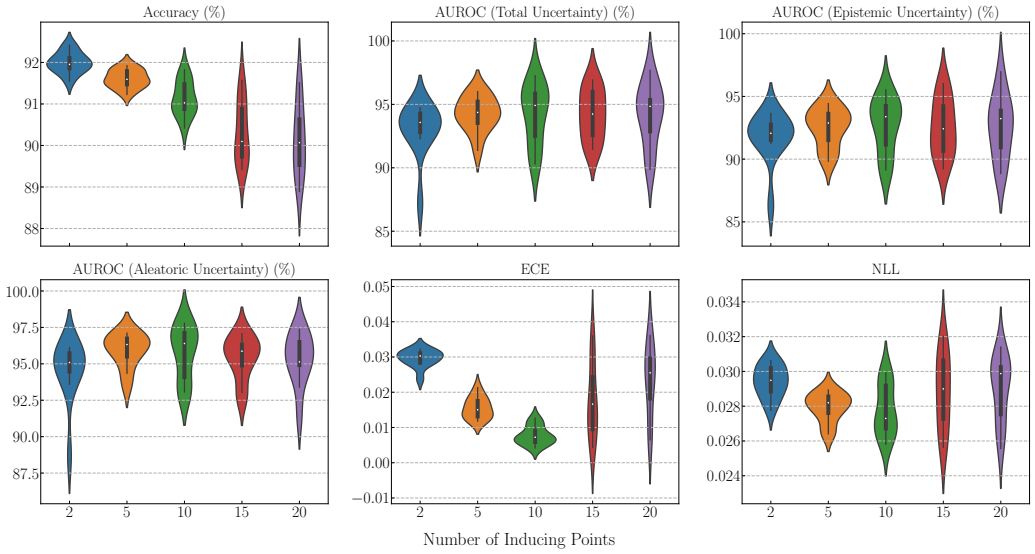
(a) FashionMNIST



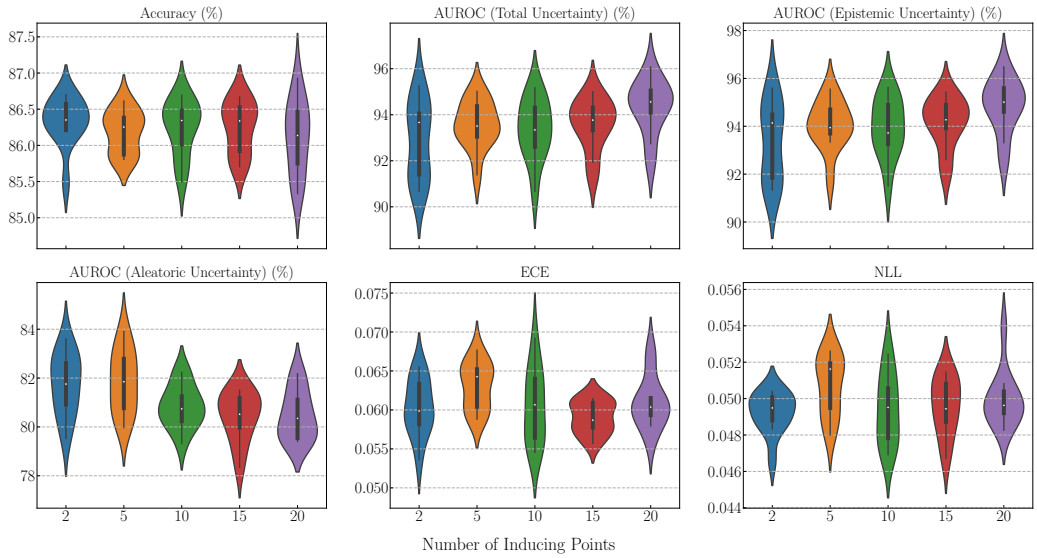
(b) CIFAR-10

Figure 20. Effect of prior variance over parameters $\mathbf{\Sigma}_0$ on in- and out-of-distribution evaluation metrics. For all experiments, 10 inducing inputs were sampled for each gradient step. For FashionMNIST, a fraction of 50% of inducing inputs were sampled from the training set, while for CIFAR-10 no inducing inputs were sampled from the training set. 5 Monte Carlo samples were used to evaluate the expected log-likelihood. Out-of-distribution detection is performed on MNIST for models trained on FashionMNIST and on SVHN for models trained on CIFAR-10.

