EMPIRICAL PRIORS FOR BAYESIAN NEURAL NETWORKS VIA WEIGHT PRUNING SENSITIVITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Bayesian neural networks provide a principled framework for uncertainty estimation, yet their performance is highly dependent on the choice of prior over network parameters. Designing informative priors for modern neural networks remains challenging because their large parameter spaces make it difficult to characterize the functional importance of individual parameters in a computationally tractable way. In this work, we propose an empirical prior for Bayesian neural networks that uses continuous pruning sensitivity to allocate Gaussian prior precision across individual parameters. Through extensive experiments across architectures, datasets, and prior configurations, we demonstrate that our approach achieves consistent performance gains across model families and dataset scales, suggesting that network pruning can be used not only for model compression, but also as a useful tool for designing informative priors in Bayesian neural networks.

1 INTRODUCTION

Bayesian neural networks (BNNs) provide a principled framework for representing uncertainty by maintaining a posterior distribution over network parameters rather than relying on a single point estimate. This Bayesian approach can improve predictive uncertainty quantification and calibration, making BNNs attractive when reliable uncertainty estimates are required (Kendall & Gal, 2017; Chakraborti et al., 2025; Wang et al., 2025; He et al., 2026). However, the behavior of a BNN depends not only on the posterior inference method but also critically on the choice of prior (Wenzel et al., 2020a; Krishnan et al., 2020; Fortuin et al., 2022). Specifying meaningful priors becomes particularly challenging for modern neural networks, where millions of parameters make it difficult to express useful prior knowledge at the level of individual weights. Consequently, simple standard Gaussian priors remain a common choice.

Existing work has explored two complementary directions. Flexible priors, such as spike-and-slab and horseshoe priors (Ghosh et al., 2018; 2019; Jantre et al., 2024), allow individual parameters to undergo different degrees of regularization, but can introduce additional complexity into posterior inference and optimization, and have primarily been studied for sparsity and model compression. Another line of work constructs empirical priors, where prior information is derived from pretrained networks or directly from the training data. Such priors have been constructed from pretrained weights for prior specification and posterior initialization, or empirical statistics of trained network (Atanov et al., 2019; Krishnan et al., 2020; Fortuin et al., 2022). Post-hoc Bayesian approaches estimate prior precision from the training data through marginal-likelihood optimization around a pretrained solution (Daxberger et al., 2021). While these methods demonstrate the value of empirical approaches to prior specification, how the functional importance of individual pretrained weights should determine parameter-wise prior precision remains underexplored, leaving limited guidance on where posterior flexibility should be retained around the pretrained solution.

A closely related source of parameter-wise information is network pruning, which identifies redundant parameters by estimating the effect of their removal on network performance. Classical pruning methods use local loss sensitivity to measure parameter importance and compress neural networks (LeCun et al., 1989; Hassibi & Stork, 1992; Molchanov et al., 2019). Subsequent studies, including the lottery ticket hypothesis further revealed substantial parameter-level structure in dense networks, while pruning-at-initialization methods showed the parameter importance can be estimated even before training (Frankle & Carbin, 2018; Lee et al., 2018; Wang et al., 2020). In BNNs, pruning has

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

primarily been studied for model compression based on posterior statistics, sparsity, or marginal-likelihood optimization (Graves, 2011; Blundell et al., 2015b; Ghosh et al., 2019; Sun et al., 2022; Dhahri et al., 2024). Beyond model compression, pruning has also been interpreted as a regularization mechanism related to noise injection (Bartoldson et al., 2020). Nevertheless, pruning criteria have rarely been considered as a source of parameter-wise information for allocating prior precision in BNNs.

This motivates the central question of this work: *given a pretrained neural network, how should prior precision be allocated across its individual weights?* We bridge network pruning and weight prior design by using the loss sensitivity to the removal of each weight as empirical information for prior construction. Rather than using this sensitivity solely to determine which parameters to remove, we use it to rank pretrained weights and allocate prior precision across individual parameters. This perspective turns pruning sensitivity into a mechanism for controlling where posterior flexibility is retained. Our main contributions are summarized as follows:

- We reinterpret network pruning as a source of information for empirical prior construction, using the loss sensitivity to individual weight removal to allocate Gaussian prior precision across network parameters. To the best of our knowledge, this is the first attempt to leverage network pruning for parameter specific prior design in Bayesian neural networks;
- We extensively evaluate the proposed prior on datasets of varying scale and across both convolutional neural network (CNN) and vision transformer (ViT) architectures, demonstrating its consistent effectiveness across data scales and architectural families; and
- We systematically analyze how different prior precision allocations shape the posterior and its predictive behavior, providing insight into the underlying mechanisms.

2 Problem Formulation

2.1 Variational Inference with a Pretrained Solution

Given a training dataset $\mathcal{D} = \{(x_n, y_n)\}_{n=1}^N$, the Bayesian predictive distribution marginalizes the likelihood over the posterior distribution of the network weights,

$$p(y \mid x, \mathcal{D}) = \int p(y \mid x, w) p(w \mid \mathcal{D}) dw. \tag{1}$$

For modern neural networks, both the posterior $p(w \mid \mathcal{D})$ and the integral are generally intractable. We therefore approximate the posterior using mean-field variational inference (MFVI) with a fully factorized Gaussian, $q_\phi(w) = \Pi_{i=1}^d \mathcal{N}(w_i \mid \mu_i, \sigma_i^2)$, where $\phi = \{(\mu_i, \sigma_i)\}_{i=1}^d$ denotes the variational parameters (Blundell et al., 2015a). MFVI optimizes the variational parameters by maximizing the evidence lower bound (ELBO),

$$\boldsymbol{\phi}^* = \arg\max_{\boldsymbol{\phi}} \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{w})} \left[\log p(\mathcal{D} \mid \mathbf{w})\right] - D_{\mathrm{KL}} \left[q_{\boldsymbol{\phi}}(\mathbf{w}) \| p(\mathbf{w})\right]. \tag{2}$$

We assume that a deterministic neural network (DNN) with the same architecture has been trained on $\mathcal{D}$, yielding a pretrained solution $\mathbf{w}^*$. We leverage this solution as the center of a fully factorized Gaussian prior,

$$p(\mathbf{w}) \equiv p(\mathbf{w} \mid \mathbf{w}^*, \boldsymbol{\Lambda}) = \mathcal{N}\left(\mathbf{w} \mid \mathbf{w}^*, \boldsymbol{\Lambda}^{-1}\right), \qquad \boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, ..., \lambda_d). \tag{3}$$

Since $\mathbf{w}^*$ is obtained from training data, we refer to this construction as an empirical prior. Fixing the prior mean at $\mathbf{w}^*$ leaves the parameter-wise precisions $\{\lambda_i\}_{i=1}^d$ to determine how strongly the posterior is constrained along different parameter directions. This raises the central problem of how these precisions should be allocated across individual pretrained weights.

2.2 Prior Precision and Posterior Flexibility

We first examine how the prior precision controls the variational posterior around the pretrained solution. Let $\mathcal{L}_{\mathrm{NLL}}(\mathbf{w}) = -\log p(\mathcal{D} \mid \mathbf{w})$ denote the negative log-likelihood (NLL), and let $\mathbf{e}_i$ denote the i-th standard basis vector. We consider a coordinate-wise perturbation around the pretrained solution, $\mathbf{w} = \mathbf{w}^* + \delta_i \mathbf{e}_i$, while keeping all other parameters fixed. Using a local quadratic

approximation,

$$\mathcal{L}_{\mathrm{NLL}}(\mathbf{w}^* + \delta_i \mathbf{e}_i) \approx \mathcal{L}_{\mathrm{NLL}}(\mathbf{w}^*) + g_i \delta_i + \frac{1}{2} H_{ii} \delta_i^2, \tag{4}$$

where g_i and $H_{ii} > 0$ denote the corresponding gradient and diagonal Hessian element, respectively. Since pretrained DNNs are generally obtained through regularized training, often corresponding to maximum a posteriori (MAP) estimation, the NLL gradient g_i at $\mathbf{w}^*$ need not vanish.

For the Gaussian prior centered at w_i^* with precision λ_i, minimizing the local negative ELBO under Eq. 4 with respect to the variational parameters yields

$$\mu_i = w_i^* - \frac{g_i}{H_{ii} + \lambda_i}, \qquad \sigma_i^2 = \frac{1}{H_{ii} + \lambda_i}. \tag{5}$$

Equation 5 shows that the prior precision directly controls posterior flexibility around the pretrained solution. Increasing λ_i suppresses both the displacement of the variational mean from w_i^* and the posterior variance, whereas decreasing λ_i allows greater adaptation and uncertainty along the corresponding parameter direction.

A natural approach to determining the prior precision is empirical Bayes, which selects prior precisions by marginal likelihood maximization (Immer et al., 2021; Daxberger et al., 2021). For the i-th parameter, while fixing all remaining parameters at their pretrained values, we consider

$$\lambda_i^* = \arg\max_{\lambda_i > 0} p(\mathcal{D} \mid \lambda_i, w_i^*) = \arg\max_{\lambda_i > 0} \int p(\mathcal{D} \mid w_i) p(w_i \mid w_i^*, \lambda_i)\, dw_i.$$

Under the same local quadratic approximation, maximizing the marginal likelihood yields

$$\lambda_i^* = \frac{H_{ii}^2}{g_i^2 - H_{ii}}, \qquad g_i^2 > H_{ii}. \tag{6}$$

For $g_i^2 \leq H_{ii}$, the objective is maximized in the limit $\lambda_i \to \infty$, for which $\mu_i \to w_i^*$ and $\sigma_i^2 \to 0$. Thus, local evidence optimization yields a finite precision only when $g_i^2 > H_{ii}$. In practice, replacing this limiting solution with a finite upper precision still strongly constrains the posterior around the pretrained solution, suppressing posterior variation along the corresponding parameter directions and potentially limiting the diversity available for Bayesian model averaging (BMA). Detailed derivations of Eqs. 5 and 6 are provided in Appendix A.1.

3 Empirical Prior Using Network Pruning Sensitivity

3.1 Loss Sensitivity to Weight Pruning

A common way to obtain parameter-wise information from a pretrained network is to probe how its loss responds to perturbations of individual weights. Network pruning provides a natural probe for this purpose, as classical pruning methods assess parameter importance through the loss sensitivity to weight removal (LeCun et al., 1989; Hassibi & Stork, 1992; Molchanov et al., 2019). We therefore use the loss sensitivity to removing each pretrained weight as parameter-wise empirical information for prior construction.

