Approximate Posteriors in Neural Networks: A Sampling Perspective

Julius Kobialka*, **Emanuel Sommer***
Department of Statistics, LMU Munich
Munich Center for Machine Learning (MCML)
{julius,emanuel}@stat.uni-muenchen.de

Juntae Kwon
Department of Statistics, LMU Munich
J.Kwon@campus.lmu.de

Daniel Dold
HTWG Konstanz
daniel.dold@htwg-konstanz.de

David Rügamer
Department of Statistics, LMU Munich
Munich Center for Machine Learning (MCML)
david@stat.uni-muenchen.de

Abstract

The landscape of neural network loss functions is known to be highly complex, and the ability of gradient-based approaches to find well-generalizing solutions to such high-dimensional problems is often considered a miracle. Similarly, Bayesian neural networks (BNNs) inherit this complexity through the model's likelihood. In applications where BNNs are used to account for weight uncertainty, recent advantages in sampling-based inference (SAI) have shown promising results outperforming other approximate Bayesian inference (ABI) methods. In this work, we analyze the approximate posterior implicitly defined by SAI and uncover key insights into its success. Among other things, we demonstrate how SAI handles symmetries differently than ABI, and examine the role of overparameterization. Further, we investigate the characteristics of approximate posteriors with sampling budgets scaled far beyond previously studied limits and explain why the localized behavior of samplers does not inherently constitute a disadvantage.

1 Introduction

By treating network weights probabilistically, Bayesian neural networks (BNNs) enable various applications, e.g., quantifying uncertainty, as a basis for active learning pipelines, improved optimization of the network itself, or compression approaches (Papamarkou et al., 2024). BNN research, alas, faces challenges in posterior inference due to the posterior's high complexity (Izmailov et al., 2021) and the overparametrization of neural networks, inducing symmetries that impede performance (see, e.g., Wiese et al., 2023; Gelberg et al., 2024).

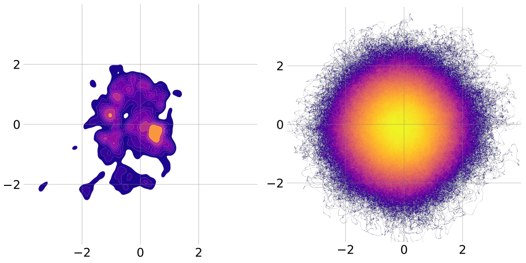

Figure 1: Evolution of the approximate posterior distribution of SAI, illustrated by marginal bivariate densities of two weights (axes, the lighter the higher the density) for 10 (left) and 10k (right) chains of posterior samples with 10k and 10M posterior samples visualized respectively.

While BNNs have been extensively studied in the context of approximate Bayesian inference (ABI), sampling-based inference (SAI) has only recently gained more traction due to its improved practical feasibility (see, e.g., Sommer et al., 2024; 2025). In contrast to ABI methods that come with certain inductive biases, SAI is tasked with a seemingly impossible idea: Obtaining a set of representative samples from a typically high dimensional and very complex posterior without approximation assumptions that help mitigating

[*] Equal contribution

non-identifiabilities of the network's weight mapping. To better understand the challenges and opportunities related to SAI for BNNs, it is crucial to comprehend the nature of SAI's (approximate) posterior. While most sampling approaches are designed to yield samples from the true posterior in the limit, this is a theoretical property. For BNNs in particular, sampling methods will be much more of a (local) approximation in comparison to other applications of SAI. It is, however, unclear, to what extent the posterior obtained by SAI is subject to an (implicit) approximation (cf. Figure 1).

Our Contributions In this work, we study the posterior obtained by SAI from three different angles: First, we investigate the behavior of SAI and the influence of prior distributions on overparameterization. Next, we discuss how non-identifiabilities will influence SAI and whether a specific treatment is necessary to address these. By analyzing the approximate posterior in the limit with an unprecedentedly large sampling budget, we further find that the posterior induced by recent SAI methods might be much more well-behaved than previously thought. Finally, we explore the practical ramifications of our findings, suggesting that local sampler behavior does not hinder the robustness and superior performance of SAI in real-world applications, and provide detailed guidelines for optimizing its performance.

2 Related Work and Open Questions

Characterizing epistemic uncertainty in machine learning is one of the main goals of probabilistic inference (Hüllermeier & Waegeman, 2021). For neural networks, a Bayesian approach has been identified as a promising direction early on (Tishby & Solla, 1989; MacKay, 1992). While some approaches such as Monte Carlo dropout (Gal & Ghahramani, 2016) and deep ensembles (Lakshminarayanan et al., 2017) have been shown to relate to the idea of BNNs, most research revolves around approximate Bayesian methods. Seminal work includes the introduction of probabilistic backpropagation approaches (Blundell et al., 2015; Hernández-Lobato & Adams, 2015), variational inference (see, e.g., Blei et al., 2017), Laplace approximation (see, e.g., Ritter et al., 2018; Daxberger et al., 2021a), as well as local approximations such as linearization (Immer et al., 2021), subnetwork inference (Daxberger et al., 2021b), or subspace inference (Izmailov et al., 2020; Dold et al., 2024).

Sampling-Based Inference An alternative to approximate approaches is sampling-based inference (SAI), usually relying on Markov Chain Monte Carlo (MCMC) methods. SAI is often considered a gold standard (see, e.g., Farquhar et al., 2020) as sampling approaches can be designed to sample from the true posterior (in the limit). While successful for smaller models or simpler hypotheses, SAI remains a challenge in high dimensions and is often considered impractical for BNNs (Papamarkou et al., 2022). With the ulterior goal to characterize the posterior landscape of BNNs (Izmailov et al., 2021), recent approaches have made progress to allow for better mixing of Markov chains (Sen et al., 2024), devising strategies for treatment of symmetries (Wiese et al., 2023; Laurent et al., 2024), and scaling samplers for larger datasets (Chen et al., 2014; Zhang et al., 2020) as well as parameter spaces (Sommer et al., 2024; 2025).

Challenges in Posterior Characterization The challenges in characterizing the posterior of BNNs using SAI are related to several factors. Two well-known reasons are the high dimensionality and multimodality of the parameter space. In addition, overparametrization of the neural network $f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$ causes non-identifiability w.r.t. its parameters $\boldsymbol{\theta} \in \boldsymbol{\Theta} \subseteq \mathbb{R}^d$. More specifically, for data $\mathcal{D} := \{(y_i, \mathbf{x}_i)\}_{i=1}^n$ with $(y_i, \mathbf{x}_i) \in \mathcal{X} \times \mathcal{Y}$, non-identifiability usually refers to parameter sets $\boldsymbol{\theta}$ in the weight space $\boldsymbol{\Theta}$ of the neural network that lead to the same functional mapping (Hecht-Nielsen, 1990), i.e., $\exists \boldsymbol{\theta}, \tilde{\boldsymbol{\theta}} \in \boldsymbol{\Theta}, \boldsymbol{\theta} \neq \tilde{\boldsymbol{\theta}} : f_{\boldsymbol{\theta}}(\mathbf{x}) = f_{\tilde{\boldsymbol{\theta}}}(\mathbf{x}) \, \forall \mathbf{x} \in \mathcal{X}$.

Symmetries One of the most prominently discussed non-identifiabilities is symmetry (Villar et al., 2024). Showing functional equivalence of networks when parameters admit an equivalence relationship, i.e., $\boldsymbol{\theta} \sim \tilde{\boldsymbol{\theta}} \Rightarrow f_{\boldsymbol{\theta}}(\mathbf{x}) = f_{\tilde{\boldsymbol{\theta}}}(\mathbf{x})$ is more straightforward (Pourzanjani et al., 2017; Petzka et al., 2020; Phuong & Lampert, 2020; Bona-Pellissier et al., 2023), while deriving parameter equivalence from equivalent outputs, i.e., $f_{\boldsymbol{\theta}}(\mathbf{x}) = f_{\tilde{\boldsymbol{\theta}}}(\mathbf{x}) \Rightarrow \boldsymbol{\theta} \sim \tilde{\boldsymbol{\theta}}$ requires stronger and often impractical assumptions (Rolnick & Kording, 2020; Phuong & Lampert, 2020; Bona-Pellissier et al., 2023). Recent work has also identified symmetries as the origin of low-capacity saddle points (Li et al., 2019; Ziyin et al., 2023). This suggests improvements in model performance and/or optimization when removing symmetries from the network (as, e.g., suggested in Ziyin et al., 2025). Proposals to deal with symmetries include bias sorting (Pourzanjani et al., 2017) or skip connections (Kurle et al., 2021) to remove permutation invariances, using invariant networks (Cohen & Welling, 2016; Zaheer et al., 2017; Hartford et al., 2018; Maron et al., 2019; Navon et al., 2023), removing scaling symmetries via regularization (Laurent et al., 2024), or computing a model average over symmetry orbits (Gelberg et al., 2024). While symmetries are also known to slow down sampling in SAI (Nalisnick, 2018; Papamarkou et al., 2022; Wiese et al., 2023), only a few papers have studied symmetries in SAI .

Open Research Questions It remains an open research question to what extent 1) the high-dimensionality and overparametrization of models change the BNN posterior obtained through SAI methods and how this, in turn, affects derived uncertainty quantification (UQ) statements. Overparamatrization also induces non-identifiabilities, making it unclear 2) whether SAI is even able to navigate such complex landscapes.

Setup In this and the following sections, we will study L-layer perceptrons given by $f_{\boldsymbol{\theta}}(\mathbf{x}) = \bigcirc_{l=1}^{L}(\phi^{(l)} \circ h^{(l)})(\mathbf{x})$, with linear function $h^{(l)}(\mathbf{x}) = \mathbf{W}^{(l)}\mathbf{x} + \mathbf{b}^{(l)}$, $\mathbf{W}^{(l)} \in \mathbb{R}^{d_l \times d_{l-1}}, \mathbf{b}^{(l)} \in \mathbb{R}^{d_l}, d_l \in \mathbb{N}$ for $l \in \{1, \dots, L\} =: [L]$, and monotonic activation function $\phi^{(l)}$. Due to their popularity, special emphasis will be placed on ReLU networks with $\phi^{(l)} \equiv \phi = \max(\cdot, 0), l \in [L-1]$. The vector $\boldsymbol{\theta}$ then denotes the flattened and stacked weights and biases with $d := \sum_{l \in [L]} d_l \cdot (d_{l-1}+1)$. If the weight of a layer is a vector, it will be denoted as $\mathbf{w}^{(l)}$, and as $w^{(l)}$, if it is a scalar. In the following, we also will assume a data distribution $y_i \overset{ind.}{\sim} \mathcal{F}(\mathbf{x}_i, \boldsymbol{\theta})$ with density $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) = \prod_{i \in [n]} p(y_i|\mathbf{x}_i, \boldsymbol{\theta})$. We will denote the prior as $p(\boldsymbol{\theta})$ and the posterior as $p(\boldsymbol{\theta}|\mathbf{X}, \mathbf{y})$. If not stated differently, we will assume a Gaussian prior $\theta_j \sim \mathcal{N}(0, \tau_j^2)$ with variance $\tau_j^2 \equiv \tau^2$ for all θ_j in $\boldsymbol{\theta}$.

3 The Effect of Overparametrization

3.1 Univariate Narrow Bayesian Networks

In the following, we start and try to build an understanding of the effects of high dimensionality and overparameterization as major drivers of posterior complexity. As an instructive example, we consider narrow Bayesian neural networks of depth $L \equiv d$ with univariate input and $h^{(l)}(x) = w^{(l)}x$. If $\phi(\cdot) = \text{Id}(\cdot)$, $f_{\boldsymbol{\theta}}$ is a narrow linear network. Given independent and identically distributed (i.i.d.) data $\{(x_i, y_i)\}_{i=1}^n$ and model assumption $y_i \sim \mathcal{N}(f_{\boldsymbol{\theta}}(x_i), \sigma^2)$, the Maximum Likelihood solution $\hat{\boldsymbol{\theta}} = \hat{\mathbf{w}} = (\hat{w}^{(L)}, \dots, \hat{w}^{(1)})$ of $\arg\min_{\mathbf{w}} p(\mathbf{y}|\mathbf{x}, \mathbf{w})$ is a set defined by $\hat{\mathbf{w}} = \{\mathbf{w} : \prod_l \hat{w}^{(l)} = (\sum_{i=1}^n x_i^2)^{-1} \sum_{i=1}^n x_i y_i =: \hat{\beta}\}$. It is straightforward to see that this network admits a scaling symmetry, as we can multiply any of the $\hat{w}^{(l)}$ by a factor $c \in \mathbb{R}^+$ and divide another $\hat{w}^{(\tilde{l})}, \tilde{l} \neq l$, by c, which will still be a maximizer of $p(\mathbf{y}|\mathbf{x}, \mathbf{w})$. By introducing weight decay, we can reduce this scaling symmetry to a sign-flip symmetry with only $|\hat{\mathbf{w}}| = 2^{L-1}$ possible solutions (or 2^L if L is even and $\text{sgn}(\hat{\beta}) = 1$), admitting

$$\hat{\mathbf{w}}_{\text{pen}} = \{\mathbf{w} : w^{(l)} = \pm\hat{\beta}^{1/L} \wedge \textstyle\prod_l \hat{w}^{(l)} = \text{sgn}(\hat{\beta})\}. \tag{1}$$

This is because the two optimization problems

$$\hat{\mathbf{w}}_{\text{pen}} := \arg\min_{\mathbf{w}} - \log p(\mathbf{y}|\mathbf{x}, \mathbf{w}) + \psi \|\mathbf{w}\|_2^2 \tag{2}$$

$$\hat{\beta}_{\text{pen}} := \arg\min_{\beta} - \log p(\mathbf{y}|\mathbf{x}, \beta) + \psi L |\beta|^{2/L} \tag{3}$$

for regularization parameter $\psi \in \mathbb{R}^+$ can be shown to have the same minima (see, e.g. Kolb et al., 2025). Note that as $\|\mathbf{w}\|_2^2$ will increase linearly with L, the optimization problem requires a smaller amount of regularization (or weight decay) ψ for increasing L.