H.3. Number of Inducing Inputs



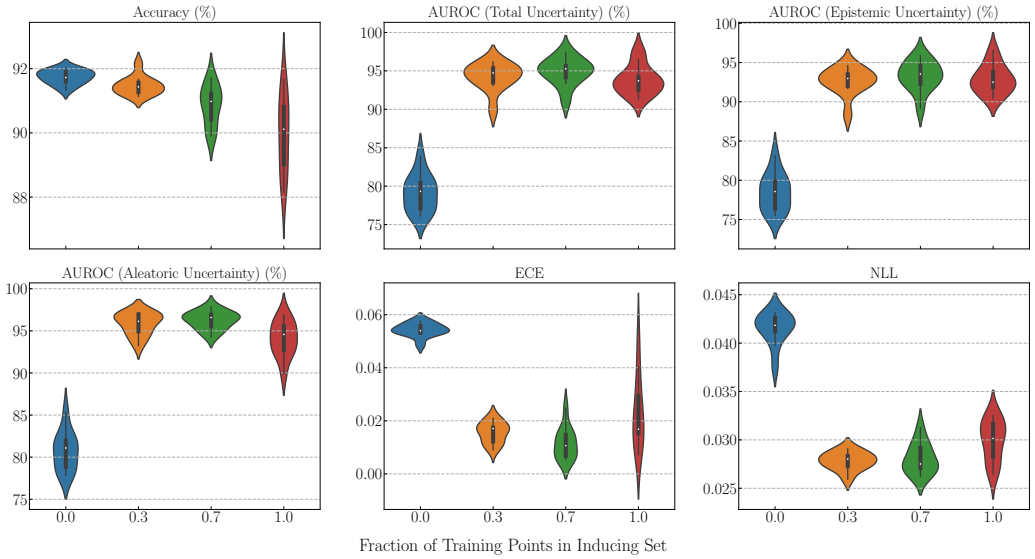
(a) FashionMNIST



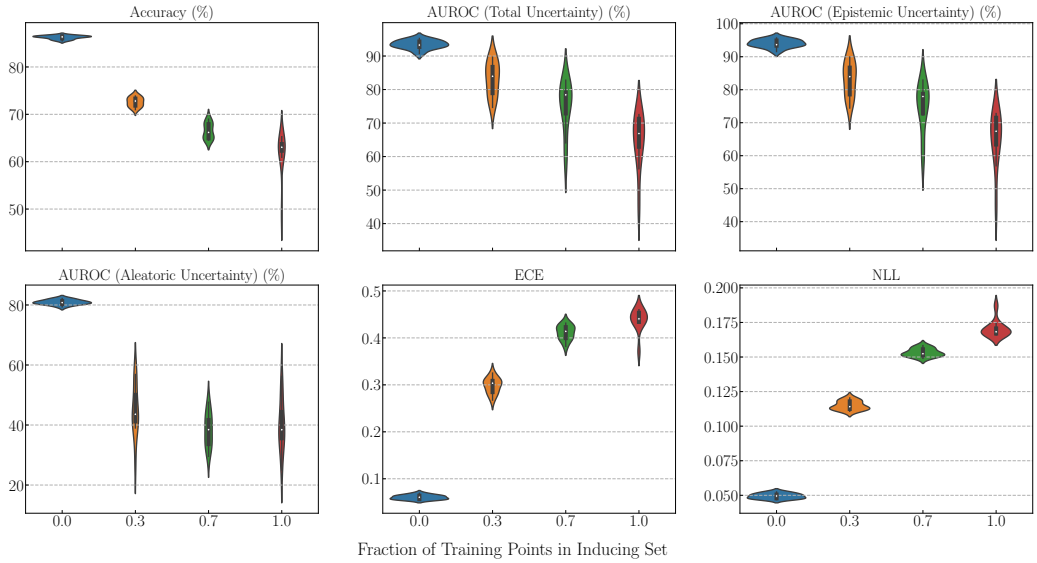
(b) CIFAR-10

Figure 21. Effect of the number of inducing points used to evaluate the KL divergence at each gradient step on in- and out-of-distribution evaluation metrics. For FashionMNIST, a fraction of 50% of inducing inputs were sampled from the training set, while for CIFAR-10 no inducing inputs were sampled from the training set. A prior variance of 10 was used for FashionMNIST and a prior variance of 0.1 was used for CIFAR-10. 5 Monte Carlo samples were used to evaluate the expected log-likelihood. Out-of-distribution detection is performed on MNIST for models trained on FashionMNIST and on SVHN for models trained on CIFAR-10.

