SCALE-INVARIANT BAYESIAN NEURAL NETWORKS WITH CONNECTIVITY TANGENT KERNEL

Anonymous authors
Paper under double-blind review

ABSTRACT

Explaining generalizations and preventing over-confident predictions are central goals of studies on the loss landscape of neural networks. Flatness, defined as loss invariability on perturbations of a pre-trained solution, is widely accepted as a predictor of generalization in this context. However, the problem that flatness and generalization bounds can be changed arbitrarily according to the scale of a parameter was pointed out, and previous studies partially solved the problem with restrictions: Counter-intuitively, their generalization bounds were still variant for the function-preserving parameter scaling transformation or limited only to an impractical network structure. As a more fundamental solution, we propose new prior and posterior distributions invariant to scaling transformations by *decomposing* the scale and connectivity of parameters, thereby allowing the resulting generalization bound to describe the generalizability of a broad class of networks with the more practical class of transformations such as weight decay with batch normalization. We also show that the above issue adversely affects the uncertainty calibration of Laplace approximation and propose a solution using our invariant posterior. We empirically demonstrate our posterior provides effective flatness and calibration measures with low complexity in such a practical parameter transformation case, supporting its practical effectiveness in line with our rationale.

1 INTRODUCTION

Though neural networks (NNs) have experienced extraordinary success, understanding the generalizability of NNs and successfully using them in real-world contexts still faces a number of obstacles (Kendall & Gal, 2017; Ovadia et al., 2019). It is a well-known enigma, for instance, why such NNs generalize well and do not suffer from overfitting (Neyshabur et al., 2015b; Zhang et al., 2017; Arora et al., 2018). Recent research on the loss landscape of NNs seeks to reduce these obstacles. Hochreiter & Schmidhuber (1995) proposed a theory known as flat minima (FM): the flatness of local minima (i.e., loss invariability w.r.t. parameter perturbations) is positively correlated with network generalizability, as empirically demonstrated by Jiang et al. (2020). Concerning overconfidence, MacKay (1992) suggested an approximated Bayesian posterior using the curvature information of local minima, and Daxberger et al. (2021a) underlined its practical utility.

Nonetheless, the limitations of the FM hypothesis were pointed out by Dinh et al. (2017); Li et al. (2018). By rescaling two successive layers, Dinh et al. (2017) demonstrated it was possible to modify a flatness measure without modifying the functions, hence allowing extraneous variability to be captured in the computation of generalizability. Meanwhile, Li et al. (2018) argued that weight decayregularization (Loshchilov & Hutter, 2019) is an important limitation of the FM hypothesis as it leads to a contradiction of the FM hypothesis in practice; the weight decay sharpens the pre-trained solutions of NNs by downscaling the parameters, whereas the weight decay actually improves the generalization performance of NNs in general cases (Zhang et al., 2019). In short, they suggest that scaling transformation on network parameters (e.g., re-scaling layers and weight decay) may lead to a contradiction of the FM hypothesis.

To resolve this contradiction, we investigate PAC-Bayesian prior and posterior distributions to derive a new **scale-invariant generalization bound**. Unlike related works (Tsuzuku et al., 2020; Kwon et al., 2021), our bound guarantees the invariance for a general class of function-preserving parameter scaling transformation with a broad class of networks (Petzka et al., 2021) (Secs. 2.2 and 2.3).

This bound is derived from the scale invariance of the prior and poster distributions, which guarantees not only the scale invariance of the bound but also its substance the Kullback-Leibler (KL) divergence-based kernel; we named this new term with scale-invariance property as empirical Connectivity Tangent Kernel (CTK) as it can be considered as a modification of empirical Neural Tangent Kernel (Jacot et al., 2018). Consequently, we define a novel sharpness metric named **Connectivity Sharpness (CS)** as a trace of CTK. Empirically, we verify our CS has a better prediction for generalization performance of NNs than existing sharpness measures (Liang et al., 2019; Keskar et al., 2017; Neyshabur et al., 2017) with a low-complexity (Sec. 2.5), with confirming its stronger correlation to generalization error (Sec. 4.1).