The Effect on ABI Using MFVI with a factorized Gaussian $q(\mathbf{w}|\boldsymbol{\zeta}) = \prod_{l \in [L]} \mathcal{N}(w^{(l)}|\mu_l, \sigma_l^2)$ as an example, the KL divergence $\mathbb{KL}(q(\mathbf{w}) \| p(\mathbf{w}|\mathbf{x}, \mathbf{y}))$ is given by

$$\textstyle\sum_{l \in [L]} [\log(\tau^2/\sigma_l^2) - 1 + (\sigma_l^2 + \mu_l^2)/\tau^2]/2.$$

Assuming $\sigma_l^2 \equiv \sigma^2 = \tau^2$, we see a similar result as in the case of Equations (2) and (3): When keeping τ^2 and n fixed and assuming a constant expected likelihood term while increasing L, the KL divergence in the ELBO increases with $\sum_l \mu_l^2/(2\tau^2)$. Let $\nu := \prod_{l=1}^{L} \mu_l$. Using the AM-GM inequality, we know that $\sum_{l\in[L]} \mu_l^2 \geq L \cdot |\nu|^{2/L}$, where the right-hand side corresponds to the problem's minimum norm solution $\hat{\mu}_l \equiv |\nu|^{1/L}\, \forall l \in [L]$. This means that the optimization is discouraged from choosing any μ_l different from $|\nu|^{1/L}$ as there would otherwise always be a smaller KL divergence for the same functional output $f_{\boldsymbol{\theta}}$. This has two consequences: 1) The individual solutions $\hat{\mu}_l$ will converge to 0 as L increases and 2) if not adapted with L, the prior will become more and more influential in the objective via the KL divergence. Both observations assume the (expected) likelihood to remain constant or at least not to grow with L. In a simple model like in this section, this seems to be a reasonable assumption.

Sampling-based Inference While normally distributed priors can be shown to have a certain mathematical equivalence to L_2-penalties and weight decay, SAI does not seek to find a surrogate posterior with strong assumptions that best match the true posterior; rather, it aims to approximate the true posterior without imposing restrictive assumptions. The setting is therefore different from the previous scenarios. Now assume that for increasing network size d, the likelihood does not increase—an assumption that can be well justified in the univariate network above since it does not increase in capacity with more layers L. Then, the prior influence on the posterior will become more and more dominant for increasing L. This is because the likelihood is bounded while the log-prior $p(\boldsymbol{\theta}) \propto \tau^{-2} \sum_{j=1}^{d} \theta_j^2$ increases linearly (since $d^{-1} \log p(\boldsymbol{\theta})$ converges to 1 almost surely by the strong law of large numbers) and hence any non-zero value for $\hat{\beta}$ becomes exponentially suppressed. This, in turn, suggests that the posterior could potentially even become simpler in structure in higher dimensions when maintaining the model performance. Intuitively, a larger network will have more flexibility and thereby also more freedom to adhere to the prior distribution.

3.2 Increasing Width

Moving to a one-hidden-layer network with $d_1 > 1$ units, it is well-known that $f_{\boldsymbol{\theta}}$ converges to a Gaussian process for $d_1 \to \infty$. However, this is only the case if the prior is adjusted for the number of parameters (see, e.g., Neal, 2012). In a BNN setting, it is, however, common practice not to increase the prior variance for different model sizes and use standard normal or Laplace priors independent of the network size (Fortuin, 2022; Sommer et al., 2024). This again induces more regularization toward zero compared to approaches that adapt the variance with parameter size. At the same time, the flexibility of the network increases, allowing the sampler to more easily adhere to the prior.

Analogous to the univariate narrow network, there is a form of exchangeability of all but the first and last weight in deep and wide linear networks (Ziyin et al., 2024, Theorem 5.4). Since the capacity of this network also does not grow with L, we expect a similar behavior as before for all intermediate layers: A posterior that is increasingly concentrated around zero, with marginals that progressively resemble the prior. In contrast to before, the first and last layers' weights have a special role being connected to the data, and will thus be more influenced by the likelihood. We confirm this hypothesis also empirically in a later experiment, depicted in Figure 7.

3.3 ReLU Activation and Biases

In contrast to linear networks, it is much more difficult to get an intuition of posterior properties when including non-linearities. To further reaffirm our conjecture, we turn to two-layer neural networks with ReLU-activation in the hidden layer $f_{\boldsymbol{\theta}}(\mathbf{x}) = {\mathbf{w}^{(2)}}^\top \phi(\mathbf{W}^{(1)}\mathbf{x} + \mathbf{b}^{(1)})$. Such networks have recently been shown to be "convexifiable" (Mishkin et al., 2022) and, in a non-Bayesian setting, can be estimated by iterating over possible activation patterns while solving a group lasso problem. In Appendix D, we discuss the properties of this network in the Bayesian setting.

While not yielding a concave posterior density, we can show that the density is unimodal in the product of weights $\mathbf{u} := {\mathbf{w}^{(2)}}^\top \mathbf{W}^{(1)}$ (up to permutations), hence even less sophisticated samplers should be able to navigate this posterior landscape. Hypothesizing that the shape of the posterior should reflect previously found patterns when the network is chosen flexible enough, we conduct SAI for a small network with 8 hidden neurons and a larger, more flexible one with 64 hidden neurons.

Empirical Findings A selection of resulting uni- and bivariate marginal densities of $\mathbf{u}$ are shown in Figure 8 and Figure 9 in the Appendix. The results confirm our hypothesis of unimodality of the posterior density of $\mathbf{u}$ and, in particular, that with increased dimensions, more force is exerted by the prior onto the sampler, pulling the otherwise strongly varying distributions (Figure 8, left) for different sampling runs together and concentrating around the origin (Figure 8, right).

General ReLU Networks In the above example, we found not only $\mathbf{u}$ to be unimodal, but the same also holds for entries in $\mathbf{W}_1$ and $\mathbf{w}_2$. This can be explained as follows: If the first layer's activations are distributed symmetrically around zero and d_1 is large enough, all weights in intermediate layers should have an equal likelihood of being negative or positive. This allows the network to be (more) compliant with the zero mean prior and therefore a similar picture as in the deep linear network arises for the intermediate layer's weights. This in turn requires the biases to regulate the product of weights and previous layer activations, which should paint a clear picture in the biases' posteriors. Figure 7 confirms our conjecture, showing more distinct patterns for the first layer weights and biases, whereas intermediate layers tend to reflect the shape of the prior.

4 Non-Identifiabilities

The previous section suggests that high dimensionality, in particular overparametrization, might potentially even work in favor of SAI (or in general probabilistic methods). However, this says little about the effectiveness of traversing the posterior, and non-identifiabilities de facto still exist in the model, even in high dimensions. We will illustrate this for symmetries in the following but also address other challenges in posterior sampling afterwards.

4.1 Symmetries

While there exists a multitude of different classifications of symmetries, we focus on countable or discrete and uncountable symmetries. We refer to Appendix A for their definitions and related literature. A typical example of (the cause of) a countable symmetry is the interchangeability of neurons within one layer—a permutation symmetry. By swapping the in- and outgoing weights of two neurons in the same layer, we obtain the same function $f_{\boldsymbol{\theta}}$ but modified weights $\mathbf{W}^{(l)}, \mathbf{W}^{(l+1)}$. For a deep linear network $f_{\boldsymbol{\theta}}(\mathbf{x}) = \mathbf{W}^{(L)} \cdots \mathbf{W}^{(l-1)}\mathbf{W}^{(l)} \cdots \mathbf{W}^{(1)}\mathbf{x}$, it is easy to see that applying such a permutation using a permutation matrix $\mathbf{P} \in \{0,1\}^{d\times d}$, the network with permuted weights $\tilde{f}_{\boldsymbol{\theta}}(\mathbf{x}) = \mathbf{W}^{(L)} \cdots (\mathbf{W}^{(l-1)}\mathbf{P}^{\top})(\mathbf{P}\mathbf{W}^{(l)}) \cdots \mathbf{W}^{(1)}\mathbf{x} = f_{\boldsymbol{\theta}}(\mathbf{x})$ since $\mathbf{P}^{\top}\mathbf{P} = \mathbf{I}_d$ ($\mathbf{P}$ is an orthogonal matrix). In contrast to permutation symmetries, uncountable symmetries such as (positive) scaling symmetries result in an infinite amount of equivalent models. This is easy to see, e.g., for the deep linear network: $\forall c \neq 0, \tilde{f}_{\boldsymbol{\theta}}(\mathbf{x}) = \mathbf{W}^{(L)} \cdots (\mathbf{W}^{(l-1)} \cdot {}^{1}\!/\!{}_{c})(c \cdot \mathbf{W}^{(l)}) \cdots \mathbf{W}^{(1)}\mathbf{x} = f_{\boldsymbol{\theta}}(\mathbf{x})$. In contrast to permutation symmetries, uncountable symmetries are connected in weight space. This makes it much more likely for methods to traverse the (generalized hyperbolic) manifold created by these symmetries while not providing any functional diversity.

4.2 Treatment in ABI

When trying to approximate the posterior $p(\mathbf{w}|\mathbf{X},\mathbf{y})$ using an approximate distribution $q(\mathbf{w}|\boldsymbol{\zeta})$ with parameters $\boldsymbol{\zeta} \in \mathbb{R}^z, z \in \mathbb{N}$, a common approach is the minimization of the (reverse) Kullback-Leibler divergence (KLD): $\mathbb{KL}(q(\mathbf{w}|\boldsymbol{\zeta})\|p(\mathbf{w}|\mathbf{X},\mathbf{y}))$. Being mode-seeking by nature and relying on stochastic optimization, such an ABI approach therefore ideally solves the problem of permutations and other countable symmetries implicitly by focusing only on one of multiple modes. In general, approaches that assume a unimodal distribution around a learned posterior mean or the maximum a-posteriori (MAP) estimator $\hat{\boldsymbol{\theta}}$ (such as Laplace approximation) will not be affected by axis-mirrored solutions—at least when considering the uncertainty of a single model optimization run. In contrast to variational inference (VI) approaches, deep ensembles (DE) can be potentially harmed in their expressiveness by running into permuted but functionally identical solutions.

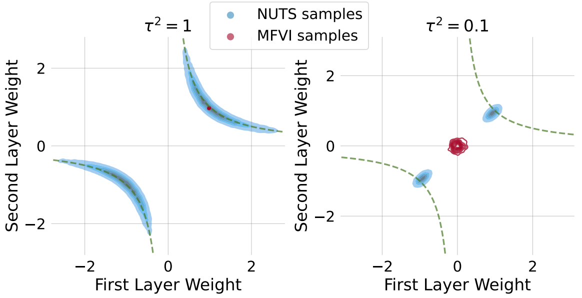

Figure 2: Posterior of the network $f_{\boldsymbol{\theta}}(x) = w_2(w_1 x)$ as characterized by SAI (using multiple chains of the NUTS sampler) and ABI using mean-field VI with different prior variances τ^2. According to Equation (3), the minimum norm solution is given by $w_1 w_2 = 1$.

As can be seen in Figure 2, ABI methods are also often "immune" by nature to uncountable symmetries such as scaling symmetries as their local approximation around an optimized solution $\hat{\boldsymbol{\theta}}$ and stochastic optimization will again focus on one specific solution. In addition, commonly used ABI methods can be shown to be scale-invariant. For example, applying a non-zero prior with slightly different σ_j^2 for every parameter θ_j in a mean-field VI (MFVI, Blei et al., 2017) approach can be shown to remove permutation and scaling symmetries with probability 1 (see Appendix A.1)

While ABI will not always be "immune" as pointed out by Gelberg et al. (2024) and apparent in Figure 2, there is explicit treatment of symmetries for these optimization-based methods. Examples include bias sorting (Pourzanjani et al., 2017) or the above-described adjustment to the prior distribution, which modifies the optimization objective as proposed by (Ziyin et al., 2025). These approaches enhance neuron identifiability and (theoretically) remove permutation symmetries.

4.3 Treatment in SAI

Unlike ABI methods, SAI approaches do not rely on local approximation assumptions, making them more susceptible to symmetries. For example, a sampler might oscillate between modes created by parameter permutations, wasting computational resources on transitions rather than exploring local regions around modes. This raises the question of whether a sampler will remain within a symmetry-induced manifold at any given time.