H.4. Inducing Input Sampling Method



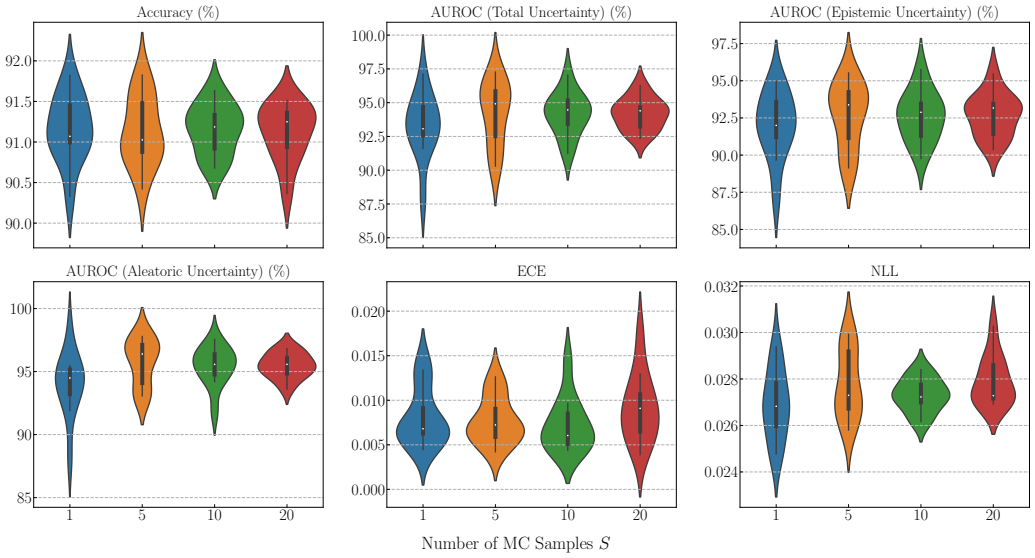
(a) FashionMNIST



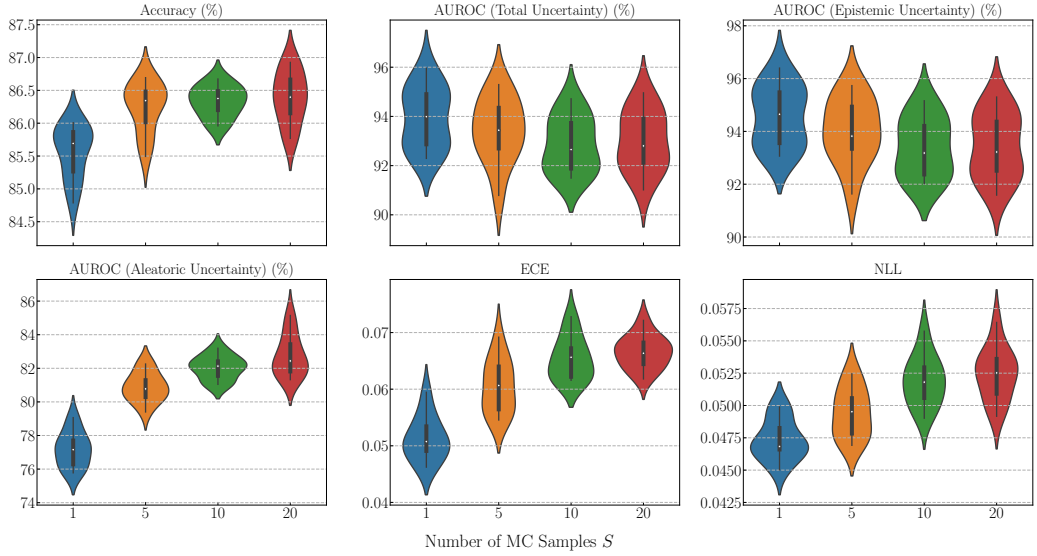
(b) CIFAR-10

Figure 22. Effect of the inducing input sampling method on in- and out-of-distribution evaluation metrics. The x-axis shows the fraction of the inducing inputs used for each gradient step that was sampled from the training batch. "0.0" means that no inducing inputs were sampled from the training data, and "1.0" means that all inducing inputs were sampled from the training set. For all experiments, 10 inducing inputs were sampled for each gradient step. A prior variance of 10 was used for FashionMNIST and a prior variance of 1.0 was used for CIFAR-10. 5 Monte Carlo samples were used to evaluate the expected log-likelihood. Out-of-distribution detection is performed on MNIST for models trained on FashionMNIST and on SVHN for models trained on CIFAR-10.

H.5. Number of Monte Carlo Samples



(a) FashionMNIST



(b) CIFAR-10

Figure 23. Effect of the number of Monte Carlo samples used to evaluate the expected log-likelihood in the variational objective on in- and out-of-distribution evaluation metrics. The x-axis shows the number of Monte Carlo samples used. For all experiments, 10 inducing inputs were sampled for each gradient step. A prior variance of 10 was used for FashionMNIST and a prior variance of 1.0 was used for CIFAR-10. Out-of-distribution detection is performed on MNIST for models trained on FashionMNIST and on SVHN for models trained on CIFAR-10.