For a pretrained parameter vector $\mathbf{w}^*$, pruning the i-th weight while keeping all other parameters fixed produces $\mathbf{w}^* - w_i^* \mathbf{e}_i$. This corresponds to the coordinate-wise perturbation $\delta_i = -w_i^*$ considered in Eq. 4, yielding

$$\mathcal{L}_{\mathrm{NLL}}\left(\mathbf{w}^* - w_i^* \mathbf{e}_i\right) \approx \mathcal{L}_{\mathrm{NLL}}(\mathbf{w}^*) - g_i w_i^* + \frac{1}{2} H_{ii} (w_i^*)^2. \tag{7}$$

We denote the corresponding change in NLL by $\Delta\mathcal{L}_i$. From Eq. 7, it can be approximated as $\Delta\mathcal{L}_i \approx -g_i w_i^* + \frac{1}{2} H_{ii} (w_i^*)^2$. Computing the Hessian diagonal exactly is expensive for modern neural networks. We therefore approximate the local curvature using the diagonal generalized Gauss–Newton (GGN) matrix, $H_{ii} \approx G_{ii}$, where G_{ii} denotes the corresponding diagonal GGN entry. We then define the pruning sensitivity score as the magnitude of the approximated NLL change,

$$s_i = |\Delta\mathcal{L}_i| \approx \left| -g_i w_i^* + \frac{1}{2} G_{ii} (w_i^*)^2 \right|. \tag{8}$$

A larger s_i therefore indicates that removing the corresponding pretrained weight has a greater effect on the NLL.

3.2 SENSITIVITY-GUIDED PRECISION ORDERING

We next use the pruning sensitivity scores to determine how strongly each parameter should be constrained around the pretrained solution. As shown in Eq. 5, a larger prior precision λ_i imposes stronger shrinkage toward w_i^*. To determine the relative prior precision across parameters, we adopt the principle of selective shrinkage, under which parameters that are important for explaining the data are allowed to escape strong shrinkage, while more weakly supported parameters are constrained more strongly (Mitchell & Beauchamp, 1988; Carvalho et al., 2009; Polson & Scott, 2011). We apply this principle around the pretrained solution using pruning sensitivity as empirical information about individual weights.

$$s_i > s_j \quad \Longrightarrow \quad \lambda_i \leq \lambda_j. \tag{9}$$

Thus, more sensitive weights are assigned lower prior precision, whereas less sensitive weights receive higher prior precision.

3.3 SENSITIVITY-TO-PRECISION MAPPING

The ordering in Eq. 9 specifies the relative allocation of prior precision, but does not determine its numerical values. Directly mapping the raw sensitivity scores to prior precision is undesirable because their magnitudes can vary substantially across parameters and architectures and may be sensitive to estimation variability. We therefore use their relative ordering through the empirical cumulative distribution function (ECDF),

$$\widehat{F}_s(x) = \frac{1}{d}\sum_{j=1}^{d} \mathbb{I}(s_j \leq x), \qquad u_i = \widehat{F}_s(s_i) \in (0, 1],$$

where d is the number of weights and $\mathbb{I}(\cdot)$ denotes the indicator function. The quantity u_i represents the normalized sensitivity rank of the i-th weight, preserving the relative ordering of the sensitivity scores independently of their absolute scale.

Any monotonically decreasing mapping from u_i to prior precision can satisfy the ordering in Eq. 9. We use affine interpolation between two reference precisions, $\lambda_{\min}$ and $\lambda_{\max}$, as a simple bounded mapping that preserves this ordering without introducing an additional shape hyperparameter:

$$\lambda_i = u_i\lambda_{\min} + (1 - u_i)\lambda_{\max}. \tag{10}$$

Since u_i increases with s_i and $\lambda_{\min} < \lambda_{\max}$, Eq. 10 satisfies the sensitivity–precision ordering in Eq. 9. The two reference precisions determine the absolute scale of the allocation, which we specify next.

3.4 REFERENCE PRECISION SCALES

The mapping in Eq. 10 determines the relative allocation of prior precision. To set the absolute precision scale, we use two reference values, $\lambda_{\min}$ and $\lambda_{\max}$. To determine a finite upper reference precision, we examine how the KL term in the MFVI objective constrains the variational mean. For the Gaussian posterior and prior, the KL divergence is

$$D_{\mathrm{KL}}\left[\mathcal{N}(\mu_i, \sigma_i^2) \| \mathcal{N}(w_i^*, \lambda_i^{-1})\right] = \frac{1}{2}\left[\lambda_i\left(\sigma_i^2 + (\mu_i - w_i^*)^2\right) - 1 - \log(\lambda_i\sigma_i^2)\right]. \tag{11}$$

Since the KL term in the negative ELBO is scaled by $1/N$, its contribution to the variational mean gradient is $\nabla_{\mu_i} D_{\mathrm{KL}}/N = \lambda_i(\mu_i - w_i^*)/N$. Considering only this contribution, a gradient step with learning rate η_t gives $\mu_i^{t+1} - w_i^* = (1 - \eta_t\lambda_i/N)(\mu_i^t - w_i^*)$. We use $0 \leq 1 - \eta_t\lambda_i/N \leq 1$ as a conservative condition on this direct KL update. Applying the condition at $\eta_{\max}$ gives $\lambda_i \leq N/\eta_{\max}$; we therefore set $\lambda_{\max} = N/\eta_{\max}$ as the upper reference precision. We set $\lambda_{\min} = 1$, corresponding to the unit-variance Gaussian prior proposed by Krishnan et al. (2020). Eq. 10 then assigns each weight a finite prior precision according to its normalized sensitivity rank.

In this way, network pruning provides an empirical basis for prior design in BNNs by allowing weights with high pruning sensitivity to adapt more freely to the likelihood while constraining less sensitive weights around $\mathbf{w}^*$.

Table 1: Predictive performance across datasets and architectures. Results are mean ± standard deviation over five seeds. We use SGD with $\eta_{\max} = 0.1$ and cosine annealing for all BNNs. The best result for each metric is shown in green and the second-best is yellow.

		CIFAR-10			ImageNet-100			ImageNet-1K		
	Method	Acc. ↑	NLL ↓	ECE ↓	Acc. ↑	NLL ↓	ECE ↓	Acc. ↑	NLL ↓	ECE ↓
CNN	$\lambda_{\min}\mathbf{I}$	.883±.004	.390±.013	.083±.006	.682±.007	1.313±.015	.179±.006	.670±.003	1.450±.016	.119±.003
	$\lambda_{\max}\mathbf{I}$	.915±.001	.338±.004	.050±.001	.842±.001	.609±.001	.034±.002	.799±.001	.770±.002	.032±.002
	EB-Global	.829±.013	.631±.018	.172±.017	.669±.008	1.427±.030	.219±.007	.668±.003	1.450±.016	.116±.003
	EB-Diag	.881±.003	.533±.010	.202±.007	.712±.006	1.327±.018	.252±.002	.652±.003	1.583±.017	.162±.003
	Correlated	.798±.011	.720±.015	.177±.013	.543±.013	2.013±.038	.246±.014	.416±.007	3.073±.035	.242±.010
	Ours	.916±.001	.247±.002	.007±.001	.840±.002	.606±.001	.028±.002	.791±.001	.817±.007	.030±.005
ViT	$\lambda_{\min}\mathbf{I}$	.615±.014	1.079±.044	.045±.005	.435±.009	2.226±.040	.095±.001	.416±.002	2.697±.015	.077±.001
	$\lambda_{\max}\mathbf{I}$	.807±.002	.598±.015	.053±.010	.558±.296	1.896±1.361	.117±.076	.632±.174	1.592±1.004	.029±.048
	EB-Global	.605±.019	1.104±.053	.042±.004	.427±.005	2.271±.009	.099±.004	.417±.004	2.699±.023	.078±.001
	EB-Diag	.660±.021	.955±.059	.049±.002	.374±.008	2.517±.029	.109±.004	.372±.003	2.968±.014	.100±.002
	Ours	.814±.001	.552±.002	.034±.002	.883±.002	.408±.004	.049±.000	.716±.001	1.116±.003	.009±.001

4 Experiments

4.1 Experimental Setup

Prior Configurations. We evaluate BNNs under MFVI with several prior configurations. All methods share the same deterministic solution $\mathbf{w}^*$ as the prior mean, allowing us to compare different specifications of prior precision around a common pretrained solution. We consider two empirical Bayes (EB) variants as baselines that optimize prior precision by marginal likelihood maximization (Immer et al., 2021; Daxberger et al., 2021). **EB-Global** optimizes a single shared precision, $\mathbf{\Lambda} = \lambda\mathbf{I}$, where λ is obtained by maximizing the sum of parameter-wise log marginal likelihoods under the same quadratic approximation as in Eq. 6. By contrast, **EB-Diagonal** independently optimizes each parameter-wise precision according to Eq. 6, yielding $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_d)$. Both are constrained to the precision range $[\lambda_{\min}, \lambda_{\max}]$ used throughout our experiments as described Section 3.4, with divergent solutions assigned to $\lambda_{\max}$. We additionally consider two uniform priors centered at $\mathbf{w}^*$, denoted $\lambda_{\min}\mathbf{I}$ and $\lambda_{\max}\mathbf{I}$, which assign the corresponding endpoint precision to every parameter. For convolutional networks, we additionally use the **correlated Gaussian** prior of Fortuin et al. (2022) with the prior mean replace by the pretrained solution, $p(\mathbf{w}) = \mathcal{N}(\mathbf{w} \mid \mathbf{w}^*, \Sigma)$, where Σ models spatial correlations within convolutional filters using a Matérn-1/2 kernel, with the prior scale adjusted according to the layer fan-in following the official implementation (Fortuin et al., 2021).

Datasets, architectures, and posterior initialization. We evaluate the prior configurations on CIFAR-10 (Alex, 2009), ImageNet-100 (Tian et al., 2020), and ImageNet-1K (Russakovsky et al., 2015) using both CNN and ViT architectures. For each dataset–architecture pair, we first obtain a deterministic solution $\mathbf{w}^*$ and use the same pretrained weights across all prior configurations. We adopt the default random initialization of the variational posterior used in Bayesian-torch (Krishnan et al., 2022) for all methods in the main experiments. Unless otherwise specified, all BNNs are trained using the same optimization protocol. Additional implementation details is reported in Appendix A.2.

4.2 In-Distribution Predictive Performance

Table 1 summarizes the predictive performance across three dataset scales and two architecture families. Overall, the proposed prior achieves robust and consistent performance across the evaluated settings. Most notably, the proposed prior yields the lowest expected calibration error (ECE) for every dataset and architecture combination, demonstrating a clear advantage in predictive calibration, while maintaining competitive accuracy and NLL across the evaluated settings.

A comparison between $\lambda_{\min}\mathbf{I}$ and $\lambda_{\max}\mathbf{I}$ provides an important observation regarding the use of the deterministic solution $\mathbf{w}^*$ as the prior mean. Across most settings, $\lambda_{\max}\mathbf{I}$ substantially outperforms $\lambda_{\min}\mathbf{I}$ in accuracy and NLL. Since $\lambda_{\min} = 1$ corresponds to a unit-variance Gaussian centered at

Table 2: OOD detection performance. Predictive entropy is used as the uncertainty score. Average denotes the macro average over the OOD datasets.