Theoretical Considerations While priors in SAI may initially appear to guide solutions toward minimum norm configurations, as seen in Laurent et al. (2024), this holds only for simple networks with two weights. Counterexamples emerge in networks with three or more weights (Appendix A.2). Yet, SAI possesses "built-in protective mechanisms" similar to ABI that prevent it from being trapped in symmetries. First, because SAI proposes new states probabilistically, it avoids being locked between countable symmetries (Figure 2) and eventually selects a direction by chance. Second, uncountable symmetries can be broken by introducing slight variations in regularization across weight dimensions, akin to using a non-constant diagonal mass matrix in Hamiltonian Monte Carlo (Ziyin et al., 2025). More generally, since the sampler operates probabilistically, it almost surely does not remain in symmetry manifolds, as these have a probability measure of zero.

Practical Considerations The previous paragraph explains why samplers might not exactly follow symmetry manifolds, leaving open the question of whether their existence impairs sampling performance. While no definite answer exists and pathological examples or challenging applications can always be found[1], we provide arguments and empirical evidence in the following section showing that SAI is not notably affected by symmetries or other non-identifiabilities in practice.

4.4 Regions of (Almost) Zero Probability

Non-identifiability includes regions where $f_{\boldsymbol{\theta}}$ learns nothing, resulting in (near) zero likelihood with nonzero probability mass. This can occur, for example, when a ReLU activation remains inactive across all data points. In models with bottlenecks or when this affects the output neuron, $f_{\boldsymbol{\theta}}$ becomes non-identifiable. Once a sampler enters such a region, it cannot move, similar to regions of extremely low probability. To mitigate this, samplers can be warm-started using solutions from a pre-optimized non-Bayesian network, which has proven highly effective (Sommer et al., 2024).

[1] This is also the reason for the great variety of samplers, each trying to solve a differently structured problem.

5 Sampling in Practice

BNN posteriors are highly complex. In Section 3, we argue that in overparameterized models, the prior can dominate, shaping the posterior. This assumes a well-performing model. Likewise, Section 4.4 advocates warm-starting the sampling process. To navigate posterior complexities, we identify three key strategies: 1) warm-starts, 2) multiple chains, and 3) focusing on localized exploration. The following sections detail their role in addressing non-identifiabilities and other challenges.

5.1 Countable Symmetries

To avoid countable symmetries like permutations, SAI can be designed to prioritize local exploitation, minimizing jumps between symmetrical solutions. Since permuted solutions reside in different orthants, transitioning between modes requires crossing the origin, which is unlikely within a limited computational budget. Even if two chains explore the same mode in different orthants, SAI remains largely unaffected. Evidence is provided in Figure 3, showing that sampling will still provide functionally diverse models (measured using the log posterior predictive density, LPPD for short) when starting 11 chains from permutations of a 12th chain. This experiment suggests a cost-effective alternative to Bayesian Deep Ensembling (BDE) (e.g., Sommer et al., 2025), requiring only a single pre-trained model while achieving comparable ensemble performance.

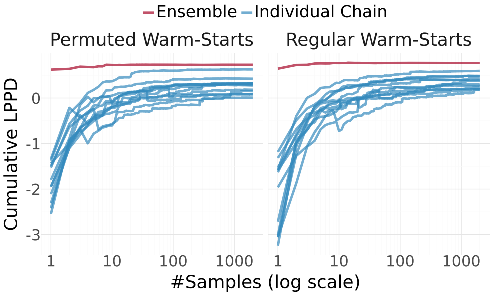

Figure 3: Cumulative LPPD of individual and ensembled sampler chains, initialized from functionally equivalent and diverse warm-starts.

5.2 Uncountable Symmetries

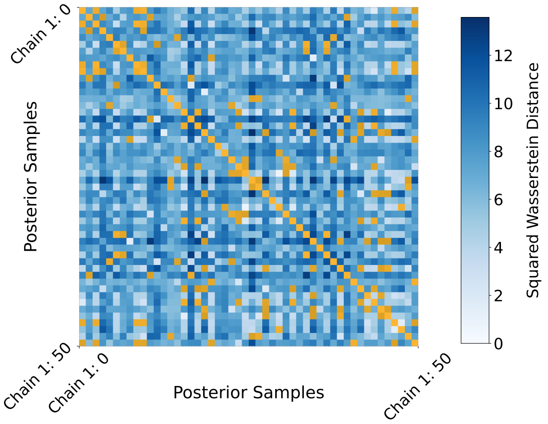

Figure 4: Pairwise squared Wasserstein-2 distances of the predictive distribution induced by each sample from a warm-start initialized sampler chain on the `airfoil` dataset. Non-significant distances are depicted in orange.

To analyze how trajectories of a sampler behave over time and whether the resulting samples show signs of functional equivalence, we analyze the predictive distribution for each sample $s \in [S]$. In two different iterations of the sampler, we obtain weight sets $\boldsymbol{\theta}^{(s)}, \boldsymbol{\theta}^{(\tilde{s})}$, which yield predictive distributions $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}^{(s)})$ and $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}^{(\tilde{s})})$. Using a Gaussian assumption for the data distribution, i.e., $\mathbf{y}|\mathbf{X}, \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma}^2))$, the two weight vectors yield two distributions $\mathcal{N}(\boldsymbol{\mu}^{(s)}, \mathrm{diag}((\boldsymbol{\sigma}^2)^{(s)}))$ and $\mathcal{N}(\boldsymbol{\mu}^{(\tilde{s})}, \mathrm{diag}((\boldsymbol{\sigma}^2)^{(\tilde{s})}))$, respectively. To assess the distance between samples within a chain, we compute the Wasserstein-2 distance between their distributions. Large distances indicate sampled models that are functionally diverse. Given that both distributions are normal, the difference $\mathbf{y}^{(s)} - \mathbf{y}^{(\tilde{s})}$ follows a normal distribution with mean $\boldsymbol{\mu}^{(s)} - \boldsymbol{\mu}^{(\tilde{s})}$ and variance $(\boldsymbol{\sigma}^2)^{(s)} + (\boldsymbol{\sigma}^2)^{(\tilde{s})} - 2\boldsymbol{\rho} \odot \boldsymbol{\sigma}^{(\tilde{s})} \odot \boldsymbol{\sigma}^{(\tilde{s})}$. To determine if the Wasserstein-2 distances reflect significant distributional changes, we compute Z-statistics for each sample pair $\boldsymbol{\theta}^{(s)}, \boldsymbol{\theta}^{(\tilde{s})}$ and compare them to the 97.5% standard normal quantile, testing at a 5% significance level (see Appendix C.5). Results in Figure 4 suggest that most sampled models are functionally diverse.

5.3 Limiting Distribution and Local Approximations

Beyond near-zero probability regions and symmetry manifolds, Sommer et al. (2024) report disconnected modes in the SAI posterior. However, it remains unclear whether these arise intrinsically or from insufficient sampling. To investigate, we extend their analysis to 10 000 chains with 1000 samples each, compared to their original 12 chains. As shown in Figure 1, the apparent disconnectedness in the margins is primarily a result of limited chains as well as the local dynamics of the sampler rather than an intrinsic property of the posterior. Further analysis of bivariate densities of the kernel and bias weights across layers (Figure 7 in the Appendix) reveals no evidence of marginally disconnected modes, while intermediate layers resemble the (shifted) priors. In particular, the kernel parameters seem to follow a multivariate zero-centered Gaussian-like distribution and first/last layers exhibit rather distinct roles as hypothesized in Section 3. The results also confirm our conjecture that the shifted margins of the biases align with ReLU activations centering hidden states. Consult Appendix C.1 for an extended discussion.

Local Exploration is Just Fine Building on our previous discussion, the effectiveness of SAI is questionable given its need for extensive sampling to characterize large posterior regions. However, a local approximation using short-chain ensembles proves sufficient for UQ, as shown in Figure 5 and our benchmark. Results with the MILE sampler (Sommer et al., 2025) indicate that UQ performance quickly saturates with fewer chains, which are easily parallelized on modern hardware. Practically, localization depends on the number of samples per chain, assuming an optimal step size for retaining a stable likelihood. Prioritizing local exploration mitigates non-identifiability issues (e.g., permutation and sign-flip symmetries) while improving computational efficiency.

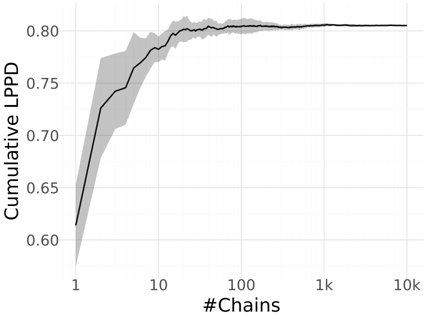

Figure 5: Cumulative LPPD over the number of chains (standard deviation across 5 random chain orderings).

Finally, we run a benchmark to demonstrate that SAI with previous considerations is a viable alternative to ABI. This further supports that properly executed sampling is effective in practice and not hindered by the intricacies of the posterior landscape. For this purpose, we compare two prominent ABI methods to different SAI strategies across a range of regression tasks. We report results in Table 2 (see Appendix C.7) using the root mean squared error (RMSE) and the LPPD for UQ. In these settings, Laplace achieves good predictive performance while ranking last in terms of LPPD. In contrast, MFVI provides improved uncertainty quantification but sacrifices predictive accuracy. DEs with 10 members demonstrate robust performance in both metrics. BDEs achieve the highest performance among all methods, even with a single chain and show improved predictive accuracy and UQ when ensembling 10 chains.

6 Discussion, Limitations and Future Research

We analyzed the approximate posterior induced by sampling-based inference for BNNs, focusing on symmetries, overparameterization, and prior influences. Our theoretical and empirical findings suggest a well-behaved posterior, supporting SAI as a viable alternative to ABI. Our approach assumes initialization from multiple well-performing neural networks akin to a DE. DEs have been shown not to yield a proper characterization of the posterior in general but to provide a mixture of distributions related to their initialization value and nearby optima instead (Wild et al., 2024). While BDEs theoretically converge to the true posterior with infinite samples, practical sampling is biased toward reachable regions when starting from DE solutions. Hence, while our approach may not yield exact posterior samples, it effectively captures the epistemic uncertainty relevant to practitioners: Given a reasonably well-performing neural network, how much parameter uncertainty remains in the model? Put differently, a practitioner would seldom be interested in the subspace of (useless) models where the posterior is (almost) zero. We primarily considered full-batch sampling, noting that while stochastic samplers may further mitigate symmetry issues, they require improvements in robustness and hyperparameter sensitivity for practical use, which is a promising avenue for future research.

REFERENCES

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight Uncertainty in Neural Networks. In *Proceedings of the 32nd International Conference on Machine Learning*. arXiv, 2015.

Joachim Bona-Pellissier, François Bachoc, and François Malgouyres. Parameter Identifiability of a Deep Feedforward ReLU Neural Network. *Machine Learning*, 112(11):4431–4493, 2023.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: Composable transformations of Python+NumPy programs, 2018. URL `http://github.com/google/jax`.

Alberto Cabezas, Adrien Corenflos, Junpeng Lao, and Rémi Louf. BlackJAX: Composable Bayesian inference in JAX, 2024.

Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic Gradient Hamiltonian Monte Carlo. In *International Conference on Machine Learning*, pp. 1683–1691. PMLR, 2014.

Taco Cohen and Max Welling. Group Equivariant Convolutional Networks. In *International Conference on Machine Learning*, pp. 2990–2999. PMLR, 2016.

Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace Redux – Effortless Bayesian Deep Learning. In *35th Conference on Neural Information Processing Systems (NeurIPS 2021)*, 2021a.

Erik Daxberger, Eric Nalisnick, James U Allingham, Javier Antorán, and José Miguel Hernández-Lobato. Bayesian Deep Learning via Subnetwork Inference. In *International Conference on Machine Learning*, pp. 2510–2521. PMLR, 2021b.

Gianluca Detommaso, Alberto Gasparin, Michele Donini, Matthias Seeger, Andrew Gordon Wilson, and Cedric Archambeau. Fortuna - A Library for Uncertainty Quantification.

Daniel Dold, David Rügamer, Beate Sick, and Oliver Dürr. Semi-Structured Subspace Inference. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research. PMLR, 2024.

Dheeru Dua and Casey Graff. UCI Machine Learning Repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Hadi Fanaee-T. Bike Sharing Dataset. UCI Machine Learning Repository, 2013.

Sebastian Farquhar, Lewis Smith, and Yarin Gal. Liberty or Depth: Deep Bayesian Neural Nets Do Not Need Complex Weight Posterior Approximations. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020.

Vincent Fortuin. Priors in Bayesian Deep Learning: A Review. *International Statistical Review*, 90 (3):563–591, 2022.

Vincent Fortuin, Adrià Garriga-Alonso, Sebastian W. Ober, Florian Wenzel, Gunnar Ratsch, Richard E Turner, Mark van der Wilk, and Laurence Aitchison. Bayesian Neural Network Priors Revisited. In *International Conference on Learning Representations*, 2022.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *International Conference on Machine Learning*, pp. 1050–1059. PMLR, 2016.