We also found the contradiction of the FM hypothesis turns into meaningless predictive uncertainty amplifying issues in the Bayesian NN regime (Sec. 3.1), and can alleviate this issue by using Bayesian NN based on the posterior distribution of our PAC-Bayesian analysis. We call the resulting Bayesian NN as **Connectivity Laplace (CL)**, as it can be seen as a variation of Laplace Approximation (LA; MacKay (1992)) using different Jacobian. In particular, we provide pitfalls of weight decay regularization with BN in LA and its remedy using our posterior (Sec. 3.1) to suggest practical utilities of our Bayesian NNs (Sec. 4.2). We summarize our contributions as follows:

- Unlike related studies, our resulting (PAC-Bayes) generalization bound guarantees the invariance for a general class of function-preserving parameter scaling transformation with a broad class of networks (Sec. 2.2 and 2.3). Based on this novel PAC-Bayes bound, we propose a low-complexity sharpness measure (Sec. 2.5).
- We provide pitfalls of weight decay regularization with BN in LA and its remedy using our posterior (Sec. 3.1).
- We empirically confirm the strong correlation between generalization error and our sharpness metric (Sec. 4.1) and visualize pitfalls of weight decay with LA in synthetic data and practical utilities of our Bayesian NNs (Sec. 4.2).

2 PAC-Bayes bound with scale-invariance

2.1 Background

Setup and Definitions We consider a Neural Network (NN), $f(\cdot,\cdot) : \mathbb{R}^D \times \mathbb{R}^P \to \mathbb{R}^K$, given input $x \in \mathbb{R}^D$ and network parameter $\theta \in \mathbb{R}^P$. Hereafter, we consider one dimensional vector a one-dimensional vector as a single column matrix unless otherwise stated. We use the output of NN $f(x,\theta)$ as a prediction for input x. Let $\mathcal{S} := \{(x_n, y_n)\}_{n=1}^N$ denote the independently and identically distributed (i.i.d.) training data drawn from true data distribution $\mathcal{D}$, where $x_n \in \mathbb{R}^D$ and $y_n \in \mathbb{R}^K$ are input and output representation of n-th training instance, respectively. For simplicity, we denote concatenated input and output of all instances as $\mathcal{X} := \{x : (x,y) \in \mathcal{S}\}$ and $\mathcal{Y} := \{y : (x,y) \in \mathcal{S}\}$, respectively and $f(\mathcal{X},\theta) \in \mathbb{R}^{NK}$ as a concatenation of $\{f(x_n,\theta)\}_{n=1}^N$. Given a prior distribution of network parameter $p(\theta)$ and a likelihood function $p(\mathcal{S}|\theta) := \prod_{n=1}^N p(y_n|x_n,\theta) := \prod_{n=1}^N p(y_n|f(x_n,\theta))$, Bayesian inference defines posterior distribution of network parameter θ as $p(\theta|\mathcal{S}) = \frac{1}{Z(\mathcal{S})}\exp(-\mathcal{L}(\mathcal{S},\theta)) := \frac{1}{Z(\mathcal{S})}p(\theta)p(\mathcal{S}|\theta), Z(\mathcal{S}) := \int p(\theta)p(\mathcal{S}|\theta)d\theta$ where $\mathcal{L}(\mathcal{S},\theta) := -\log p(\theta) - \sum_{n=1}^N \log p(y_n|x_n,\theta)$ is training loss and $Z(\mathcal{S})$ is normalizing factor. For example, the likelihood function for regression task will be Gaussian: $p(y|x,\theta) = \mathcal{N}(y|f(x,\theta),\sigma^2\mathbf{I}_k)$ where σ is (homoscedastic) observation noise scale. For classification task, we treat it as a one-hot regression task following Lee et al. (2019a) and He et al. (2020). While we applied this modification for theoretical tractability, Lee et al. (2020); Hui & Belkin (2021) showed this modification offers reasonable performance competitive to the cross-entropy loss. Details on this treatment is given in Appendix B.

Laplace Approximation In general, the exact computation for the Bayesian posterior of a network parameter is intractable. The Laplace Approximation (LA; MacKay (1992)) proposes to approximate the posterior distribution with a Gaussian distribution defined as $p_{\mathrm{LA}}(\psi|\mathcal{S}) \sim \mathcal{N}(\psi|\theta^*, (\nabla^2_\theta \mathcal{L}(\mathcal{S},\theta^*))^{-1})$ where $\theta^* \in \mathbb{R}^P$ is a pre-trained parameter with training loss and $\nabla^2_\theta \mathcal{L}(\mathcal{S},\theta^*) \in \mathbb{R}^{P\times P}$ is Hessian of loss function w.r.t. parameter at θ^*.