	CIFAR-10	TinyImageNet		Places		Textures		Average	
	Method	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓
CNN	$\lambda_{\min}\mathbf{I}$	.840±.005	.664±.010	.846±.013	.653±.031	.851±.010	.674±.026	.846±.008	.664±.019
	$\lambda_{\max}\mathbf{I}$	.852±.001	.685±.004	.870±.002	.651±.006	.858±.004	.696±.012	.860±.002	.678±.007
	EB-Global	.799±.014	.733±.020	.811±.019	.716±.024	.782±.024	.793±.029	.797±.017	.747±.022
	EB-Diag	.831±.004	.700±.010	.832±.015	.705±.028	.829±.008	.758±.019	.830±.004	.721±.014
	Correlated	.769±.016	.788±.011	.772±.013	.783±.012	.719±.027	.883±.012	.753±.015	.818±.009
	Ours	.861±.002	.632±.002	.878±.001	.592±.006	.867±.003	.662±.011	.869±.001	.628±.003
ViT	$\lambda_{\min}\mathbf{I}$	.675±.006	.869±.011	.612±.012	.898±.011	.587±.008	.950±.005	.625±.008	.906±.009
	$\lambda_{\max}\mathbf{I}$	.759±.002	.784±.002	.745±.012	.797±.013	.753±.012	.804±.019	.752±.009	.795±.011
	EB-Global	.667±.014	.875±.015	.608±.021	.901±.015	.580±.024	.953±.011	.619±.019	.910±.013
	EB-Diag	.697±.012	.836±.017	.659±.022	.863±.016	.619±.025	.932±.009	.658±.019	.877±.014
	Ours	.767±.001	.771±.001	.756±.003	.780±.003	.747±.003	.812±.004	.757±.002	.787±.002
	ImageNet-1K	NINCO		iNaturalist		Textures		Average	
	Method	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓
CNN	$\lambda_{\min}\mathbf{I}$	.774±.003	.846±.007	.878±.004	.603±.016	.820±.004	.702±.008	.824±.002	.717±.006
	$\lambda_{\max}\mathbf{I}$	.837±.000	.702±.005	.921±.001	.404±.005	.872±.001	.512±.005	.877±.001	.539±.003
	EB-Global	.775±.004	.845±.008	.876±.004	.601±.014	.822±.003	.690±.011	.824±.003	.712±.009
	EB-Diag	.765±.004	.847±.012	.865±.005	.633±.033	.797±.007	.736±.022	.809±.004	.739±.019
	Correlated	.661±.010	.868±.014	.731±.019	.834±.030	.606±.026	.831±.019	.666±.015	.844±.019
	Ours	.850±.002	.695±.007	.934±.003	.377±.020	.874±.003	.534±.014	.886±.002	.535±.011
ViT	$\lambda_{\min}\mathbf{I}$	.609±.003	.940±.003	.682±.007	.880±.007	.666±.004	.866±.006	.652±.004	.896±.004
	$\lambda_{\max}\mathbf{I}$	.746±.089	.842±.033	.838±.084	.670±.087	.801±.072	.679±.049	.795±.082	.730±.055
	EB-Global	.607±.002	.940±.002	.680±.005	.883±.007	.667±.005	.864±.009	.651±.003	.895±.005
	EB-Diag	.578±.001	.949±.002	.634±.006	.909±.005	.634±.004	.889±.007	.616±.002	.916±.003
	Ours	.792±.001	.818±.002	.885±.004	.597±.014	.832±.002	.641±.009	.836±.001	.685±.006

$\mathbf{w}^*$, these results suggest that, when $\mathbf{w}^*$ is obtained from a well-trained deterministic network, the prior should place substantially greater confidence around this solution rather than allowing broad variation around it.

However, uniformly increasing the precision is not always beneficial. While $\lambda_{\max}\mathbf{I}$ generally provides competitive accuracy and NLL for CNNs, it consistently yields higher ECE than the proposed prior. This behavior is consistent with the analysis in Section 2.2: assigning a large precision to every parameter suppresses both the displacement of the variational mean from $\mathbf{w}^*$ and the posterior variance, thereby reducing the posterior diversity available to BMA. Thus, $\lambda_{\max}\mathbf{I}$ prior can effectively preserve the predictive quality inherited from the pretrained solution, but leaves limited posterior flexibility for further improving calibration. The limitation becomes more apparent for ViTs. Although $\lambda_{\max}\mathbf{I}$ performs well on CIFAR-10, it exhibits substantial variability on ImageNet-100 and ImageNet-1K. These results suggest that uniformly imposing the maximum precision can overly restrict parameter directions that still require adaptation to the likelihood. In contrast, the proposed prior retains greater flexibility in high-sensitivity directions while strongly anchoring low-sensitivity parameters, leading to more consistent predictive performance and calibration across architectures and dataset scales.

The empirical Bayes variants generally underperform the proposed prior across datasets and architectures. Although EB-Diagonal improves accuracy and NLL over EB-Global in some settings, the improvement is not consistent and is often accompanied by worse calibration; we examine this behavior further in Section 5.

4.3 Out of Distribution Detection Performance

We additionally evaluate out-of-distribution (OOD) detection performance following the near- and far-OOD categorization of OpenOOD v1.5 (Zhang et al., 2023). For models trained on CIFAR-10, we use TinyImageNet (Wu et al., 2017) as a near-OOD dataset, and Places (Huang & Li, 2021) and Textures (Cimpoi et al., 2014) as far-OOD datasets. For models trained on ImageNet-1K, we use NINCO (Bitterwolf et al., 2023) as a near-OOD dataset, and iNaturalist (Van Horn et al., 2018) and Textures (Cimpoi et al., 2014) as far-OOD datasets. We use predictive entropy as the uncertainty

Table 3: Posterior predictive decomposition on CIFAR-10. $\Delta_{\mathrm{BMA}} = \mathcal{L}_{\mathrm{test}}(\mu) - \mathcal{L}_{\mathrm{test}}^{\mathrm{BMA}}$, where a positive value indicates that BMA improves upon the variational mean predictor.

	CNN				ViT			
Method	$\mathcal{L}_{\mathrm{test}}(\mu) \downarrow$	$\mathcal{L}_{\mathrm{test}}(\mu, \sigma) \downarrow$	$\mathcal{L}_{\mathrm{test}}^{\mathrm{BMA}} \downarrow$	Δ_{BMA}	$\mathcal{L}_{\mathrm{test}}(\mu) \downarrow$	$\mathcal{L}_{\mathrm{test}}(\mu, \sigma) \downarrow$	$\mathcal{L}_{\mathrm{test}}^{\mathrm{BMA}} \downarrow$	Δ_{BMA}
$\lambda_{\min}\mathbf{I}$	.3586$\pm$.0131	.7091$\pm$.0207	.3900$\pm$.0132	$-$.0314	1.1340$\pm$.0586	1.1603$\pm$.0423	1.0793$\pm$.0444	+.0547
$\lambda_{\max}\mathbf{I}$	.3550$\pm$.0043	.3604$\pm$.0043	.3382$\pm$.0039	+.0169	.6196$\pm$.0202	.6199$\pm$.0173	.5979$\pm$.0148	+.0217
EB-Global	.4933$\pm$.0448	1.0612$\pm$.0333	.6308$\pm$.0183	$-$.1375	1.1726$\pm$.0491	1.1841$\pm$.0505	1.1043$\pm$.0529	+.0682
EB-Diag	.3001$\pm$.0128	1.1295$\pm$.0433	.5333$\pm$.0096	$-$.2333	.9996$\pm$.0607	1.0415$\pm$.0538	.9546$\pm$.0591	+.0450
Ours	.3352$\pm$.0027	.4761$\pm$.0067	.2473$\pm$.0023	+.0879	.6091$\pm$.0018	.6323$\pm$.0018	.5519$\pm$.0022	+.0572

score and report the area under the receiver operating characteristic curve (AUROC) and the false positive rate at 95% (FPR95).

As shown in Table 2, the proposed prior provides the most consistent OOD detection performance across datasets and architectures. It achieves both the highest macro-average AUROC and the lowest macro-average FPR95 in all four dataset–architecture settings. Notably, although the $\lambda_{\max}\mathbf{I}$ prior is highly competitive for in-distribution prediction, this advantage does not consistently translate to OOD detection, where the proposed prior achieves higher average AUROC and lower average FPR95 across all evaluated settings.

The proposed prior also performs well across both near- and far-OOD datasets. It achieves the best AUROC and FPR95 on TinyImageNet and NINCO for both CNN and ViT architectures, and remains best on most far-OOD datasets. The main exceptions occur on Textures, where $\lambda_{\max}\mathbf{I}$ performs better for the CIFAR-10 ViT and achieves lower FPR95 for the ImageNet-1K CNN. Nevertheless, the proposed prior retains the best macro-average performance for both CIFAR-10 and ImageNet-1K, suggesting that its advantage is not tied to a particular OOD dataset or architectural family. Together with the in-distribution results in Table 1, these results indicate that the improved predictive calibration of the proposed prior is accompanied by strong and consistent OOD discrimination.

5 Analysis

Section 2.2 showed that prior precision controls both the displacement of the posterior mean and the uncertainty retained around the pretrained solution. We analyze the proposed prior from two perspectives. First, through posterior predictive decomposition, we examine the predictive quality of the variational mean, individual posterior samples, and their aggregation through Bayesian model averaging. Second, we study the effect of precision allocation across parameters by comparing alternative shrinkage priors centered at the same pretrained solution, including spike-and-slab and horseshoe priors, and by contrasting the proposed and opposite sensitivity–precision orderings through their loss landscapes.

5.1 Posterior Predictive Decomposition

Let $\mathcal{L}_{\mathrm{test}}(\mathbf{w})$ denote the NLL on the test set for a network with parameters $\mathbf{w}$. For the variational posterior $q_\phi(\mathbf{w})$, we first evaluate the posterior mean network using $\mathcal{L}_{\mathrm{test}}(\boldsymbol{\mu})$, which measures the predictive quality of the learned variational mean independently of posterior sampling. We then define $\mathcal{L}_{\mathrm{test}}(\boldsymbol{\mu}, \boldsymbol{\sigma}) = \mathbb{E}_{\mathbf{w}\sim q_\phi}[\mathcal{L}_{\mathrm{test}}(\mathbf{w})]$, which measures the expected NLL of individual networks sampled from the posterior. Finally, $\mathcal{L}_{\mathrm{test}}^{\mathrm{BMA}} = \mathbb{E}_{(x,y)\sim\mathcal{D}_{\mathrm{test}}}\left[-\log \mathbb{E}_{\mathbf{w}\sim q_\phi}[p(y \mid x, \mathbf{w})]\right]$ denotes the NLL of the approximate posterior predictive distribution in Eq. 1, obtained by replacing $p(\mathbf{w} \mid \mathcal{D})$ with $q_\phi(\mathbf{w})$, and corresponds to the predictive NLL reported in the main experiments.