Yoav Gelberg, Tycho FA van der Ouderaa, Mark van der Wilk, and Yarin Gal. Variational Inference Failures Under Model Symmetries: Permutation Invariant Posteriors for Bayesian Neural Networks. In *ICML 2024 Workshop on Geometry-grounded Representation Learning and Generative Modeling*, 2024.

A. Gelman, J. Hwang, and A. Vehtari. Understanding Predictive Information Criteria for Bayesian Models. *Statistics and Computing*, 24(6):997–1016, 2014.

Xavier Glorot and Yoshua Bengio. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256. JMLR Workshop and Conference Proceedings, 2010.

Jason Hartford, Devon Graham, Kevin Leyton-Brown, and Siamak Ravanbakhsh. Deep Models of Interactions Across Sets. In *International Conference on Machine Learning*, pp. 1909–1918. PMLR, 2018.

Robert Hecht-Nielsen. On the Algebraic Structure of Feedforward Network Weight Spaces. In *Advanced Neural Computers*, pp. 129–135. Elsevier, 1990.

José Miguel Hernández-Lobato and Ryan Adams. Probabilistic Backpropagation for Scalable Learning of Bayesian Neural Networks. In *International Conference on Machine Learning*, pp. 1861–1869. PMLR, 2015.

Matthew D Hoffman and Andrew Gelman. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15:1351–1381, 2014.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods. *Machine Learning*, 110(3):457–506, 2021.

Alexander Immer, Maciej Korzepa, and Matthias Bauer. Improving Predictions of Bayesian Neural Nets via Local Linearization. In *International Conference on Artificial Intelligence and Statistics*, pp. 703–711. PMLR, 2021.

Pavel Izmailov, Wesley J. Maddox, Polina Kirichenko, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Subspace Inference for Bayesian Deep Learning. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pp. 1169–1179, 2020.

Pavel Izmailov, Sharad Vikram, Matthew D. Hoffman, and Andrew Gordon Wilson. What Are Bayesian Neural Network Posteriors Really Like? In *Proceedings of the 38th International Conference on Machine Learning, PMLR 139,*, 2021.

Chris Kolb, Christian L. Müller, Bernd Bischl, and David Rügamer. Smoothing the Edges: A General Framework for Smooth Optimization in Sparse Regularization using Hadamard Overparametrization. (arXiv:2307.03571), 2023.

Chris Kolb, Tobias Weber, Bernd Bischl, and David Rügamer. Deep Weight Factorization: Sparse Learning Through the Lens of Artificial Symmetries. In *The Thirteenth International Conference on Learning Representations*, 2025.

Richard Kurle, Tim Januschowski, Jan Gasthaus, and Yuyang Bernie Wang. On Symmetries in Variational Bayesian Neural Nets. 2021.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *Proceedings of the 31st Conference on Neural Information Processing Systems*, 2017.

Olivier Laurent, Emanuel Aldea, and Gianni Franchi. A Symmetry-Aware Exploration of Bayesian Neural Network Posteriors. In *The Twelfth International Conference on Learning Representations*, 2024.

Xingguo Li, Junwei Lu, Raman Arora, Jarvis Haupt, Han Liu, Zhaoran Wang, and Tuo Zhao. Symmetry, Saddle Points, and Global Optimization Landscape of Nonconvex Matrix Factorization. *IEEE Transactions on Information Theory*, 65(6):3489–3514, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, 2019.

David JC MacKay. A practical bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992.

Haggai Maron, Heli Ben-Hamu, Nadav Shamir, and Yaron Lipman. Invariant and Equivariant Graph Networks. In *International Conference on Learning Representations*, 2019.

Aaron Mishkin, Arda Sahiner, and Mert Pilanci. Fast Convex Optimization for Two-Layer ReLU Networks: Equivalent Model Classes and Cone Decompositions. In *International Conference on Machine Learning*, pp. 15770–15816. PMLR, 2022.

Eric Thomas Nalisnick. *On Priors for Bayesian Neural Networks*. PhD thesis, University of California, Irvine, 2018.

Aviv Navon, Aviv Shamsian, Idan Achituve, Ethan Fetaya, Gal Chechik, and Haggai Maron. Equivariant architectures for learning in deep weight spaces. *arXiv preprint arXiv:2301.12780*, 2023.

R.M. Neal. *Bayesian Learning for Neural Networks*. Lecture Notes in Statistics. Springer New York, 2012. ISBN 9781461207450.

Theodore Papamarkou, Jacob Hinkle, M. Todd Young, and David Womble. Challenges in Markov Chain Monte Carlo for Bayesian Neural Networks. *Statistical Science*, 37(3), 2022.

Theodore Papamarkou, Maria Skoularidou, Konstantina Palla, Laurence Aitchison, Julyan Arbel, David Dunson, Maurizio Filippone, Vincent Fortuin, Philipp Hennig, José Miguel Hernández-Lobato, Aliaksandr Hubin, Alexander Immer, Theofanis Karaletsos, Mohammad Emtiyaz Khan, Agustinus Kristiadi, Yingzhen Li, Stephan Mandt, Christopher Nemeth, Michael A Osborne, Tim G. J. Rudner, David Rügamer, Yee Whye Teh, Max Welling, Andrew Gordon Wilson, and Ruqi Zhang. Position: Bayesian Deep Learning is Needed in the Age of Large-Scale AI. In *Proceedings of the 41st International Conference on Machine Learning*. PMLR, 2024.

Henning Petzka, Martin Trimmel, and Cristian Sminchisescu. Notes on the Symmetries of 2-Layer ReLU-Networks. In *Proceedings of the northern lights deep learning workshop*, volume 1, pp. 6–6, 2020.

Mary Phuong and Christoph Lampert. Functional vs. Parametric Equivalence of ReLU Networks. In *8th International Conference on Learning Representations*, 2020.

Mert Pilanci and Tolga Ergen. Neural Networks are Convex Regularizers: Exact Polynomial-time Convex Optimization Formulations for Two-layer Networks. In *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 2020.

Arya A. Pourzanjani, Richard M. Jiang, and Linda R. Petzold. Improving the Identifiability of Neural Networks for Bayesian Inference. In *Second Workshop on Bayesian Deep Learning*, 2017.

Hippolyt Ritter, Aleksandar Botev, and David Barber. A Scalable Laplace Approximation for Neural Networks. In *The 6th International Conference on Learning Representations*, 2018.

David Rolnick and Konrad Kording. Reverse-Engineering Deep ReLU Networks. In *International conference on machine learning*. PMLR, 2020.

Deborshee Sen, Theodore Papamarkou, and David Dunson. Bayesian Neural Networks and Dimensionality Reduction. In *Handbook of Bayesian, Fiducial, and Frequentist Inference*. Chapman and Hall/CRC, 2024.

Emanuel Sommer, Lisa Wimmer, Theodore Papamarkou, Ludwig Bothmann, Bernd Bischl, and David Rügamer. Connecting the Dots: Is Mode-Connectedness the Key to Feasible Sample-Based Inference in Bayesian Neural Networks? In *Proceedings of the 41st International Conference on Machine Learning*. PMLR, 2024.

Emanuel Sommer, Jakob Robnik, Giorgi Nozadze, Uros Seljak, and David Rügamer. Microcanonical Langevin Ensembles: Advancing the Sampling of Bayesian Neural Networks. In *The Thirteenth International Conference on Learning Representations*, 2025.

Naftali Tishby and Sara Solla. Consistent Inference of Probabilities in Layered Networks: Predictions and Generalizations. In *International 1989 Joint Conference on Neural Networks*. IEEE, 1989.

Ba-Hien Tran, Simone Rossi, Dimitrios Milios, and Maurizio Filippone. All You Need is a Good Functional Prior for Bayesian Deep Learning. *Journal of Machine Learning Research*, 23(74): 1–56, 2022.

Athanasios Tsanas and Angeliki Xifara. Accurate Quantitative Estimation of Energy Performance of Residential Buildings Using Statistical Machine Learning Tools. *Energy and Buildings*, 49, 2012.

Soledad Villar, David W. Hogg, Weichi Yao, George A. Kevrekidis, and Bernhard Schölkopf. Towards Fully Covariant Machine Learning. *Transactions on Machine Learning Research*, 2024.

Jonas Gregor Wiese, Lisa Wimmer, Theodore Papamarkou, Bernd Bischl, Stephan Günnemann, and David Rügamer. Towards efficient MCMC sampling in Bayesian neural networks by exploiting symmetry. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 459–474. Springer, 2023.

Veit David Wild, Sahra Ghalebikesabi, Dino Sejdinovic, and Jeremias Knoblauch. A Rigorous Link between Deep Ensembles and (Variational) Bayesian Methods. *Advances in Neural Information Processing Systems*, 36, 2024.

I-C Yeh. Modeling of Strength of High-Performance Concrete Using Artificial Neural Networks. *Cement and Concrete research*, 28(12), 1998.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep Sets. *Advances In Neural Information Processing Systems*, 30, 2017.

Ruqi Zhang, Chunyuan Li, Jianyi Zhang, Changyou Chen, and Andrew Gordon Wilson. Cyclical Stochastic Gradient MCMC for Bayesian Deep Learning. In *Proceedings of the Eighth International Conference on Learning Representations*, 2020.

Liu Ziyin. Symmetry Induces Structure and Constraint of Learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*. PMLR, 2024.

Liu Ziyin, Botao Li, Tomer Galanti, and Masahito Ueda. The Probabilistic Stability of Stochastic Gradient Descent. *arXiv preprint arXiv:2303.13093*, 2023.

Liu Ziyin, Mingze Wang, Hongchao Li, and Lei Wu. Parameter Symmetry and Noise Equilibrium of Stochastic Gradient Descent. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Liu Ziyin, Yizhou Xu, and Isaac Chuang. Remove Symmetries to Control Model Expressivity. In *The Thirteenth International Conference on Learning Representations*, 2025.

A Further Discussion on Neural Network Symmetries

Following Rolnick & Kording (2020); Phuong & Lampert (2020); Bona-Pellissier et al. (2023); Laurent et al. (2024), we introduce the most commonly discussed symmetries in the literature: permutation (discrete), sign-flip (discrete), and scaling (uncountable) symmetries (see Definitions A.1, A.3 and A.2).

Possibly one if not the most prominently discussed symmetry in neural networks is the **permutation symmetry**.

Definition A.1 (Permutation Symmetry)**.** Let $\pi(i), i \in [d]$ be a permutation of the elements in $[d]$, and $\mathbf{P} \in \{0,1\}^{d \times d}$ with elements $\{\delta_{\pi(i),j}\}_{i,j\in[d]}$ and Kronecker delta function δ. We say a network $f_{\boldsymbol{\theta}}$ contains *permutation symmetries* if $\exists \mathbf{P} \neq \boldsymbol{I}_d : f_{\boldsymbol{\theta}}(\mathbf{x}) \equiv f_{\mathbf{P}\boldsymbol{\theta}}(\mathbf{x}) \, \forall \mathbf{x} \in \mathcal{X}$.

Trivially, for any factorized isotropic prior choice the prior is invariant w.r.t. to permutation symmetries. The challenge with BNNs which previous symmetry removal approaches targeting the breaking of this invariance have not addressed is the random nature of the weights and biases. Due to their randomness, the approach by Pourzanjani et al. (2017) to sort the biases is likely ill-defined as the distributions of the parameters might still overlap, and hence a change of order in the biases might occur even without an actual neuron permutation taking place.

Further, permutation symmetries can be seen as a special case of a **scaling symmetry** (cf. Figure 2), defined as follows:

Definition A.2 (Scaling Symmetry)**.** Let $\lambda_j \in \mathbb{R}\backslash\{0\}, j \in [d]$ and $\boldsymbol{\Lambda} := \operatorname{diag}(\lambda_1, \ldots, \lambda_d)$. We say a network $f_{\boldsymbol{\theta}}$ contains *scaling symmetries* if $\exists \boldsymbol{\Lambda} \neq \boldsymbol{I}_d : f_{\boldsymbol{\theta}}(\mathbf{x}) \equiv f_{\boldsymbol{\Lambda}\boldsymbol{\theta}}(\mathbf{x}) \, \forall \mathbf{x} \in \mathcal{X}$. We further call the symmetry a *positive scaling symmetry* if it holds $\lambda_j > 0 \, \forall j \in [d]$.

While permutation symmetries can potentially be connected via scaling symmetry hyperbolas, this is usually not the case for **sign-flip symmetries** as this would imply traversing through the origin.

Definition A.3 (Sign-flip Symmetry)**.** Let $\boldsymbol{\Sigma} = \operatorname{diag}(\boldsymbol{\varsigma})$ with $\boldsymbol{\varsigma} \in \{-1, 1\}^d$. We say a network $f_{\boldsymbol{\theta}}$ contains *sign-flip symmetries* if $\exists \boldsymbol{\Sigma} \neq \boldsymbol{I}_d : f_{\boldsymbol{\theta}}(\mathbf{x}) \equiv f_{\boldsymbol{\Sigma}\boldsymbol{\theta}}(\mathbf{x}) \, \forall \mathbf{x} \in \mathcal{X}$.

Sign-flip symmetries are special in that some activation functions are invariant w.r.t. joint sign-flips in adjacent layers (such as $\tanh$), while others like the ReLU function do not admit sign-flips. Thus in this case the non-invariance of the likelihood of ReLU networks w.r.t. single sign-flips reduces the amount of posterior symmetries. The usually deployed symmetric zero-centered priors do not help in the reduction of symmetries in this setting. We can easily formalize this prior invariance in the following Proposition.

Proposition A.4. *Symmetric zero-centered factorized priors are invariant w.r.t. sign-flips.*

Proof. We have

$$p(\boldsymbol{\Sigma}\boldsymbol{\theta}) = \prod_j p(\varsigma_j \theta_j) \overset{(*)}{=} \prod_j p(\theta_j) = p(\boldsymbol{\theta})$$

where $(*)$ is due to the symmetry and zero-centeredness of the priors. □

Notably, the likelihood of ReLU although non-invariant w.r.t. a single sign-flip can be invariant with respect to multiple sign-flips. In the context of ReLU networks one can think of sign-flip matrix Σ as encoding the activation of different paths through the network. So if there are functionally redundant paths through the network one can find a corresponding matrix Σ to create a symmetric parameter set.

A.1 Scaling Symmetries for MFVI

When performing mean-field variational inference with a diagonal Gaussian, it is common to encounter symmetrical solutions that can slow or hinder convergence, such as the low-capacity fix point depicted in Figure 2 on the right. Introducing small, fixed offsets to the mean vector and the variance parameters of the prior as proposed in Ziyin et al. (2025) helps break these symmetries by slightly shifting each dimension away from identical configurations and it rescales the regularization in different directions. As a result, the optimization is nudged toward distinct, stable modes rather than remaining stuck in symmetry axes, thereby improving the quality of the final variational approximation. This is formally demonstrated in the following analysis.

Proposition A.5. *Using an adjusted prior $p(\boldsymbol{\theta}) = N(\boldsymbol{\theta}; \boldsymbol{\theta}_0, \boldsymbol{\Sigma})$ with $\boldsymbol{\theta}_0 \sim N(0, \sigma_0)$ and $1/\Sigma_{ii} \sim U(1-\epsilon, 1+\epsilon)$ with a small ϵ breaks scaling and permutation symmetries with probability 1 and ensures better convergence of the MFVI algorithm.*

Similar to Ziyin et al. (2025), optimizing the ELBO with the adjusted prior from above results in:

$$\begin{aligned}\text{ELBO} &= \mathbb{E}_{q(\boldsymbol{\theta})}[\log p(y \mid \boldsymbol{\theta})] - D_{\text{KL}}(q(\boldsymbol{\theta}) \parallel p(\boldsymbol{\theta})) \\ &= \mathbb{E}_{q(\boldsymbol{\theta})}[\log p(y \mid \boldsymbol{\theta}) - \log q(\boldsymbol{\theta}) + \mathcal{N}(\boldsymbol{\theta}; \boldsymbol{\theta}_0, \boldsymbol{\Sigma})] \\ &= \mathbb{E}_{q(\boldsymbol{\theta})}[\log p(y \mid \boldsymbol{\theta}) - \log q(\boldsymbol{\theta}) - (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \text{constant}]\end{aligned} \tag{4}$$

This formulation is consistent with the “advanced removal” loss function from the paper by Ziyin et al. (2025) where the term $(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_0)$ encourages the solution to deviate from symmetric configurations by regularizing the difference from the shifted prior.

A.2 Scaling Symmetries for SAI

In this analysis of scaling symmetries for SAI, we focus on parameters $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ with strictly log-concave factorized (SLCF; Definition A.7) priors $p \in \mathcal{P}$ defined on $\boldsymbol{\Theta} \equiv \mathbb{R}^d$, which are symmetric around $\text{mode}(p) = \mathbf{0}$. Using strictly log-concave priors allows us to exclude degenerated cases such as piecewise zero or constant priors, which make general statements about symmetries (even) less tractable. The symmetry around zero and factorization is arguably one of the most common choices in Bayesian deep learning, but also encodes prior knowledge obtained in non-Bayesian NN literature, where weight initialization is typically set to be symmetric around zero without interdependence between the initialization of different weights. The set of SLCF priors $\mathcal{P}$ includes many common prior distributions used in the literature (Fortuin, 2022) such as the standard Gaussian, Beta, or Laplace prior. For completeness sake, we formalize the definition of SLCF priors and review the important properties.

Definition A.6 (Log-concavity)**.** A continuous random variable T is log-concavely distributed on Θ, a convex subset of $\mathbb{R}^k$, if, for any $\theta_j, \theta_i \in \Theta$ and any $\lambda \in [0, 1]$,

$$p(\lambda\theta_j + (1-\lambda)\theta_i) \geq [p(\theta_j)]^\lambda [p(\theta_i)]^{1-\lambda}. \tag{5}$$

Definition A.7 (Strict log-concave distribution)**.** A density p of a distribution is said to be strictly log-concave (SLC), if $\forall\, 0 < \lambda < 1$

$$\log p(\lambda\theta_j + (1-\lambda)\theta_i) > \lambda \log p(\theta_j) + (1-\lambda)\log p(\theta_i). \tag{6}$$

Strict log-concavity implies that the density p decreases more rapidly than a linear combination of its values.

It is straightforward to see that this property can be extended to k dimensions when the density factorizes:

Proposition A.8 (Product of independent strictly log-concave distributions)**.** *The product of independent strictly log-concave univariate distributions is again strictly log-concave.*

Proof. Assume $p(\theta_1), p(\theta_2)$ are SLC. Then it holds for $p(\theta_1, \theta_2)$:

$$\log p(\theta_1, \theta_2) = \log(p(\theta_1) \cdot p(\theta_2)) = \log p(\theta_1) + \log p(\theta_2). \tag{7}$$

Since $\log p(\theta_1)$ and $\log p(\theta_2)$ are concave functions, their sum is also concave (since the sum of concave functions is concave), and hence $p(\theta_1, \theta_2)$ is log-concave. □

Corollary A.9 (SLCF priors)**.** *SLC priors that factorize, i.e., where $p(\theta_1, \dots, \theta_p) = \prod_{j=1}^p p(\theta_j)$, are again SLC.*

Proof. This follows directly from Proposition A.8. □

Pairwise analysis Often, scaling symmetries in ReLU networks are studied using an adjacent pair of weights and without the bias term like $w^{(2)}\phi(w^{(1)}x)$. The positive symmetric scaling - excluding permutations - then corresponds to $\widetilde{w^{(1)}} := w^{(1)}c$ and $\widetilde{w^{(2)}} := w^{(2)}/c$ with $c \in \mathbb{R}_{+\backslash\{1, w^{(2)}/w^{(1)}\}}$. For this simple pairwise setting, one can show the non-invariance of the posterior assuming commonly used priors i.e. that the posterior does not have the defined scaling symmetries. This stems from the fact that the likelihood component is obviously equivalent but the prior must be different if one assumes a strictly log-concave prior like a Gaussian or Laplace which is factorized (SLCF). We show this in the following in Proposition A.10 and would like to emphasize that there is a striking similarity to min-norm solution analyses.

Proposition A.10. *SLCF priors with zero mean are not pairwise invariant w.r.t. positive scaling when excluding parameter permutation.*

Proof. We must prove that

$$p(\theta_1) \cdot p(\theta_2) \neq p(\theta_1 c) \cdot p(\theta_2/c) \tag{8}$$

for $c > 0$, $c \notin \{1, \frac{\theta_2}{\theta_1}\}$, which constitutes the point of parameter permutation that we explicitly exclude. W.l.o.g., assume $\theta_2 > \theta_1$. We differentiate between two cases, $1 < c < \frac{\theta_2}{\theta_1}$ and $c \in (0,1) \cup (\frac{\theta_2}{\theta_1}, \infty)$.

Case 1: Let $1 < c < \frac{\theta_2}{\theta_1}$. Instead of varying $c \in (1, \frac{\theta_2}{\theta_1})$, we can also use a convex combination of θ_1 and θ_2, i.e., $\exists \lambda \in (0,1)$ s.t.

$$p(\theta_1 c) \cdot p(\theta_2/c) = p(\lambda\theta_1 + (1-\lambda)\theta_2) \cdot p((1-\lambda)\theta_1 + \lambda\theta_2). \tag{9}$$

Using Definition A.7, we have

$$\begin{aligned} \log(p(\theta_1 c) \cdot p(\theta_2/c)) &= \log(p(\lambda\theta_1 + (1-\lambda)\theta_2) \cdot p((1-\lambda)\theta_1 + \lambda\theta_2)) \\ &> \lambda \log p(\theta_1) + (1-\lambda)\log p(\theta_2) + \lambda \log p(\theta_2) + (1-\lambda)\log p(\theta_1) \\ &= \log p(\theta_1) + \log p(\theta_2) = \log(p(\theta_1)p(\theta_2)) \end{aligned} \tag{10}$$

and therefore the inequality between scaled priors for $1 < c < \frac{\theta_2}{\theta_1}$.
Case 2: Let $c \in (0,1) \cup (\frac{\theta_2}{\theta_1}, \infty)$. Define the function

$$\varpi(c) = \log p(\theta_1 c) + \log p(\theta_2/c) - \log p(\theta_1) - \log p(\theta_2).$$

If we can show that $\nexists c > 0, c \notin \{1, \frac{\theta_2}{\theta_1}\}$, s.t., $\varpi(c) = 0$, then Eq. 8 is true. For $c \in (1, \frac{\theta_2}{\theta_1})$ this was already shown in Case 1. We further know that $\varpi(1) = 0$ and $\varpi(\frac{\theta_2}{\theta_1}) = 0$. Further, we have

$$\varpi'(c) = \underbrace{\frac{p'(\theta_1 c)}{p(\theta_1 c)}}_{\triangleq \nabla(\theta_1 c)} \cdot \theta_1 - \underbrace{\frac{p'\left(\theta_2/c\right)}{p\left(\theta_2/c\right)}}_{\triangleq \nabla(\theta_2/c)} \cdot \frac{\theta_2}{c^2}. \tag{11}$$

Since p is strictly log-concave, $\log p$ is strictly concave, and $\nabla(x) \triangleq \partial \log p(x)/\partial x$ is larger zero for $x < 0 = \operatorname{argmax} \log p$ and smaller zero for $x > 0$ and moreover strictly monotonically decreasing. Therefore

a) If $\theta_1 = 0$, we have $p(\theta_2) \overset{!}{=} p(\theta_2/c)$, which cannot hold for $c \neq 1$ since p is strictly log-concave.

b) If $0 < \theta_1 < \theta_2$, $\nabla(\theta_1 c) > \nabla(\theta_2 c)$ and $|\nabla(\theta_1 c)| < |\nabla(\theta_2 c)|$ as it is the derivative of a strictly concave function with 0 origin i.e. $\forall x > 0$ it holds that $\nabla(x) < 0$ and $\nabla(\cdot)$ strictly monotonically decreasing. Furthermore, for the case that $c \in (0,1)$ we have

$$\theta_1 c < \theta_1 < \theta_2 < \theta_2 c^{-1} < \theta_2 c^{-2}. \tag{12}$$

With these insights at hand we can show that for $c \in (0,1)$ Eq. 11 must always be positive:

$$\varpi'(c) = \nabla(\theta_1 c)\cdot\theta_1 - \nabla(\theta_2/c)\cdot\frac{\theta_2}{c^2} = -\underbrace{|\nabla(\theta_1 c)|}_{<|\nabla(\theta_2 c)|}\cdot\underbrace{\theta_1}_{<\theta_2 c^{-2}} + |\nabla(\theta_2/c)|\cdot\frac{\theta_2}{c^2} > 0. \tag{13}$$

Since $\varpi'(1)$ is already positive by the same argument and $\varpi(1) = 0$, we know that $\varpi(c) < 0 \, \forall c \in (0,1)$.