Table 3 reveals qualitatively different predictive behaviors across the prior configurations. For the $\lambda_{\max}\mathbf{I}$ prior, $\mathcal{L}_{\mathrm{test}}(\boldsymbol{\mu})$ and $\mathcal{L}_{\mathrm{test}}(\boldsymbol{\mu}, \boldsymbol{\sigma})$ are nearly identical for both architectures, indicating that posterior sampling has little effect on the average test NLL. Accordingly, BMA provides only a modest improvement over the variational mean predictor. This is consistent with Section 2.2, where a large uniform precision suppresses posterior variation around the pretrained solution.

At the opposite extreme, the $\lambda_{\min}\mathbf{I}$ prior produces a substantially larger increase in NLL under posterior sampling, and this variation is not consistently beneficial; for the CNN, BMA performs

worse than the variational mean predictor. In contrast, the proposed prior exhibits a larger gap between $\mathcal{L}_{\text{test}}(\mu)$ and $\mathcal{L}_{\text{test}}(\mu, \sigma)$, while BMA substantially reduces the NLL relative to individual posterior samples. As a result, the proposed prior yields a larger positive Δ_{BMA} than the $\lambda_{\max}\mathbf{I}$ prior for both CNN and ViT. Thus, the posterior variation under the proposed prior is more effectively utilized through BMA than that induced by the uniform high-precision prior.

The empirical Bayes variants exhibit a different behavior. For the CNN, posterior sampling substantially increases the NLL for both EB variants. In particular, EB-Diagonal attains the best variational mean predictor among all compared methods, but the expected NLL of posterior samples increases sharply, and BMA does not fully recover the performance of the variational mean predictor, resulting in a negative Δ_{BMA}. In contrast, for the ViT, the EB variants already exhibit poor NLL at the variational mean, while posterior sampling causes only a modest additional increase and BMA improves upon the mean predictor. These results indicate that the poor predictive performance of EB under MFVI cannot be attributed to posterior sampling alone and may also reflect sensitivity in optimizing the variational posterior.

To examine this sensitivity, we repeat the posterior predictive decomposition under two warm-start posterior initializations (Krishnan et al., 2020) in Appendix A.3. Warm-starting generally improves predictive performance across methods, including the EB variants, although the favorable initialization varies across datasets and architectures. Nevertheless, the proposed prior achieves the lowest $\mathcal{L}_{\text{test}}^{\text{BMA}}$ in six of the eight warm-start comparisons, indicating that its favorable BMA performance is not confined to the default random initialization. This also suggests that the weaker EB performance cannot be explained by initialization alone, but is related to the precision selected by local evidence optimization.

5.2 Effect of Precision Allocation

The posterior decomposition above shows that the proposed prior maintains competitive predictive performance at the variational mean while benefiting substantially from BMA. We next examine how prior precision allocation affects the observed predictive gains, including whether selective shrinkage around the pretrained solution alone can explain these gains.

Table 4: Effect of heterogeneous prior precision allocation on CIFAR-10. *Shuffled* randomly permutes the proposed precisions within each layer, while *Opposite* reverses the sensitivity–precision ordering. AUROC is evaluated on TinyImageNet.

	Allocation	Acc. ↑	NLL ↓	ECE ↓	AUROC ↑
CNN	Spike&slab	.894±.003	.331±.005	.035±.006	.850±.004
	Horseshoe	.906±.005	.301±.003	.033±.004	.847±.008
	Shuffled	.915±.001	.260±.003	.022±.000	.859±.001
	Opposite	.915±.001	.308±.006	.043±.001	.855±.002
	Ours	.916±.001	.247±.002	.007±.001	.861±.002
ViT	Spike&slab	.744±.061	.803±.155	.079±.008	.722±.042
	Horseshoe	.802±.006	.598±.012	.046±.006	.751±.006
	Shuffled	.813±.004	.566±.006	.039±.002	.766±.001
	Opposite	.797±.009	.607±.008	.043±.020	.760±.003
	Ours	.814±.001	.552±.002	.034±.002	.767±.001

We compare the proposed allocation with spike-and-slab and horseshoe priors (Ghosh et al., 2018), centered at the same pretrained solution $\mathbf{w}^*$, such that their shrinkage acts on deviations from $\mathbf{w}^*$ rather than toward zero. To further isolate the effect of the sensitivity–precision correspondence, *Shuffled* randomly permutes the proposed precisions within each layer, preserving the layer-wise precision distribution while removing the correspondence with sensitivity. *Opposite* reverses the proposed ordering, $\lambda_i = u_i\lambda_{\max} + (1 - u_i)\lambda_{\min}$, assigning higher precision to more sensitive parameters.

As shown in Table 4, the proposed prior achieves the best performance across all metrics for both architectures. Although the spike-and-slab and horseshoe priors introduce additional shrinkage variables, they do not match the proposed prior in these settings, suggesting that greater prior flexibility alone does not explain the observed gains. By contrast, *Opposite* consistently performs worse than *Shuffled* across all metrics for both architectures. Since *Shuffled* removes the sensitivity–precision correspondence, whereas *Opposite* reverses it, this result suggests that the opposite ordering is less effective than removing the correspondence altogether. The proposed allocation also improves upon *Shuffled*, supporting the proposed precision ordering in which more sensitive parameters receive lower prior precision and retain greater posterior flexibility.

To further examine the effect of the sensitivity–precision ordering, we compare the predictive loss landscape under *Ours* and *Opposite*. We partition the parameters into low- and high-sensitivity groups, $\mathcal{G}_L = \{i : u_i \leq 0.5\}$ and $\mathcal{G}_H = \{i : u_i > 0.5\}$, and perturb each group separately. Figure 1

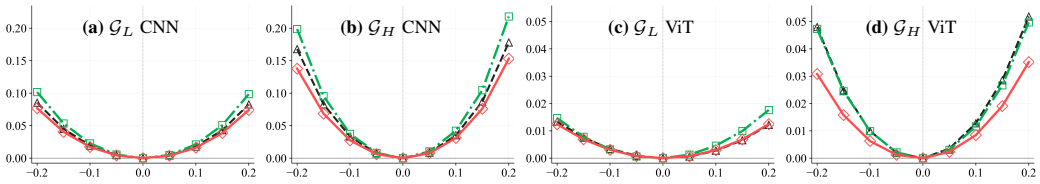


Figure 1: Predictive loss landscapes. Panels (a,b) show CNN results for $\mathcal{G}_L$ and $\mathcal{G}_H$, while panels (c,d) show the corresponding ViT results. The x-axis denotes perturbation magnitude and the y-axis the increase in predictive NLL from the unperturbed model; larger values indicate sharper landscapes. Curves denote DNN, *Opposite*, and *Ours*.

visualizes the predictive loss landscapes obtained by perturbing each group along filter-wise normalized directions (Li et al., 2018), with the pretrained DNN serving as a reference. The difference between *Ours* and *Opposite* is most pronounced in $\mathcal{G}_H$, where *Ours* yields a flatter landscape and a smaller increase in predictive NLL for both CNN and ViT. *Ours* also exhibits flatter landscapes than the pretrained DNN across both sensitivity groups, while the landscapes in $\mathcal{G}_L$ are relatively flat overall. These results show that the proposed precision ordering is associated with flatter predictive landscapes, particularly in sensitive directions, and are consistent with the improved ECE and OOD detection reported in Tables 1 and 2. Further details are provided in Appendix A.2.

5.3 Conventional Priors for BNNs Trained from Scratch

We additionally evaluate BNNs trained from scratch without pretrained weights. Conventional priors remain reasonably competitive on CIFAR-10, but their predictive performance degrades much more severely on the larger-scale ImageNet-100 setting, particularly for ViTs. Low ECE alone can be misleading in these cases, as some priors remain well calibrated despite substantial loss of predictive accuracy. The larger performance gap on ImageNet-100 suggests that leveraging a pretrained solution becomes increasingly beneficial at scale by providing an informative prior for MFVI.

Table 5: Comparison with conventional priors under MFVI. The conventional priors are trained from scratch, whereas Ours uses the pretrained deterministic solution. For CIFAR-10, OOD detection is evaluated on TinyImageNet, while for ImageNet-100, it is evaluated on NINCO.

		CIFAR-10					ImageNet-100				
	Prior	Acc. ↑	NLL ↓	ECE ↓	AUROC ↑	FPR95 ↓	Acc. ↑	NLL ↓	ECE ↓	AUROC ↑	FPR95 ↓
CNN	Gaussian	.883±.003	.388±.008	.081±.004	.846±.004	.657±.007	.728±.008	1.115±.030	.159±.005	.795±.009	.774±.021
	Laplace	.876±.004	.458±.011	.131±.004	.844±.005	.663±.012	.700±.012	1.314±.030	.217±.008	.767±.005	.798±.003
	Spike&Slab	.839±.015	.504±.045	.054±.006	.814±.017	.730±.045	.566±.007	1.716±.028	.111±.013	.722±.014	.808±.020
	Horseshoe	.880±.007	.354±.021	.014±.004	.831±.009	.673±.008	.469±.071	2.185±.319	.155±.072	.665±.041	.865±.045
	Ours	.916±.001	.247±.002	.007±.001	.861±.002	.632±.002	.840±.002	.606±.001	.028±.002	.868±.002	.633±.005
ViT	Gaussian	.597±.013	1.130±.042	.045±.005	.669±.009	.878±.010	.156±.001	3.690±.013	.081±.004	.429±.012	.954±.006
	Laplace	.624±.012	1.053±.030	.060±.005	.674±.010	.861±.012	.155±.006	3.696±.019	.082±.007	.438±.010	.953±.003
	Spike&Slab	.628±.021	1.033±.062	.015±.002	.675±.008	.866±.009	.144±.004	3.700±.022	.036±.003	.435±.019	.956±.009
	Horseshoe	.767±.015	.665±.040	.012±.002	.747±.012	.798±.013	.135±.126	3.847±.759	.034±.007	.477±.037	.950±.009
	Ours	.814±.001	.552±.002	.034±.002	.767±.001	.771±.001	.883±.002	.408±.004	.049±.000	.904±.002	.490±.007

6 Conclusion

We studied how prior precision should be allocated across individual weights when a pretrained neural network is available. We proposed an empirical prior that leverages pruning sensitivity to retain greater posterior flexibility in sensitive directions while strongly anchoring less sensitive parameters. Across datasets and CNN and ViT architectures, the proposed allocation achieves strong calibration while maintaining competitive predictive likelihood and OOD detection. Our analyses further show that its benefit arises from where posterior flexibility is retained, rather than from heterogeneous precision alone. Overall, these results suggest that pruning provides useful parameter-wise information for designing informative priors in BNNs, extending its role beyond model compression.

AI USE STATEMENT

We used generative AI tools to assist with mathematical derivations, drafting and editing parts of the manuscript, and debugging implementation code. All AI-assisted mathematical arguments, code-related suggestions, and manuscript content were reviewed and verified by the authors. All experiments were conducted by the authors, and the authors take full responsibility for the final content, claims, code, and results presented in this work.