For $c > \frac{\theta_2}{\theta_1}$, by using $\tilde{c} \triangleq \frac{\theta_2}{\theta_1}/c \Leftrightarrow c = \frac{\theta_2}{\theta_1}/\tilde{c}$, i.e., rewriting c as a fraction $\tilde{c} < 1$ of its lower bound $\frac{\theta_2}{\theta_1}$ Eq. 11 can be rewritten as

$$\begin{aligned}\varpi'(c) &= \frac{p'(\theta_2/\tilde{c})}{p(\theta_2/\tilde{c})}\cdot\theta_1 - \frac{p'\,(\theta_1\tilde{c})}{p\,(\theta_1\tilde{c})}\cdot\frac{\theta_1^2\tilde{c}^2}{\theta_2} &(14)\\ &= \frac{\theta_1}{\theta_2}\left[\nabla(\theta_2/\tilde{c})\cdot\theta_2 - \nabla(\theta_1\tilde{c})\cdot\theta_1\tilde{c}^2\right]. &(15)\end{aligned}$$

Using $\theta_2 > \theta_1\tilde{c}^2$ we can follow the arguments of Eq. 13 to obtain

$$\begin{aligned}\varpi'(c) &= \frac{\theta_1}{\theta_2}\left[\nabla(\theta_2/\tilde{c})\cdot\theta_2 - \nabla(\theta_1\tilde{c})\cdot\theta_1\tilde{c}^2\right]. &(16)\\ &= \frac{\theta_1}{\theta_2}\underbrace{\left[-\underbrace{|\nabla(\theta_2/\tilde{c})|}_{>|\nabla(\theta_1\tilde{c})|}\cdot\underbrace{\theta_2}_{>\theta_1\tilde{c}^2} + |\nabla(\theta_1\tilde{c})|\cdot\theta_1\tilde{c}^2\right]}_{<0} < 0. &(17)\end{aligned}$$

Now a similar argument as before can be used since $\varpi(\theta_2/\theta_1) = 0$ and $\varpi'(\theta_2/\theta_1) < 0$ for all $c > \theta_2/\theta_1 \Leftrightarrow \tilde{c} < 1$. This implies that $\varpi'(c) < 0 \, \forall c > \theta_2/\theta_1$.

c) If $\theta_1 < \theta_2 < 0$, we have the same result as in b), as roles flip, in particular $\nabla(\theta_1 c) > \nabla(\theta_2 c)$ still holds but $\nabla(\cdot)$ is strictly positive valued and the negative parameters θ_1 and θ_2 would be influencing the argument on $\varpi'(\cdot)$ in the same way as the $\nabla(\cdot)$ terms in b).

d) If $\theta_1 < 0 < \theta_2$ and w.l.o.g. assume $\theta_2 > |\theta_1|$, now the same derivation as in b) is applicable. For the case $c \in (0,1)$ we have again a strictly positive $\varpi'(c)$ as $|\theta_2| > |\theta_1|$ with θ_1 having negative and θ_2 having positive sign, as well as positive $\nabla(\theta_1 c)$ and negative $\nabla(\theta_2/c)$, for which it holds that $|\nabla(\theta_1 c)| < |\nabla(\theta_2/c)|$. Analogously one can show the case of $c \in (\frac{\theta_2}{\theta_1}, \infty)$ leveraging the reparameterization of c proposed in b).

□

General ReLU networks Sadly, this pairwise result does not apply to general ReLU networks. In the following, we provide a counterexample of the invariance of the prior to positive symmetric scaling. The example is the simple ReLU net from above amended by a bias term: $w^{(2)}\phi(w^{(1)}x + b^{(1)})$. This implies $\widetilde{b^{(1)}} := b^{(1)}c$ for the scaling to be symmetric. For this case, one can construct invariant counterexamples by choosing the classically used $\mathcal{N}(0,1)$ prior for each of the three weights. One then just has to show that $\exists c \in \mathbb{R}_{+\setminus\{1,w^{(2)}/w^{(1)},w^{(2)}/b^{(1)}\}}$ and admissible parameters s.t. $p(b^{(1)})p(w^{(1)})p(w^{(2)}) = p(b^{(1)}c)p(w^{(1)}c)p(w^{(2)}/c)$ holds. We can reframe this as the setting where for some fixed weights we have to show that there exist positive roots of the function $h(c) = p(b^{(1)})p(w^{(1)})p(w^{(2)}) - p(b^{(1)}c)p(w^{(1)}c)p(w^{(2)}/c)$ that are not $\{1, w^{(2)}/w^{(1)}, w^{(2)}/b^{(1)}\}$ i.e. permutations or the trivial root one. Therefore we can simply fix the parameters to some values $w^{(1)} = 0.5$, $w^{(2)} = 0.8$, $b^{(1)} = 0.9$ and consequently have to find roots that are not in $\{1, 1.6, 0.8/0.9 \approx 0.88\}$. Calculating the roots of this function h leads to a single non-trivial positive root at ≈ 0.77 which is visualized in Figure 6. This concludes the counterexample and shows that there still exist symmetries in simple ReLU networks with biases that are not completely removed by imposing SLCF priors.

Also, we highlight the assumption about the exclusion of permutations realized via scaling in Proposition A.10 - in particular across layers. The following example for invariance in the prior holds

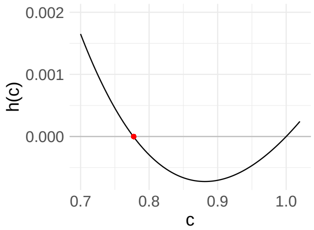

Figure 6: Visualization of the function $h(c)$ for $w^{(1)} = 0.5$, $w^{(2)} = 0.8$, $b^{(1)} = 0.9$ and standard Normal priors. The non-trivial root is highlighted in red.

even without adding biases to the model. We do this using weights in distant layers that relate as $w^{(l+1)} := w^{(1)} * c$ and $w^{(l)} := w^{(2)}/c$ with $l > 2$. Then one trivially gets an invariant factorized prior if $w^{(1)}$ and $w^{(2)}$ are scaled with c and $w^{(l)}$ and $w^{(l+1)}$ with c^{-1}.

After all, it is evident that the factorization of the prior gives rise to many invariances especially with a growing number of factors. Thus, we show by counterexample on a simple ReLU network with biases that for realistic ReLU networks there are irreducible scaling symmetries within the posterior induced by an invariance of both likelihood and the classic choice of SLCF priors. Nevertheless, it is clear that SLCF priors reduce the volume of such invariant symmetric manifolds e.g. in Proposition A.10.

Remark A.11. Proposition A.10 sheds new (positive) light on weight priors that have been discussed controversially by advocates of functional priors in Bayesian deep learning (see, e.g., Tran et al., 2022). Using SLCF helps to reduce scaling symmetries and harmonizes well in combination with ReLU networks that preclude sign-flips.

B Experimental Setup

Software Our software is implemented in Python and mainly relies on the `jax` (Bradbury et al., 2018) and `BlackJAX` (Cabezas et al., 2024) libraries. Our code is available at `https://github.com/EmanuelSommer/sampled-approx-posteriors`.

Computing Environment The experiments were conducted on two NVIDIA RTX A6000 GPUs and an AMD Ryzen™ Threadripper™ PRO 5000WX/3000WX CPU with 64 cores. For most experiments, 10 chains were sampled in parallel on the CPU, enabling efficient parallelization and allowing multiple experiments to run concurrently. For larger-scale experiments involving thousands of chains, 50 chains were sampled in parallel to maximize resource utilization.

Datasets Table 1 summarizes the benchmark datasets utilized in our experiments. If not specified otherwise, we use a 70% train, 10% validation and 20% test split as well as a fully connected model architecture with 3 hidden layers, 16 neurons per layer.

Table 1: Benchmark regression datasets overview.

Dataset	Size	Features	Source
Airfoil	1503	5	Dua & Graff (2017)
Bikesharing	17379	13	Fanaee-T (2013)
Concrete	1030	8	Yeh (1998)
Energy	768	8	Tsanas & Xifara (2012)

Performance Evaluation To quantify the quality of the posterior predictive approximation and thus the UQ capabilities of the models we use the log posterior predictive density (LPPD; Gelman et al., 2014; Wiese et al., 2023; Sommer et al., 2025) over a test set $\mathcal{D}_{\text{test}}$, defined as

$$\text{LPPD} = \frac{1}{n_{\text{test}}} \sum_{(\boldsymbol{y}^*, \boldsymbol{x}^*) \in \mathcal{D}_{\text{test}}} \log \left(\frac{1}{K \cdot S} \sum_{k=1}^{K} \sum_{s=1}^{S} p\left(\boldsymbol{y}^* | \boldsymbol{\theta}^{(k,s)}(\boldsymbol{x}^*)\right) \right). \tag{18}$$

Here, K denotes the number of chains, S the number of samples per chain, and $\boldsymbol{\theta}^{(k,s)}$ the parameters from the s-th sample of the k-th chain. Intuitively, the LPPD quantifies how well the predictive distribution aligns with the observed labels, with higher values indicating higher density coverage i.e. improved UQ performance.

In addition, we employ the root mean squared error (RMSE) for regression tasks to check for the accuracy of point predictions.

C Experimental Details and Further Analyses

C.1 Exploring the Limits of BDEs

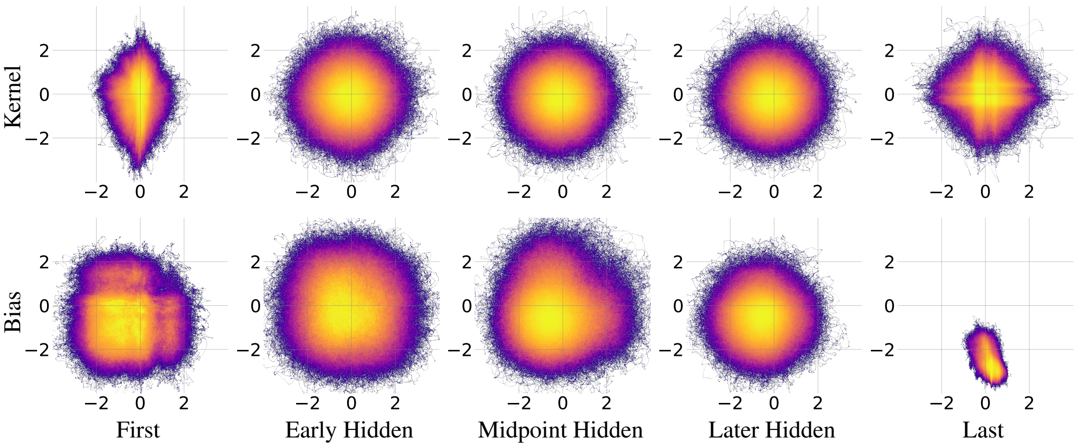

Figure 7: Bivariate empirical marginal posterior densities of a 4-hidden layer BNN fitted on UCI benchmark data (10M posterior samples obtained from 10k independent chains). The rows and columns show representative densities of randomly selected input, hidden, and output weights.

We extend the limited analysis of Sommer et al. (2024) which only considers at 12 chains to 10k chains of 1k samples each and also use a more than twice as large fully-connected neural network (4 hidden layers of 16 neurons each) to perform distributional regression. For this, we use the recently proposed MILE approach (Sommer et al., 2025) and configure it exactly as suggested by the authors. Due to the immense computational load of sampling this amount of chains and also evaluating the posterior samples (compressed the samples roughly amount to 100GB for a single experiment) we focus on one benchmark dataset, namely, `airfoil`. We have also conducted a slightly smaller experiment for the `bikesharing` dataset with 1k instead of 10k chains, which confirms the findings of the larger experiment. In our analysis, we focus on two major aspects. First, we analyze how the performance of the model develops when adding chains to the Bayesian Model Average (BMA). Second, in the spirit of Sommer et al. (2024) we take a closer look at bivariate margins of the empirical posterior derived from SAI.

The cumulative performance, which we—focusing on UQ quality—measure with the LPPD, of adding chains to the BMA obviously depends on the order in which chains are added. Thus, we consider 5 different orderings and report means and standard deviations of the cumulative LPPD over chains in Figure 5. The result suggests that with even a rather small amount of chains the performance saturates quite fast, but slowly increases further until exhibiting a very strong performance for 10k chains. Parallelizing 10-20 chains on modern hardware is very easy and comes with no considerable cost overhead over single-chain sampling. This also has very positive implications on memory requirements and inference time, rendering the approach practically feasible.

Both in Figure 1 and 7 we show representative marginal plots of the above-described experiment on the `airfoil` dataset. Thereby we only focus on within-layer margins. For completeness, we will also include a whole grid of plots that features within and across layer margins in the code repository, as a large amount of high-resolution densities hinders a smooth rendering of the manuscript. Before starting the interpretation of the visualizations one should stress that the marginal view is a limited perspective on the high-dimensional posterior of interest. From 1, which displays the empirical marginal posterior approximation for 10 and for 10k chains for two intermediate layer weights respectively, one can tell that the reported marginal mode disconnectedness in Sommer et al. (2024) is merely a result of the limited amount of sampling performed. While the more localized approximation of 10 chains does not cover the margins as continuously as the 10k chains, we know from Figure 5 that the exploration already supports good performance. Figure 7 provides a more nuanced view of how the different weights in the network act in their margins. One can observe two distinct patterns, namely differences based on the layer and the role (bias or kernel) of the weight. The distinct pattern where weights in the input and output connected layers exhibit identifiable roles forming distinct e.g. multimodal or concentrated margins, and intermediate layer weights exhibit margins that perfectly align with their (shifted) standard Gaussian prior, is supported by the proof for the functional arbitrariness of intermediate weights by Ziyin et al. (2024) for linear networks. This is also in line with Sommer et al. (2024) who provided evidence for increased exploration of the sampler for intermediate layer weights. Furthermore, the margins of the biases are centered around -1 which reflects the centering of the ReLU-induced positive hidden states. In the spirit of Fortuin et al. (2022), which derive proposals for priors from empirical weight distributions, one could consider adjusting the priors for the biases in ReLU networks accordingly in future work.

C.2 Convexifiable Network

For the experiments in Figure 8 and 9, we employed a small fully-connected network with 8 hidden neurons and a larger one with 64 hidden neurons on the `airfoil` dataset. For posterior sampling, we applied the NUTS sampler Hoffman & Gelman (2014) with pre-trained warm-starts, 100 warmup steps and 1000 posterior samples per chain across 10 chains. The Hadamard product of the weights was calculated according to the derivations in Mishkin et al. (2022); Kolb et al. (2023). We follow Kolb et al. (2023) in defining the group Hadamard product $\odot_{\mathcal{G}}$ of two vectors $\boldsymbol{v} \in \mathbb{R}^d$ and $\boldsymbol{w} \in \mathbb{R}^L$ as $\boldsymbol{v} \odot_{\mathcal{G}} \boldsymbol{w} \triangleq (\boldsymbol{v}_j w_j)_{j \in \mathcal{G}}$ for a given partition $\mathcal{G}$. We apply the group Hadamard product to our first- and second-layer-weights, where the partition $\mathcal{G}$ is given by the incoming weights of each first layer hidden neuron, resulting in the Hadamard product $\boldsymbol{u}$, as described in Section 3.3. In Figure 8 and 9, we plot the resulting densities of a random selection of components of $\boldsymbol{u}$ and observe unimodality across different sampler chains.

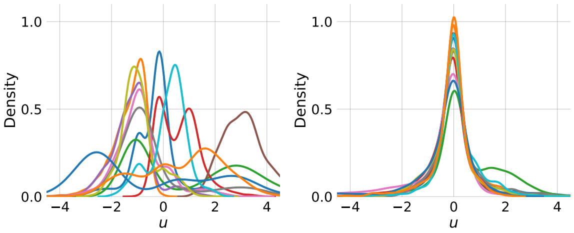

Figure 8: Marginal posterior densities of the Hadamard product of weights for a ReLU-activated neural network with 8 (left) and 64 neurons (right) in the hidden layer on the `airfoil` dataset, colored by sampler chain.

C.3 Hyperbolic Posterior Illustration

In Figure 2, rescaling symmetries and their handling in SAI and MFVI is illustrated. As a model, we use a univariate narrow Bayesian network, as described in Section 3.1. We employ the model on a dataset where the optimal solution is a regression line with slope $\hat{\beta} = 1$. In this simple setting, we know the hyperbolas that constitute the rescaling symmetry, as the product of both weights w_1 and w_2 must be equal to 1. In Figure 2 (left), we have depicted the samples obtained by running 12 warm-started NUTS sampler chains with 1000 posterior samples each in blue and samples drawn from a MFVI run in red. It is to be noted that single NUTS chains are not able to jump from one hyperbola to the opposing one. As expected, we observe that MFVI is not able to excape the low-capacity fix point at $(0, 0)$ as we increase the prior strength (decrease the prior variance), as the stochastic optimization of the ELBO cannot outweigh the drag towards the origin that is exerted by the prior and the symmetry-mirror $w_1 = w_2$ (Ziyin, 2024).

C.4 Permuted Initializations

In order to explore the entrapment of the sampler in symmetric solutions while exploring the posterior, we initialize 10 sampler chains in permuted warm-start parameter configurations that all induce the

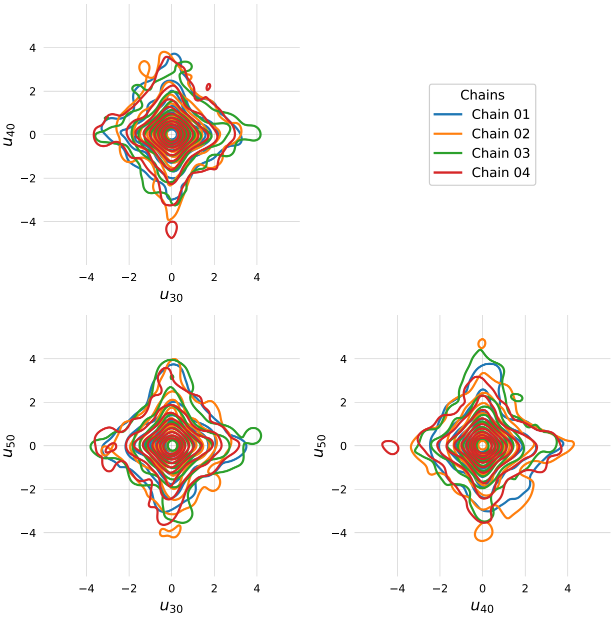

Figure 9: Bivariate marginal density of two random weight products of a ReLU-activated neural network with 64 hidden neurons on a regression task.

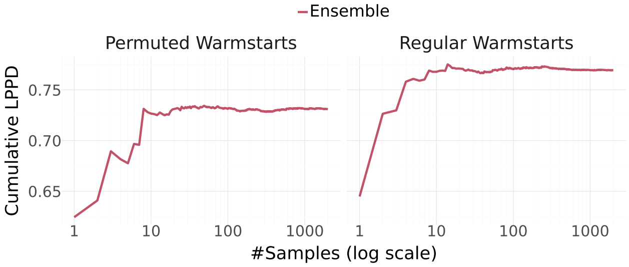

Figure 10: Chain ensemble performance for a three hidden layer fully connected network with ReLU activation on the `airfoil` dataset (zoomed-in illustration of Figure 3).

exact same functional mapping. We use a fully-connected, ReLU-actived network with 3 hidden layers with 16 neurons each. The MILE sampler (Sommer et al., 2025) is employed for posterior inference on the `airfoil` dataset, with a rather small warmup sample budget of 100 samples. In Figure 3, we observe that the individual sampler chains are nonetheless able to explore the vicinity of the permuted posterior modes and produce functionally diverse samples, thereby completely recovering from the point of symmetry. This is further supported by the observation of a considerable improvement of the cumulative LPPD when ensembling all individual chains. This points towards the sampler being able to recover functionally diverse parameters along the individual chains, even though these were initialized in a point of symmetry. In effect, the ensemble LPPD is almost on par with the one produced by a regularly warm-started BDE.

This finding is interesting for many reasons. First, it underscores the importance of proper chain initialization with parameter states that already induce a high model likelihood, no matter the functional diversity induced by this parameter state. Moving towards the typical set, the sampler benefits from its stochastic components, whereas samples from the typical set then quickly drive up the LPPD. The cumulative LPPD then increases more slowly, as every new sample contributes less and less to the Bayesian model average that is formed to approximate the posterior predictive distribution with a finite sample size. Eventually, the chain performances and the ensemble performance converge

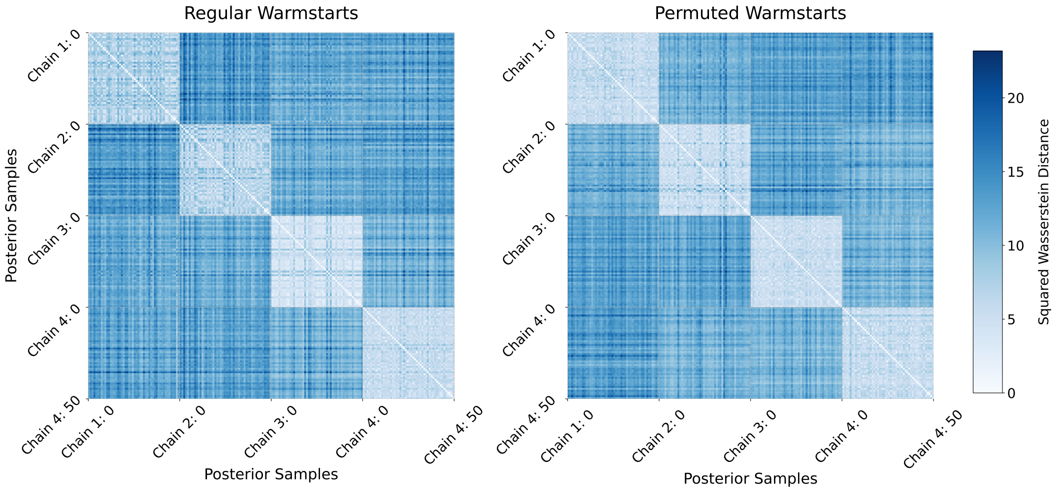

Figure 11: Squared Wasserstein-2 distance of the predictive distribution induced by individual samples from four different sampler chains on the `airfoil` dataset.

towards their respective local approximation of the posterior given the chain or the whole ensemble of chains, which is plotted separately in Figure 10.

In an alternative illustration, we have depicted the squared Wasserstein-2 distances of the predictive distributions $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}^{(s)})$ implied by every sample $\boldsymbol{\theta}^{(s)}$ from a chain across 4 different chains and 50 samples in Figure 11. It is clearly visible that the within-chain distance of the implied predictive distributions is lower compared to the between-chain distance with very similar patterns for both the chains started using regular warm-starts and permuted warm-starts.

The result also carries the possible practical implication that a diverse set of functionally different warm-starts as chain initializers might not be crucial for a well-performing Bayesian model average. This could prove helpful for large architectures, where obtaining warm-start parameters is increasingly costly.

C.5 Computing Wasserstein Distances for Uncountable Symmetries

In order to check whether distributions are significantly different, we first estimate the correlation between two samples using $\hat{\rho} = \mathrm{Cor}(\boldsymbol{\mu}^{(s)}, \boldsymbol{\mu}^{(\tilde{s})})$. For every possible combination $s, \tilde{s} \in [S], s \neq \tilde{s}$, we then compute

$$\mu_{D,i} = \mu_i^{(s)} - \mu_i^{(\tilde{s})} \tag{19}$$

and calculate the mean over all observations, which should have distribution

$$\bar{\mu}_D := n^{-1} \sum_{i=1}^{n} \mu_{D,i} \sim \mathcal{N}(0, n^{-1}\sigma_D^2), \tag{20}$$

where

$$\sigma_D^2 = (\sigma^2)^{(s)} + (\sigma^2)^{(\tilde{s})} - 2\rho\sigma^{(\tilde{s})}\sigma^{(\tilde{s})}. \tag{21}$$

Our test statistic can then be computed as

$$Z = \bar{\mu}_D / (\sqrt{n}\sigma_D) \tag{22}$$

and compared against the 95% standard normal quantile (1.96) to highlight values where one would reject the hypothesis that the estimated distributions are equivalent.

C.6 Influence of the Prior Strength

As discussed in the main sections, typical SAI applications fix a common prior variance across the network. To the best of our knowledge, it is not common to increase variances with increased network depth as analogies to regularized optimization might imply. In order to provide empirical evidence that a constant variance irrespective of the network depth works well, we perform a small benchmark, where Figure 12 in depicts the resulting LPPD and RMSE performances, confirming that a standard Gaussian distribution is a well-working choice with little changes when altering the network size. An extend discussion of prior choice follows below.

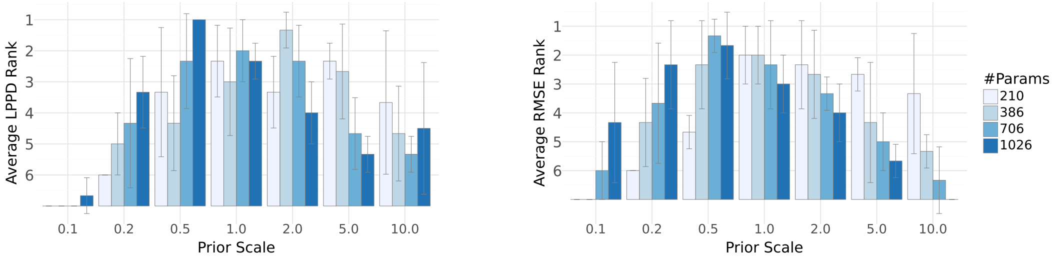

Figure 12: Average LPPD and RMSE ranks (± standard deviation across 3 train-test splits) of varying scale parameters for the zero-centered Gaussian priors. The different model sizes are differentiated by color. Better ranks (i.e. 1>2) correspond to better performance of the prior scale for the given model size.

The standard isotropic Gaussian remains the most widely used prior in Bayesian neural networks. It is the default in industry standard software like `fortuna` (Detommaso et al.) and is shown to lead to high performing results in works like Sommer et al. (2025). While Fortuin et al. (2022) argue that heavy tailed priors are superior for fully-connected BNNs, the isotropic assumption is rarely challenged. As discussed in Section 4 and illustrated in Figure 2, the scale and thus the prior's strength is related to the likelihood of scaling symmetries to appear in the posterior. At first glance it might seem intuitive that for changing dimensionality of the problem the prior's pull towards the origin has to be adjusted based on the dimension. While the optimization literature would suggest a looser regularization for increased overparameterization to be sufficient, the opposite argument can be made for smaller scales in light of taming the functional explosion of deep networks. To grasp the effects of increased dimensionality and prior scale we have conducted roughly 100 experiments assessing the practical performance of SAI on 3 distinct tasks. For three benchmark regression datasets (`airfoil`, `bikesharing`, `energy`) we fitted four fully-connected BNNs with 1 up to 4 hidden layers of 16 hidden neurons. We repeated this for various prior scales of a zero-centered isotropic Gaussian and report the average and standard deviation of RMSE and LPPD ranks in Figure 12. The results confirm that regardless of the number of model parameters the isotropic standard Gaussian is a robust choice for attaining good performance. Also, small deviations do not seem to harm BNN performance, but with both much smaller and much larger variance, clear deterioration in performance is visible, which indicates that neither a systematic adjustment of the prior scale proportional to the dimensionality nor to the inverse direction is required.