REFERENCES

Krizhevsky Alex. Learning multiple layers of features from tiny images. *https://www. cs. toronto. edu/kriz/learning-features-2009-TR. pdf*, 2009.

Andrei Atanov, Arsenii Ashukha, Kirill Struminsky, Dmitry Vetrov, and Max Welling. The deep weight prior. In *International Conference on Learning Representations*, 2019.

Brian Bartoldson, Ari Morcos, Adrian Barbu, and Gordon Erlebacher. The generalization-stability tradeoff in neural network pruning. *Advances in neural information processing systems*, 33: 20852–20864, 2020.

Julian Bitterwolf, Maximilian Müller, and Matthias Hein. In or out? Fixing ImageNet out-of-distribution detection evaluation. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2471–2506. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/bitterwolf23a.html.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pp. 1613–1622. PMLR, 2015a.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight Uncertainty in Neural Networks, May 2015b. URL http://arxiv.org/abs/1505.05424. arXiv:1505.05424 [stat].

Carlos M. Carvalho, Nicholas G. Polson, and James G. Scott. Handling sparsity via the horseshoe. In David van Dyk and Max Welling (eds.), *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pp. 73–80, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL https://proceedings.mlr.press/v5/carvalho09a.html.

Tapabrata Chakraborti, Christopher RS Banerji, Ariane Marandon, Vicky Hellon, Robin Mitra, Brieuc Lehmann, Leandra Bräuninger, Sarah McGough, Cagatay Turkay, Alejandro F Frangi, et al. Personalized uncertainty quantification in artificial intelligence. *Nature Machine Intelligence*, 7(4):522–530, 2025.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux-effortless bayesian deep learning. *Advances in neural information processing systems*, 34:20089–20103, 2021.

Rayen Dhahri, Alexander Immer, Betrand Charpentier, Stephan Günnemann, and Vincent Fortuin. Shaving weights with occam's razor: Bayesian sparsification for neural networks using the marginal likelihood. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 24959–24989. Curran Associates, Inc., 2024. doi: 10.52202/079017-0787. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/2c572cad9ae98c5cb6f3fca040b2bc54-Paper-Conference.pdf.

Zhi-Quan Feng, Ying-Jia Lin, and Hung-Yu Kao. Learning in the fisher subspace: A guided initialization for lora fine-tuning. *arXiv preprint arXiv:2605.01046*, 2026.

Vincent Fortuin, Adrià Garriga-Alonso, Mark van der Wilk, and Laurence Aitchison. BNNpriors: A library for Bayesian neural network inference with different prior distributions. *Software Impacts*, 9:100079, 2021.

Vincent Fortuin, Adrià Garriga-Alonso, Florian Wenzel, Gunnar Rätsch, Richard E. Turner, Mark van der Wilk, and Laurence Aitchison. Bayesian neural network priors revisited. In *International Conference on Learning Representations*, 2022.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

Soumya Ghosh, Jiayu Yao, and Finale Doshi-Velez. Structured variational learning of bayesian neural networks with horseshoe priors. In *International conference on machine learning*, pp. 1744–1753. PMLR, 2018.

Soumya Ghosh, Jiayu Yao, and Finale Doshi-Velez. Model selection in bayesian neural networks via horseshoe priors. *Journal of Machine Learning Research*, 20(182):1–46, 2019.

Alex Graves. Practical variational inference for neural networks. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper_files/paper/2011/file/7eb3c8be3d411e8ebfab08eba5f49632-Paper.pdf.

Babak Hassibi and David Stork. Second order derivatives for network pruning: Optimal brain surgeon. In S. Hanson, J. Cowan, and C. Giles (eds.), *Advances in Neural Information Processing Systems*, volume 5. Morgan-Kaufmann, 1992. URL https://proceedings.neurips.cc/paper_files/paper/1992/file/303ed4c69846ab36c2904d3ba8573050-Paper.pdf.

Wenchong He, Zhe Jiang, Tingsong Xiao, Zelin Xu, and Yukun Li. A survey on uncertainty quantification methods for deep learning. *ACM Computing Surveys*, 58(7):1–35, 2026.

Rui Huang and Yixuan Li. Mos: Towards scaling out-of-distribution detection for large semantic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

Alexander Immer, Matthias Bauer, Vincent Fortuin, Gunnar Rätsch, and Khan Mohammad Emtiyaz. Scalable marginal likelihood estimation for model selection in deep learning. In *International Conference on Machine Learning*, pp. 4563–4573. PMLR, 2021.

Sanket Jantre, Shrijita Bhattacharya, and Tapabrata Maiti. Spike-and-slab shrinkage priors for structurally sparse bayesian neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 36(6):11176–11188, 2024.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

Ranganath Krishnan, Mahesh Subedar, and Omesh Tickoo. Specifying weight priors in bayesian deep neural networks with empirical bayes. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 4477–4484, 2020.

Ranganath Krishnan, Pi Esposito, and Mahesh Subedar. Bayesian-torch: Bayesian neural network layers for uncertainty estimation, January 2022. URL https://doi.org/10.5281/zenodo.5908307.

Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. In D. Touretzky (ed.), *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1989. URL https://proceedings.neurips.cc/paper_files/paper/1989/file/6c9882bbac1c7093bd25041881277658-Paper.pdf.

Namhoon Lee, Thalaiyasingam Ajanthan, and Philip HS Torr. Snip: Single-shot network pruning based on connection sensitivity. *arXiv preprint arXiv:1810.02340*, 2018.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018.

TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016.

T. J. Mitchell and J. J. Beauchamp. Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):1023–1032, 1988. doi: 10.1080/01621459.1988.10478694. URL https://www.tandfonline.com/doi/abs/10.1080/01621459.1988.10478694.

Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11256–11264. IEEE, 2019.

Nicholas G. Polson and James G. Scott. Shrink globally, act locally: Sparse bayesian regularization and prediction. In José M. Bernardo, M. J. Bayarri, James O. Berger, A. P. Dawid, David Heckerman, Adrian F. M. Smith, and Mike West (eds.), *Bayesian Statistics 9*. Oxford University Press, 10 2011. ISBN 9780199694587. doi: 10.1093/acprof:oso/9780199694587.003.0017. URL https://doi.org/10.1093/acprof:oso/9780199694587.003.0017.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Yan Sun, Qifan Song, and Faming Liang. Learning sparse deep neural networks with a spike-and-slab prior. *Statistics & probability letters*, 180:109246, 2022.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive Multiview Coding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. doi: 10.1007/978-3-030-58621-8_45. URL https://mlanthology.org/eccv/2020/tian2020eccv-contrastive/.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The INaturalist Species Classification and Detection Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018. doi: 10.1109/CVPR.2018.00914. URL https://mlanthology.org/cvpr/2018/horn2018cvpr-inaturalist/.

Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. *arXiv preprint arXiv:2002.07376*, 2020.

Zhaoyuan Wang, Siqi Bu, Jiaxin Wen, and Can Huang. A comprehensive review on uncertainty modeling methods in modern power systems. *International Journal of Electrical Power & Energy Systems*, 166:110534, 2025.

Florian Wenzel, Kevin Roth, Bastiaan Veeling, Jakub Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, and Sebastian Nowozin. How good is the bayes posterior in deep neural networks really? In *International Conference on Machine Learning*, pp. 10248–10259. PMLR, 2020a. URL https://proceedings.mlr.press/v119/wenzel20a.html.

Florian Wenzel, Kevin Roth, Bastiaan S Veeling, Linh Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, Sebastian Nowozin, et al. How good is the bayes posterior in deep neural networks really? *arXiv preprint arXiv:2002.02405*, 2020b.

Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Jiayu Wu, Qixiang Zhang, and Guoxi Xu. Tiny imagenet challenge. 2017. *URL https://api. semanticscholar. org/CorpusID*, 212697711, 2017.

Jingyang Zhang, Jingkang Yang, Pengyun Wang, Haoqi Wang, Yueqian Lin, Haoran Zhang, Yiyou Sun, Xuefeng Du, Kaiyang Zhou, Wayne Zhang, Yixuan Li, Ziwei Liu, Yiran Chen, and Li Hai. Openood v1.5: Enhanced benchmark for out-of-distribution detection. *arXiv preprint arXiv:2306.09301*, 2023.

A Appendix

A.1 Derivation of the Local Empirical Bayes Precision

We provide the derivation of the local empirical Bayes estimate in Eq. 6. We consider each parameter w_i independently while fixing all remaining parameters at the deterministic solution $\mathbf{w}^*$. For the i-th parameter, we place a Gaussian prior centered at w_i^*,

$$p(w_i \mid w_i^*, \lambda_i) = \mathcal{N}(w_i \mid w_i^*, \lambda_i^{-1}),$$

where $\lambda_i > 0$ denotes the prior precision. The corresponding local marginal likelihood is

$$p(\mathcal{D} \mid \lambda_i, w_i^*) = \int p(\mathcal{D} \mid w_i) p(w_i \mid w_i^*, \lambda_i) dw_i.$$

The empirical Bayes estimate is obtained by maximizing the local marginal likelihood with respect to the prior precision:

$$\lambda_i^* = \arg\max_{\lambda_i > 0} p(\mathcal{D} \mid \lambda_i, w_i^*).$$

We define the local displacement from the deterministic solution as $w_i = w_i^* + \delta_i$. Let $\mathcal{L}_{\mathrm{NLL}}(\mathbf{w}) = -\log p(\mathcal{D} \mid \mathbf{w})$. Around w_i^*, we approximate the negative log likelihood along the i-th coordinate using a second-order Taylor expansion:

$$\mathcal{L}_{\mathrm{NLL}}(w_i^* + \delta_i) \approx \mathcal{L}_{\mathrm{NLL}}(w_i^*) + g_i \delta_i + \frac{1}{2} H_{ii} \delta_i^2,$$

where g_i and H_{ii} denote the gradient and the i-th diagonal element of the Hessian of the NLL, respectively, evaluated at $\mathbf{w}^*$. Since pretrained DNNs are usually obtained from regularized training, often corresponding to maximum a posteriori (MAP) estimation, the NLL gradient at w^* need not vanish. We assume $H_{ii} > 0$, corresponding to positive local curvature along the i-th coordinate.

Accordingly, the local likelihood can be approximated as

$$p(\mathcal{D} \mid w_i) = p(\mathcal{D} \mid w_i^* + \delta_i) \approx \exp\left(-\mathcal{L}_{\mathrm{NLL}}(w_i^* + \delta_i)\right) = \exp\left(-\mathcal{L}_{\mathrm{NLL}}(w_i^*) - g_i \delta_i - \frac{1}{2} H_{ii} \delta_i^2\right).$$

The second term in the integral is the prior distribution over w_i. We assume an independent Gaussian prior centered at the deterministic solution w_i^*, with precision λ_i:

$$p(w_i \mid w_i^*, \lambda_i) = \mathcal{N}(w_i \mid w_i^*, \lambda_i^{-1}) = \sqrt{\frac{\lambda_i}{2\pi}} \exp\left[-\frac{\lambda_i}{2}(w_i - w_i^*)^2\right].$$

Using the local displacement defined above, the prior can therefore be rewritten as

$$p(w_i^* + \delta_i \mid w_i^*, \lambda_i) = \sqrt{\frac{\lambda_i}{2\pi}} \exp\left(-\frac{\lambda_i}{2}\delta_i^2\right).$$

We can now substitute the local approximation of the likelihood and the Gaussian prior into the marginal likelihood:

$$p(\mathcal{D} \mid \lambda_i, w_i^*) \approx \int \exp\left(-\mathcal{L}_{\mathrm{NLL}}(w_i^*) - g_i \delta_i - \frac{1}{2} H_{ii} \delta_i^2\right) \sqrt{\frac{\lambda_i}{2\pi}} \exp\left[-\frac{\lambda_i}{2}\delta_i^2\right] d\delta_i.$$

To evaluate the integral, we complete the square in δ_i. Specifically,

$$-g_i \delta_i - \frac{1}{2}(H_{ii} + \lambda_i)\delta_i^2 = -\frac{H_{ii} + \lambda_i}{2}\left(\delta_i + \frac{g_i}{H_{ii} + \lambda_i}\right)^2 + \frac{g_i^2}{2(H_{ii} + \lambda_i)}.$$

Thus,

$$p(\mathcal{D} \mid \lambda_i, w_i^*) \approx \exp[-\mathcal{L}_{\mathrm{NLL}}(w_i^*)] \sqrt{\frac{\lambda_i}{2\pi}} \exp\left[\frac{g_i^2}{2(H_{ii} + \lambda_i)}\right] \times \int \exp\left[-\frac{H_{ii} + \lambda_i}{2}\left(\delta_i + \frac{g_i}{H_{ii} + \lambda_i}\right)^2\right] d\delta_i.$$

Since the integral is a Gaussian integral, it gives

$$\int \exp\left[-\frac{H_{ii}+\lambda_i}{2}\left(\delta_i+\frac{g_i}{H_{ii}+\lambda_i}\right)^2\right] d\delta_i = \sqrt{\frac{2\pi}{H_{ii}+\lambda_i}}.$$

Therefore,

$$p(\mathcal{D} \mid \lambda_i, w_i^*) \approx \exp[-\mathcal{L}_{\mathrm{NLL}}(w_i^*)]\sqrt{\frac{\lambda_i}{H_{ii}+\lambda_i}}\exp\left[\frac{g_i^2}{2(H_{ii}+\lambda_i)}\right].$$

Taking the logarithm and discarding terms independent of λ_i, the local empirical Bayes objective becomes

$$\mathcal{J}_i(\lambda_i) = \frac{1}{2}\log\lambda_i - \frac{1}{2}\log(H_{ii}+\lambda_i) + \frac{g_i^2}{2(H_{ii}+\lambda_i)}.$$

To obtain the optimal λ_i, we differentiate the objective with respect to λ_i, which yields

$$\frac{\partial \mathcal{J}_i}{\partial \lambda_i} = \frac{1}{2\lambda_i} - \frac{1}{2(H_{ii}+\lambda_i)} - \frac{g_i^2}{2(H_{ii}+\lambda_i)^2}.$$

Setting the derivative to zero gives the stationary point

$$\lambda_i^* = \frac{H_{ii}^2}{g_i^2 - H_{ii}}, \qquad g_i^2 > H_{ii}.$$

The derivative can equivalently be written as

$$\frac{\partial \mathcal{J}_i}{\partial \lambda_i} = \frac{H_{ii}^2 + \lambda_i(H_{ii}-g_i^2)}{2\lambda_i(H_{ii}+\lambda_i)^2}.$$

When $g_i^2 > H_{ii}$, the derivative changes sign from positive to negative at λ_i^*, making it the unique finite maximizer. Because $H_{ii} > 0$ and $\lambda_i > 0$, the denominator is strictly positive. When $g_i^2 \le H_{ii}$, the numerator also satisfies

$$H_{ii}^2 + \lambda_i(H_{ii}-g_i^2) > 0 \qquad \forall \lambda_i > 0.$$

Hence,

$$\frac{\partial \mathcal{J}_i}{\partial \lambda_i} > 0 \qquad \forall \lambda_i > 0.$$

Therefore, $\mathcal{J}_i(\lambda_i)$ increases monotonically with λ_i, and the empirical Bayes solution is attained in the limit

$$\lambda_i^* \to \infty, \qquad g_i^2 \le H_{ii}.$$

We next derive the local variational optimum in Eq. 5. Let $q(w_i) = \mathcal{N}(w_i \mid \mu_i, \sigma_i^2)$. Under the same local quadratic approximation, the expected NLL is

$$\mathbb{E}_{q(w_i)}[\mathcal{L}_{\mathrm{NLL}}(w_i)] \approx \mathcal{L}_{\mathrm{NLL}}(w_i^*) + g_i(\mu_i - w_i^*) + \frac{1}{2}H_{ii}\left[(\mu_i - w_i^*)^2 + \sigma_i^2\right].$$

The KL divergence to the Gaussian prior is

$$D_{\mathrm{KL}}[q(w_i)\|p(w_i \mid w_i^*, \lambda_i)] = \frac{1}{2}[\lambda_i(\sigma_i^2 + (\mu_i - w_i^*)^2) - 1 - \log(\lambda_i\sigma_i^2)].$$

Therefore, up to terms independent of μ_i and σ_i^2, the local negative ELBO is

$$\mathcal{J}_i(\mu_i, \sigma_i^2) = g_i(\mu_i - w_i^*) + \frac{1}{2}(H_{ii}+\lambda_i)[(\mu_i - w_i^*)^2 + \sigma_i^2] - \frac{1}{2}\log\sigma_i^2.$$

Taking derivatives with respect to the variational mean and variance gives

$$\frac{\partial \mathcal{J}_i}{\partial \mu_i} = g_i + (H_{ii}+\lambda_i)(\mu_i - w_i^*),$$
$$\frac{\partial \mathcal{J}_i}{\partial \sigma_i^2} = \frac{1}{2}(H_{ii}+\lambda_i) - \frac{1}{2\sigma_i^2}.$$

Setting both derivatives to zero yields

$$\mu_i = w_i^* - \frac{g_i}{H_{ii}+\lambda_i}, \qquad \sigma_i^2 = \frac{1}{H_{ii}+\lambda_i},$$

which gives Eq. 5.

A.2 Implementation Details

Architectures and deterministic solutions. For CIFAR-10, we use ResNet-20 and a DeiT-Tiny variant adapted to 32×32 inputs. The DeiT variant uses a patch size of 4, an embedding dimension of 192, 12 transformer blocks, 3 attention heads, and an MLP ratio of 2. The deterministic CIFAR-10 models are trained on the corresponding training set to obtain $\mathbf{w}^*$. For ImageNet-100, which consists of a subset of ImageNet-1K classes, we use ImageNet-1K pretrained ResNet-18 and DeiT-Tiny models and adapt their classification heads to the 100 target classes without additional deterministic training. For ImageNet-1K, we use an ImageNet-1K pretrained ResNet-50 from TorchVision (maintainers & contributors, 2016) and `deit_tiny_patch16_224.fb_in1k` from `timm` (Wightman, 2019). For each dataset and architecture, the same deterministic solution $\mathbf{w}^*$ is used for all prior configurations.

MFVI training. For the main experiments, we use SGD with momentum 0.9. The batch size is 128 for CIFAR-10 and 256 for ImageNet-100 and ImageNet-1K. We use cosine learning rate annealing with a maximum learning rate of 0.1 and a minimum of 0.001, without additional weight decay. Convolutional and linear layers are treated probabilistically, while normalization layers remain deterministic, following the Bayesian-Torch implementation (Krishnan et al., 2022). Additional implementation details and hyperparameter configurations are available in our anonymous code repository: `https://anonymous.4open.science/r/empirical_prior_pruning`.

Sensitivity estimation. Sensitivity scores are computed once at the deterministic solution $\mathbf{w}^*$ before MFVI training. We compute the NLL gradient g_i and approximate the diagonal GGN entry G_{ii} using the training set. The diagonal GGN is estimated using three Monte Carlo samples ($S = 3$).

Evaluation. Predictive probabilities are approximated by averaging 30 samples from the variational posterior. ECE is computed using 15 equal-width confidence bins. For OOD detection, we use predictive entropy as the uncertainty score and report AUROC and FPR95. All reported results are averaged over five random seeds.

Loss landscape. We analyze predictive loss landscapes within parameter subspaces defined by pruning sensitivity. Among the convolutional and linear weights included in the sensitivity ranking, we define $\mathcal{G}_{\mathrm{H}} = \{i : u_i > 0.5\}$ and $\mathcal{G}_{\mathrm{L}} = \{i : u_i \leq 0.5\}$, corresponding to the top and bottom 50% of the globally ranked parameters, respectively. When analyzing a group, only its parameters are perturbed, while all other parameters remain fixed. For each group $\mathcal{G}$, we sample $K = 10$ masked Gaussian directions. Let $\mathbf{z}_{\ell,j}^{(k)}$ denote the Gaussian direction for output filter j in layer ℓ, and let $\mathbf{M}_{\mathcal{G},\ell,j}$ denote the corresponding group mask. We first restrict the direction to the selected group:

$$\widetilde{\mathbf{d}}_{\mathcal{G},\ell,j}^{(k)} = \mathbf{M}_{\mathcal{G},\ell,j} \odot \mathbf{z}_{\ell,j}^{(k)}.$$

Following the filter normalization of Li et al. (2018), each nonzero filter is then rescaled according to

$$\mathbf{d}_{\mathcal{G},\ell,j}^{(k)} = \frac{\left\|\mathbf{w}_{\ell,j}^{\star}\right\|_2}{\left\|\widetilde{\mathbf{d}}_{\mathcal{G},\ell,j}^{(k)}\right\|_2} \widetilde{\mathbf{d}}_{\mathcal{G},\ell,j}^{(k)},$$

where $\mathbf{w}^\star$ denotes the pretrained DNN weights. Filters containing no parameters from the selected group remain zero. Within each architecture, the same perturbation directions are shared across the DNN, all BNN methods, and all training seeds. We evaluate nine uniformly spaced perturbation coefficients $t \in [-0.2, 0.2]$:

$$\mathbf{w}(t; \mathcal{G}, k) = \mathbf{w}_0 + t\mathbf{d}_{\mathcal{G}}^{(k)}.$$

For the DNN, $\mathbf{w}_0 = \mathbf{w}^\star$. For a BNN, the perturbation translates the variational mean while keeping the learned posterior variance fixed:

$$q_{t,k}(\mathbf{w}; \mathcal{G}) = \mathcal{N}\Big(\mathbf{w} \mid \boldsymbol{\mu} + t\mathbf{d}_{\mathcal{G}}^{(k)}, \mathrm{diag}(\boldsymbol{\sigma}^2)\Big).$$

Predictive NLL is evaluated on all 10,000 CIFAR-10 test examples. For each BNN training seed, we draw $S = 30$ posterior weight samples once and reuse them across all groups, directions, and

Table A1: Symmetric predictive-NLL rise at $t = \pm 0.2$ for the $\mathcal{G}_H$ and $\mathcal{G}_L$ sensitivity groups. For the DNN, variation is measured across 10 perturbation directions. Lower values indicate flatter landscapes.