Another naïve approach to not worry about the scale parameter would be to leverage common neural network initialization strategies to set the prior scale layerwise and depending on the architecture. One common representative of such initializations is the Glorot initialization (Glorot & Bengio, 2010) which specifies the scale parameter as

$$\sigma = \sqrt{2/(n_{\text{in}} + n_{\text{out}})}. \tag{23}$$

In the above considered architectures most kernels have $n_{\text{in}} = n_{\text{out}} = 16$ leading to a scale of 0.25 rather constantly over the network. As we can see in Figure 12 this scale level is however not suboptimal, as the scale is too small. This is the case for most popular network initialization schemes. Thus in this setting we can not find evidence that deviating from the isotropic prior assumption in favor of layerwise adjustments in the spirit of network initializations is beneficial in terms of model performance.

C.7 UCI Benchmark

Table 2: Mean RMSE (↓) and LPPD (↑) results (± standard deviation across 3 train-test splits) for a 3 hidden-layer BNN on regression tasks. Brackets denote ensemble members/chains.

	Dataset	**Laplace**	**MFVI**	**DE (10)**	**BDE (1)**	**BDE (10)**
LPPD	Airfoil	-1.056 ± 0.003	-0.975 ± 0.004	-0.293 ± 0.096	0.016 ± 0.293	$\mathbf{0.665 \pm 0.062}$
	Bikesharing	-1.046 ± 0.001	-0.990 ± 0.005	-0.223 ± 0.181	-0.060 ± 0.096	$\mathbf{0.226 \pm 0.043}$
	Concrete	-1.131 ± 0.036	-0.998 ± 0.007	-0.510 ± 0.189	0.042 ± 0.056	$\mathbf{0.080 \pm 0.061}$
	Energy	-1.046 ± 0.004	-0.945 ± 0.002	1.561 ± 0.101	1.947 ± 0.047	$\mathbf{2.204 \pm 0.024}$
RMSE	Airfoil	0.237 ± 0.013	0.276 ± 0.009	0.269 ± 0.016	0.184 ± 0.016	$\mathbf{0.152 \pm 0.014}$
	Bikesharing	0.252 ± 0.006	0.318 ± 0.018	0.253 ± 0.015	0.262 ± 0.018	$\mathbf{0.229 \pm 0.016}$
	Concrete	0.482 ± 0.100	0.350 ± 0.025	0.297 ± 0.032	0.270 ± 0.034	$\mathbf{0.258 \pm 0.037}$
	Energy	0.065 ± 0.008	0.126 ± 0.007	0.050 ± 0.001	0.041 ± 0.003	$\mathbf{0.032 \pm 0.002}$

For the UCI benchmark presented in Section 5.3 and Table 2, we fit classical mean regression to the different tasks corresponding to the datasets described in Table 1. In the process, we always use a fully-connected feed-forward neural network with 3 hidden layers of size 16 each resulting in about 700 total model parameters. If sampling from the posterior is done we use 1000 samples per ensemble member (chain). We describe the configuration of the employed methods one by one:

- For the **Laplace approximation** (LA), we utilize a JAX-based implementation to first train MAP solutions using the Adam optimizer with decoupled weight decay (Loshchilov & Hutter, 2019) for 10 000 epochs with a learning rate of 0.005 to then carry out last-layer LA with a generalized Gauss Newton Hessian approximation and closed-form predictive approximation as detailed in Daxberger et al. (2021a). The variance of the predictive distribution is calculated according to Daxberger et al. (2021a) with a small additional noise variance term.
- For **mean-field variational inference** (MFVI), we utilize a Gaussian posterior approximation with independence assumption. We optimize the evidence lower bound (ELBO) for 5000 epochs with the Adam optimizer and a learning rate of 0.005. The variance of the predictive distribution is calculated as the variance over the the predictions made with 100 samples from the fitted approximate posterior with a small additional observation noise term.
- As the recently proposed **Microcanonical Langevin Ensemble** (MILE) approach provides both an optimized **Deep Ensemble** (DE) and a **Bayesian Deep Ensemble** (BDE), we follow the suggested setup of Sommer et al. (2025) i.e. the DE is optimized with the Adam optimizer with decoupled weight decay with (memberwise) early stopping and the sampling then uses the proposed auto-tuning strategy of MILE comprising 50k steps before then providing 1k samples (after the thinning of 10k samples).

Each method is evaluated using three distinct train-test splits to assess the robustness of its performance.

D Convexifiable Network from a Bayesian Perspective

Pilanci & Ergen (2020) showed that the the optimal value(s) of the optimization problem of a one-hidden-layer ReLU-activated fully-connected neural network without bias, which is regularized with weight decay, can be recovered from an equivalent reformulation of the optimization problem as a group-ℓ_1-regularized optimization problem. This result relies on "enumerating" all possible activation states of a single neuron in the hidden layer on a fixed dataset $\boldsymbol{X} \in \mathbb{R}^{n \times p}$, namely, $\mathcal{D}_X = \{D = \text{diag}(\mathbb{1}(\boldsymbol{X}\boldsymbol{v} \geq \mathbf{0}) : \boldsymbol{v} \in \mathbb{R}^p)\}$. Since learning with all possible activation patterns of such a network on a fixed dataset is computationally infeasible in most cases, Mishkin et al. (2022) propose a sub-sampling approach, where they only sample a subset of all possible activation patterns, $\tilde{D}$, and are still able to find optimal points to the optimization problem using proximal-gradient solvers. It is notable that this is analogous to the posterior sampling process in a Bayesian neural network framework. In our context, the computation of ReLU neurons is inherently tied to the posterior sampling process, implicitly generating their associated activation patterns.

From this perspective, given S posterior samples, we can express $\tilde{D}$ as

$$\tilde{D} = \left\{\operatorname{diag}(\mathbb{1}(\boldsymbol{W}^{(1)}\boldsymbol{x}_{(s)} \geq \mathbf{0}),\, \forall \ell \in [L])\right\}_{s=1}^{S}.$$

Compared to the experiment in Mishkin et al. (2022), where the size of $\tilde{D}$ is limited to 100, our approach allows for generating a significantly larger number of activation patterns, resulting in greater expressiveness. In the next step, we apply the group Hadamard parameterization map (GHPP, Kolb et al., 2023) described above and denoted by $\mathcal{K}$ to the sampled weights that implicitly include numerous possible ReLU activation patterns.

Lemma D.1. *Let $f(\boldsymbol{W}_1, \boldsymbol{w}_2)$ be the log posterior up to constants and use independent zero mean Gaussian priors with variance τ^2 for every weight. Further, define $\boldsymbol{X}$ as vertically stacked observations $\boldsymbol{x}_i$ and $\boldsymbol{y}$ as vector of targets for a regression task. Then $f(\boldsymbol{W}_1, \boldsymbol{w}_2)$ majorizes $g(u)$ under the map $\mathcal{K}(\boldsymbol{W}_1^\top, \boldsymbol{w}_2^\top) = \boldsymbol{W}_1^\top \boldsymbol{w}_2^\top = \boldsymbol{u}$.*

Proof.

$$\begin{aligned}
f(\boldsymbol{W}_1^\top, \boldsymbol{w}_2^\top) &= \frac{1}{2\sigma^2}\|\sum_{D_i\in\tilde{D}} \boldsymbol{D}_i\boldsymbol{X}\boldsymbol{W}_{1i}^\top\boldsymbol{w}_{2i}^\top - \boldsymbol{y}\|_2^2 + \frac{1}{2\tau^2}\sum_{D_i\in\tilde{D}} \|\boldsymbol{W}_{1i}^\top\|_2^2 + |\boldsymbol{w}_{2i}^\top|^2 \\
&\overset{\text{(AM-GM)}}{\geq} \frac{1}{2\sigma^2}\|\sum_{D_i\in\tilde{D}} \boldsymbol{D}_i\boldsymbol{X}\boldsymbol{W}_{1i}^\top\boldsymbol{w}_{2i}^\top - \boldsymbol{y}\|_2^2 + \frac{1}{\tau^2}\sum_{D_i\in\tilde{D}} \|\boldsymbol{W}_{1i}^\top\|_2 \cdot |\boldsymbol{w}_{2i}^\top| \\
&= \frac{1}{2\sigma^2}\|\sum_{D_i\in\tilde{D}} \boldsymbol{D}_i\boldsymbol{X}\boldsymbol{W}_{1i}^\top\boldsymbol{w}_{2i}^\top - \boldsymbol{y}\|_2^2 + \frac{1}{\tau^2}\sum_{D_i\in\tilde{D}} \|\boldsymbol{W}_{1i}^\top\boldsymbol{w}_{2i}^\top\|_2 \\
&= \frac{1}{2\sigma^2}\|\sum_{D_i\in\tilde{D}} \boldsymbol{D}_i\boldsymbol{X}\boldsymbol{u}_i - \boldsymbol{y}\|_2^2 + \frac{1}{\tau^2}\sum_{D_i\in\tilde{D}} \|\boldsymbol{u}_i\|_2 \\
&= g(\boldsymbol{u}),
\end{aligned}$$

where equality in the AM-GM inequality holds if and only if $\|W_{1i}\|_2^2 = |w_{2i}|^2$. □

As a direct consequence, we have the following:

Corollary D.2. *The distribution of* $\mathbf{u}$ *(up to permutations) is unimodal.*

This results from Lemma D.1 and the fact that we can apply Theorem 2.10 in Kolb et al. (2023), which shows that f and g must share the same global and local optimal values.

E Results for Univariate Networks

E.1 Derivations of Results from Section 3.1

We start by analyzing the negative log density of the unconstrained posterior:

$$-\log p(\mathbf{w}|\mathbf{y},\mathbf{x}) \propto \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \textstyle\prod_{l=1}^{L} w^{(l)}x_i)^2 + \frac{1}{2\tau^2}\sum_{l=1}^{L} {w^{(l)}}^2. \tag{24}$$

Using the AM-GM inequality we obtain

$$\begin{aligned}
-\log p(\mathbf{w}|\mathbf{y},\mathbf{x}) &\propto \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \textstyle\prod_{l=1}^{L} w^{(l)}x_i)^2 + \frac{1}{2\tau^2}\sum_{l=1}^{L}(w^{(l)})^2 & (25)\\
&\geq \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \textstyle\prod_{l=1}^{L} w^{(l)}x_i)^2 + \frac{1}{2\tau^2}L\prod_{l=1}^{L}|w^{(l)}| & (26)\\
&:= \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \beta x_i)^2 + \frac{1}{2\tau^2}L|\beta|^{2/L} =: g_1(\beta) + g_2(\beta) =: g(\beta) & (27)
\end{aligned}$$

In particular, applying Theorem 2.10 from Kolb et al. (2023), we know that $-\log p(\mathbf{w}|\mathbf{y}, \mathbf{x})$ and $g(\beta)$ have matching local and global minima. Multiplying by -1 and exponentiating both terms does not change this property.

2-Layer Networks For $L = 2$, it is easy to see that $g(\beta)$ is convex as g corresponds to the Lasso problem. As a consequence, we know that there is one global posterior mode for $\mathbf{w}$, which coincides with the maximum of $-g(\beta)$, given by

$$\hat{\beta} = \begin{cases} 0 & \text{if } |\sum_{i=1}^n x_i y_i| \leq 1/\tau^2 \\ \text{sgn}(\sum_{i=1}^n x_i y_i) \cdot \left(|\sum_{i=1}^n x_i y_i| - 1/\tau^2 \right) & \text{if } |\sum_{i=1}^n x_i y_i| > 1/\tau^2 \end{cases} .$$

The posterior of β does not have an analytical form but can be simulated from by using a Laplace prior. However, since identity in equation 26 only holds for $(w^{(l)})^2 \equiv |\beta|^{2/L}$, the posterior density of $\mathbf{w}$ only needs to coincide in the mode and must be unimodal due to the matching optima theorem, but is not necessarily convex.

Networks with $L > 2$ For $L > 2$, $g(\beta)$ is not convex anymore and hence the log posterior is not necessarily unimodal anymore. Due to the exchangeability of all $w^{(l)}$, we, however, know that for $\hat{\beta} = \arg\min_{\beta} g(\beta)$, one mode of $p(\mathbf{w}|\mathbf{y}, \mathbf{x})$ is located at $w^{(l)} = (\hat{\beta})^{1/L}$, for $l \in [L]$. When fixing n and starting to increase the network's overparametrization, i.e., increasing L, we observe a trade-off between prior and likelihood. Assuming $r_i^2 \coloneqq (y_i - \beta x_i)^2 \lll \infty$ and $\partial\beta/\partial L = 0$, we have that $g_1(\beta)/g_2(\beta) \to 0$ as $L \to \infty$. This is because the likelihood term stays constant for increasing L while $|\beta|^{2/L}$ approaches 1 in the limit. In other words, for increasing L, the likelihood term will have less and less influence on the posterior while the prior will become more and more influential.