	CNN		ViT	
Method	$\mathcal{G}_H \downarrow$	$\mathcal{G}_L \downarrow$	$\mathcal{G}_H \downarrow$	$\mathcal{G}_L \downarrow$
DNN	$.1730_{\pm .0411}$	$.0841_{\pm .0169}$	$.0499_{\pm .0063}$	$.0128_{\pm .0014}$
Shuffled	$.1554_{\pm .0078}$	$.0781_{\pm .0012}$	$.0346_{\pm .0014}$	$.0133_{\pm .0013}$
Opposite	$.2084_{\pm .0154}$	$.1002_{\pm .0042}$	$.0485_{\pm .0018}$	$.0161_{\pm .0018}$
Ours	$.1455_{\pm .0020}$	$.0756_{\pm .0010}$	$.0330_{\pm .0020}$	$.0124_{\pm .0008}$

perturbation coefficients. The Monte Carlo Bayesian model averaging NLL is

$$\widehat{\mathcal{L}}_{\text{test}}^{\text{BMA}}(t; \mathcal{G}, k) = -\frac{1}{N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} \log \left[\frac{1}{S} \sum_{s=1}^{S} p\Big(y_n \mid x_n, \mathbf{w}^{(s)} + t\mathbf{d}_{\mathcal{G}}^{(k)}\Big) \right].$$

For the DNN, predictive NLL is evaluated using its single deterministic weight vector. The loss increase for each direction is measured relative to the corresponding unperturbed model:

$$\Delta\mathcal{L}(t; \mathcal{G}, k) = \widehat{\mathcal{L}}_{\text{test}}^{\text{BMA}}(t; \mathcal{G}, k) - \widehat{\mathcal{L}}_{\text{test}}^{\text{BMA}}(0).$$

Figure 1 reports these curves after averaging over the $K = 10$ directions. We quantify endpoint flatness using the symmetric predictive-NLL (Feng et al., 2026) increase at $t = \pm 0.2$:

$$R_{\mathcal{G}} = \frac{1}{2K} \sum_{k=1}^{K} \left[\Delta\mathcal{L}(-0.2; \mathcal{G}, k) + \Delta\mathcal{L}(0.2; \mathcal{G}, k) \right].$$

Smaller $R_{\mathcal{G}}$ indicates a flatter and less perturbation-sensitive predictive landscape. Table A1 quantitatively supports the landscapes shown in Figure 1. *Ours* achieves the smallest mean symmetric NLL rise for every architecture and sensitivity group. The absolute difference between *Ours* and *Opposite* is largest in the top-sensitivity group for both CNN and ViT, showing that the sensitivity–precision ordering has its strongest effect along directions containing sensitive parameters. The bottom-sensitivity groups are flatter overall and exhibit smaller differences between the methods. The shuffled allocation generally lies between the two ordered allocations, indicating that the correspondence between sensitivity and prior precision affects the resulting predictive geometry.

A.3 Warm-Start Posterior Initialization under MFVI

Since all prior configurations have access to the same pretrained solution $\mathbf{w}^*$, we additionally examine whether their predictive behavior depends on the initialization of the variational posterior. In the main experiments, we use the default variational initialization of Bayesian-torch (Krishnan et al., 2022). Specifically, for both weights and biases, the variational parameters are initialized as

$$\mu_i^{(0)} \sim \mathcal{N}(0, 0.1^2), \qquad \rho_i^{(0)} \sim \mathcal{N}(-3, 0.1^2), \qquad \sigma_i^{(0)} = \log\left(1 + \exp(\rho_i^{(0)})\right).$$

We further consider two posterior warm-start configurations based on by MOPED (Krishnan et al., 2020). In the first configuration, denoted MOPED, we initialize only the variational mean at the pretrained solution while using a fixed initial posterior scale,

$$\mu_i^{(0)} = w_i^*, \qquad \rho_i^{(0)} = -3, \qquad \sigma_i^{(0)} = \log\left(1 + \exp(-3)\right).$$

This configuration evaluates warm-starting the variational mean at $\mathbf{w}^*$ while using a fixed initial posterior scale. In the second configuration, denoted MOPED*, we additionally initialize the posterior scale according to the magnitude of the pretrained weight,

$$\mu_i^{(0)} = w_i^*, \qquad \sigma_i^{(0)} = 0.1|w_i^*|.$$

Thus, the comparison from Random to MOPED examines the effect of warm-starting the posterior mean, whereas the comparison from MOPED to MOPED* further examines the effect of a

Table A2: Sensitivity of the posterior predictive decomposition to variational posterior initialization. The upper and lower blocks correspond to CIFAR-10 and ImageNet-100, respectively. Here, $\Delta_{\mathrm{BMA}} = L_{\mathrm{test}}(\mu) - L_{\mathrm{test}}^{\mathrm{BMA}}$.

		CNN				ViT			
Init.	Method	$L_{\mathrm{test}}(\mu)\downarrow$	$L_{\mathrm{test}}(\mu,\sigma)\downarrow$	$L_{\mathrm{test}}^{\mathrm{BMA}}\downarrow$	Δ_{BMA}	$L_{\mathrm{test}}(\mu)\downarrow$	$L_{\mathrm{test}}(\mu,\sigma)\downarrow$	$L_{\mathrm{test}}^{\mathrm{BMA}}\downarrow$	Δ_{BMA}
Random	$\lambda_{\min}\mathbf{I}$	.3586±.0131	.7091±.0207	.3900±.0132	−.0314	1.1340±.0586	1.1603±.0423	1.0793±.0444	+.0547
	$\lambda_{\max}\mathbf{I}$	.3550±.0043	.3604±.0043	.3382±.0039	+.0169	.6196±.0202	.6199±.0173	.5979±.0148	+.0217
	EB-Global	.4933±.0448	1.0612±.0333	.6308±.0183	−.1375	1.1726±.0491	1.1841±.0505	1.1043±.0529	+.0682
	EB-Diag	.3001±.0128	1.1295±.0433	.5333±.0096	−.2333	.9996±.0607	1.0415±.0538	.9546±.0591	+.0450
	Ours	.3352±.0027	.4761±.0067	.2473±.0023	+.0879	.6091±.0018	.6323±.0018	.5519±.0022	+.0572
MOPED	$\lambda_{\min}\mathbf{I}$	.3582±.0093	.6898±.0086	.3742±.0115	−.0160	.5339±.0060	.6144±.0021	.4844±.0027	+.0495
	$\lambda_{\max}\mathbf{I}$	.3529±.0059	.3587±.0059	.3376±.0051	+.0153	.6250±.0072	.6253±.0068	.6016±.0082	+.0234
	EB-Global	.3868±.0830	1.0475±.0951	.5675±.0502	−.1807	.5339±.0060	.6144±.0021	.4844±.0027	+.0495
	EB-Diag	.2902±.0065	1.0958±.0416	.5172±.0069	−.2270	.5369±.0032	.6210±.0012	.4873±.0014	+.0496
	Ours	.3342±.0033	.4765±.0109	.2484±.0030	+.0859	.6102±.0040	.6335±.0033	.5536±.0049	+.0566
MOPED*	$\lambda_{\min}\mathbf{I}$	.4218±.0359	.4298±.0366	.4066±.0302	+.0151	.6332±.0067	.6408±.0061	.6260±.0069	+.0072
	$\lambda_{\max}\mathbf{I}$	.3577±.0046	.3617±.0044	.3467±.0045	+.0110	.6314±.0036	.6327±.0034	.6124±.0034	+.0190
	EB-Global	.3465±.0053	.3828±.0066	.2785±.0060	+.0680	.6332±.0067	.6408±.0061	.6260±.0069	+.0072
	EB-Diag	.3282±.0031	.3672±.0031	.2664±.0040	+.0618	.6359±.0110	.6452±.0114	.6252±.0106	+.0107
	Ours	.3410±.0046	.3893±.0030	.2684±.0042	+.0727	.6201±.0050	.6298±.0051	.5893±.0047	+.0308
Random	$\lambda_{\min}\mathbf{I}$	1.1990±.0132	1.7450±.0117	1.3129±.0154	−.1139	2.2364±.0498	2.3689±.0396	2.2258±.0399	+.0106
	$\lambda_{\max}\mathbf{I}$	.6290±.0014	.6441±.0012	.6087±.0009	+.0203	1.8569±1.3340	1.9236±1.3524	1.8958±1.3611	−.0389
	EB-Global	1.2441±.0318	1.9770±.0367	1.4266±.0298	−.1824	2.2944±.0158	2.4200±.0095	2.2706±.0088	+.0238
	EB-Diag	1.0258±.0091	1.9441±.0309	1.3274±.0183	−.3016	2.4669±.0296	2.6978±.0269	2.5169±.0288	−.0501
	Ours	.6302±.0014	.7844±.0056	.6059±.0015	+.0242	.3830±.0048	.4785±.0046	.4078±.0039	−.0248
MOPED	$\lambda_{\min}\mathbf{I}$	.6459±.0050	.9575±.0069	.6413±.0031	+.0045	1.9051±.0217	2.0271±.0229	1.8806±.0238	+.0245
	$\lambda_{\max}\mathbf{I}$	.6290±.0018	.6395±.0017	.6102±.0012	+.0188	2.1888±1.0619	2.2517±1.0685	2.2243±1.0762	−.0355
	EB-Global	.6982±.0094	1.2581±.0054	.8092±.0031	−.1111	1.9233±.0287	2.0655±.0225	1.9077±.0229	+.0156
	EB-Diag	.7062±.0064	1.2879±.0065	.8370±.0069	−.1308	2.0663±.0425	2.3006±.0427	2.1076±.0428	−.0414
	Ours	.6327±.0043	.7479±.0050	.5983±.0025	+.0344	.3710±.0032	.4348±.0025	.3811±.0029	−.0100
MOPED*	$\lambda_{\min}\mathbf{I}$	.6448±.0033	.6652±.0034	.6223±.0033	+.0225	.7113±.0092	.7259±.0086	.6832±.0082	+.0282
	$\lambda_{\max}\mathbf{I}$	.6227±.0020	.6247±.0021	.6185±.0020	+.0042	.4085±.0112	.4587±.0173	.4197±.0147	−.0112
	EB-Global	.6330±.0045	.6591±.0047	.6082±.0051	+.0249	.6874±.0075	.7029±.0074	.6600±.0071	+.0275
	EB-Diag	.6270±.0053	.6517±.0051	.6056±.0045	+.0214	.4905±.0049	.5179±.0037	.4753±.0034	+.0153
	Ours	.6211±.0023	.6629±.0030	.5988±.0018	+.0223	.3527±.0041	.3951±.0032	.3572±.0025	−.0044

parameter-dependent posterior-scale initialization. Table A2 shows that the effect of posterior initialization depends on the prior configuration under MFVI. The EB variants are particularly sensitive: initializing only the variational mean (MOPED) or additionally using a weight-scaled posterior variance (MOPED*) can substantially change their $\mathcal{L}_{\mathrm{test}}^{BMA}$, and the favorable choice depends on the architecture and datasets. For the CIFAR-10 ViT, initializing the variational mean at $\mathbf{w}^*$ substantially improves the EB variants, whereas additionally initializing the posterior scale with MOPED* reduces this improvement.

In contrast, the proposed prior exhibits considerably smaller variation across the initializations. Although a favorable initialization can make EB outperform the proposed prior in individual CIFAR-10 settings, the proposed prior remains consistently competitive and achieves the best BMA NLL across all initializations on ImageNet-100. These results suggest that sensitivity-guided precision allocation reduces the dependence of posterior predictive performance on variational initialization.

A.4 Efficient Approximations of Pruning Sensitivity

We use the local quadratic pruning criterion in Eq. 8 because it estimates parameter importance from the local loss change induced by weight removal, following conventional pruning methods (LeCun et al., 1989; Hassibi & Stork, 1992; Molchanov et al., 2019).

Table A3: Ablation of sensitivity criteria on CIFAR-10. Here, abs(w), 1st, and 1st+2nd denote $|w_i^*|$, $|g_i w_i^*|$, and $\left|-g_i w_i^* + \frac{1}{2} G_{ii} (w_i^*)^2\right|$, respectively. ρ denotes the Spearman correlation with the combined ranking. Cost reports the mean wall-clock time required to compute the sensitivity scores on a single RTX 4090.

	Sensitivity	Acc ↑	NLL ↓	ECE ↓	AUROC ↑	ρ ↑	Cost [sec] ↓
CNN	abs(w)	.916±.001	.253±.004	.017±.001	.861±.001	.724	0.002
	1st	.917±.001	.247±.001	.006±.002	.860±.002	.967	15.12
	1st+2nd	.916±.001	.247±.002	.007±.001	.861±.002	1.000	142.0
ViT	abs(w)	.813±.003	.562±.006	.034±.002	.762±.001	.787	0.012
	1st	.814±.002	.555±.006	.034±.001	.766±.003	.999	32.49
	1st+2nd	.814±.001	.552±.002	.034±.002	.767±.001	1.000	356.0

Estimating the diagonal GGN curvature, however, introduces additional computational cost. We therefore examine whether the combined criterion can be approximated using the first-order term alone. At a MAP solution, the first-order term can be further related to weight magnitude. For an L_2-regularized objective with a shared coefficient γ, the stationary condition gives $g_i + \gamma w_i^* \approx 0$, and consequently $|g_i w_i^*| \approx \gamma (w_i^*)^2$. Thus, under this assumption, the ordering induced by the first-order criterion can be recovered from $|w_i^*|$ alone, motivating magnitude as an approximation that requires neither gradients nor curvature estimates.

Table A3 compares the combined criterion with these two computationally cheaper alternatives. The first-order criterion achieves predictive performance comparable to the combined criterion and closely preserves its allocation ordering, with rank correlations of $\rho = .967$ for the CNN and $\rho = .999$ for the ViT. Although magnitude exhibits lower rank correlations of $\rho = .724$ and $\rho = .787$, it still achieves competitive predictive performance across both architectures. These results suggest that the ordering derived from pruning sensitivity for prior precision allocation can be approximated effectively by inexpensive importance proxies, with weight magnitude providing a particularly efficient alternative when computational cost is a concern.

A.5 Sensitivity to the Optimization and Prior Reference Scale

Table A4: Sensitivity to the maximum learning rate $\eta_{\max}$ and the corresponding upper prior precision $\lambda_{\max} = N/\eta_{\max}$ on CIFAR-10. For $\eta_{\max} = 0.1$ and 0.01, the corresponding upper reference precisions are 4.5×10^5 and 4.5×10^6, respectively. OOD detection is evaluated on TinyImageNet.

		$\eta_{\max} = 0.1$					$\eta_{\max} = 0.01$				
	Prior	Acc. ↑	NLL ↓	ECE ↓	AUROC ↑	FPR95 ↓	Acc. ↑	NLL ↓	ECE ↓	AUROC ↑	FPR95 ↓
CNN	$\lambda_{\min}\mathbf{I}$	.883±.004	.390±.013	.083±.006	.840±.005	.664±.010	.777±.006	.767±.014	.171±.005	.754±.004	.800±.006
	$\lambda_{\max}\mathbf{I}$	.915±.001	.338±.004	.050±.001	.852±.001	.685±.004	.911±.002	.291±.005	.035±.003	.847±.002	.661±.006
	EB-Global	.829±.013	.631±.018	.172±.017	.799±.014	.733±.020	.763±.013	.858±.030	.212±.009	.747±.016	.804±.024
	EB-Diagonal	.881±.003	.533±.010	.202±.007	.831±.004	.700±.010	.813±.010	.799±.033	.263±.007	.756±.016	.794±.020
	Ours	.916±.001	.247±.002	.007±.001	.861±.002	.632±.002	.915±.001	.247±.002	.008±.001	.860±.001	.637±.005
ViT	$\lambda_{\min}\mathbf{I}$	.615±.014	1.079±.044	.045±.005	.675±.006	.869±.011	.545±.008	1.272±.014	.056±.004	.629±.004	.907±.002
	$\lambda_{\max}\mathbf{I}$	.807±.002	.598±.015	.053±.010	.759±.002	.784±.002	.802±.005	.608±.006	.051±.005	.758±.003	.789±.008
	EB-Global	.605±.019	1.104±.053	.042±.004	.667±.014	.875±.015	.545±.008	1.272±.014	.056±.004	.629±.004	.907±.002
	EB-Diagonal	.660±.021	.955±.059	.049±.002	.697±.012	.836±.017	.552±.006	1.255±.013	.056±.003	.636±.004	.901±.007
	Ours	.814±.001	.552±.002	.034±.002	.767±.001	.771±.001	.809±.002	.568±.004	.029±.004	.763±.001	.777±.004

Since the upper reference precision is defined as $\lambda_{\max} = N/\eta_{\max}$, changing $\eta_{\max}$ simultaneously alters both the optimization scale and the prior precision range. We therefore compare the default setting $\eta_{\max} = 0.1$ with $\eta_{\max} = 0.01$, corresponding to $\lambda_{\max} = 4.5 \times 10^5$ and $\lambda_{\max} = 4.5 \times 10^6$, respectively. Despite this increase of one order of magnitude in the upper reference precision, the proposed prior maintains nearly unchanged predictive and OOD performance for CNNs and shows only modest changes for ViTs, while providing the strongest overall performance across the reported metrics under both settings. In contrast, $\lambda_{\min}\mathbf{I}$, EB-Global, and EB-Diagonal show substantially larger performance degradation, particularly for CNNs, whereas $\lambda_{\max}\mathbf{I}$ remains relatively stable but yields consistently lower performance than the proposed prior. These results indicate that

the proposed parameter specific precision allocation is relatively robust to changes in the coupled optimization and prior reference scale.

A.6 Comparison with Tuned Isotropic Priors

Table A5: Effect of global prior precision on CIFAR-10. We sweep isotropic prior precisions between $\lambda_{\min}$ and $\lambda_{\max}$ under the default random posterior initialization. OOD detection is evaluated on TinyImageNet.

	CNN					ViT				
Prior	Acc. ↑	NLL ↓	ECE ↓	AUROC ↑	FPR95 ↓	Acc. ↑	NLL ↓	ECE ↓	AUROC ↑	FPR95 ↓
$\lambda_{\min}\mathbf{I}$	.883±.004	.390±.013	.083±.006	.840±.005	.664±.010	.615±.014	1.079±.044	.045±.005	.675±.006	.869±.011
λ=75k	.914±.001	.282±.005	.034±.002	.856±.001	.642±.003	.810±.002	.588±.005	.055±.001	.760±.004	.779±.009
λ=150k	.916±.001	.305±.004	.043±.001	.853±.002	.650±.007	.802±.006	.605±.011	.055±.011	.758±.004	.790±.007
λ=225k	.916±.001	.319±.002	.046±.001	.853±.001	.667±.003	.809±.002	.599±.006	.056±.007	.759±.001	.780±.003
λ=300k	.916±.001	.329±.002	.048±.001	.852±.002	.676±.005	.808±.003	.600±.005	.058±.006	.759±.002	.783±.004
λ=375k	.915±.003	.335±.005	.049±.000	.851±.002	.680±.007	.805±.008	.604±.012	.058±.008	.760±.004	.782±.007
$\lambda_{\max}\mathbf{I}$	.915±.001	.338±.004	.050±.001	.852±.002	.685±.004	.807±.002	.598±.015	.053±.010	.759±.002	.784±.002
Ours	.916±.001	.247±.002	.007±.001	.861±.002	.632±.002	.814±.001	.552±.002	.034±.002	.767±.001	.771±.001

To examine whether the gains of the proposed prior can be reproduced by tuning a single global precision, we sweep intermediate isotropic precisions between $\lambda_{\min}$ and $\lambda_{\max}$ while keeping the MFVI optimization protocol fixed. For the CNN, tuning the isotropic precision substantially improves over the two endpoint configurations, but none of the tested values matches the NLL, ECE, AUROC, or FPR95 of the proposed parameter-wise allocation. The best isotropic accuracy is effectively identical to Ours. For the ViT, Ours outperforms all tested isotropic precisions across all reported metrics. These results indicate that the observed gains cannot be explained solely by selecting a better global precision scale.

A.7 Pretrained Deterministic References

Table A6 reports the predictive performance of the pretrained deterministic solutions used as empirical prior means. The pretrained deterministic models remain competitive with, and in some settings outperform, their MFVI counterparts. Such behavior has also been observed in deep Bayesian neural networks, where predictive performance can be highly sensitive to prior specification and posterior inference (Wenzel et al., 2020b). This further motivates the need for informative prior design when converting pretrained deterministic networks into BNNs.

Table A6: Predictive performance of the pretrained deterministic solutions used as empirical prior means.

Dataset	Model	Acc. ↑	NLL ↓	ECE ↓
CIFAR-10	ResNet-20	.916	.274	.035
	ViT	.791	.625	.054
ImageNet-100	ResNet-18	.845	.589	.028
	DeiT-Tiny	.900	.392	.061
ImageNet-1K	ResNet-50	.801	1.422	.407
	DeiT-Tiny	.721	1.219	.119