Recent works on LA replace the Hessian matrix with (Generalized) Gauss-Newton matrix to make computation tractable (Khan et al., 2019; Immer et al., 2021). With this approximation, the LA

posterior of regression problem can be represented as:

$$p_{\mathrm{LA}}(\psi|\mathcal{S}) \sim \mathcal{N}(\psi|\theta^*, (\underbrace{\mathbf{I}_P/\alpha^2}_{\text{Damping}} + \underbrace{\mathbf{J}_\theta^\top \mathbf{J}_\theta/\sigma^2}_{\text{Curvature}})^{-1}) \tag{1}$$

where $\alpha, \sigma > 0$ and $\mathbf{I}_P \in \mathbb{R}^{P\times P}$ is a identity matrix and $\mathbf{J}_\theta \in \mathbb{R}^{NK\times P}$ is a concatenation of $\mathbf{J}_\theta(x, \theta^*) \in \mathbb{R}^{K\times P}$ (Jacobian of f w.r.t. θ at input x and parameter θ^*) along training input $\mathcal{X}$. Since covariance of equation 1 is inverse of $P \times P$ matrix, further sub-curvature approximation was considered including diagonal, Kronecker-factored approximate curvature (KFAC), last-layer, and sub-network (Ritter et al., 2018; Kristiadi et al., 2020; Daxberger et al., 2021b). Furthermore, they found that proper selection of prior scale α is needed to balance the dilemma between overconfidence and underfitting in LA.

PAC-Bayes bound with data-dependent prior We consider a PAC-Bayes generalization error bound of classification task used in McAllester (1999); Perez-Ortiz et al. (2021) (especially, equation (7) of Perez-Ortiz et al. (2021)). Let $\mathbb{P}$ be any PAC-Bayes prior distribution over $\mathbb{R}^P$ independent of training dataset $\mathcal{S}$ and $\mathrm{err}(\cdot,\cdot) : \mathbb{R}^{K\times K} \to [0, 1]$ be a error function which is defined separately from the loss function. For any constant $\delta \in (0, 1]$ and $\lambda > 0$, and any PAC-Bayes posterior distribution $\mathbb{Q}$ over $\mathbb{R}^P$, the following holds with probability at least $1-\delta$: $\mathrm{err}_{\mathcal{D}}(\mathbb{Q}) \leq \mathrm{err}_{\mathcal{S}}(\mathbb{Q}) + \sqrt{\frac{\mathrm{KL}[\mathbb{Q}\|\mathbb{P}]+\log(2\sqrt{N}/\delta)}{2N}}$ where $\mathrm{err}_{\mathcal{D}}(\mathbb{Q}) := \mathbb{E}_{(x,y)\sim\mathcal{D},\theta\sim\mathbb{Q}}[\mathrm{err}(f(x,\theta), y)]$, $\mathrm{err}_{\mathcal{S}}(\mathbb{Q}) := \mathbb{E}_{(x,y)\sim\mathcal{S},\theta\sim\mathbb{Q}}[\mathrm{err}(f(x,\theta), y)]$, and N denotes the cardinality of $\mathcal{S}$. That is, $\mathrm{err}_{\mathcal{D}}(\mathbb{Q})$ and $\mathrm{err}_{\mathcal{S}}(\mathbb{Q})$ are generalization error and empirical error, respectively. The only restriction on $\mathbb{P}$ here is that it *cannot depend* on the dataset S.

Following the recent discussion in Perez-Ortiz et al. (2021), one can construct data-dependent PAC-Bayes bounds by (i) randomly partitioning dataset $\mathcal{S}$ into $\mathcal{S}_{\mathbb{Q}}$ and $\mathcal{S}_{\mathbb{P}}$ so that they are independent, (ii) using a PAC-Bayes prior distribution $\mathbb{P}_{\mathcal{D}}$ only dependent of $\mathcal{S}_{\mathbb{P}}$ (i.e., independent of $\mathcal{S}_{\mathbb{Q}}$ so $\mathbb{P}_{\mathcal{D}}$ belongs to $\mathbb{P}$), (iii) using a PAC-Bayes posterior distribution $\mathbb{Q}$ dependent of entire dataset $\mathcal{S}$, and (iv) computing empirical error $\mathrm{err}_{\mathcal{S}_{\mathbb{Q}}}(\mathbb{Q})$ with target subset $\mathcal{S}_{\mathbb{Q}}$ (not entire dataset $\mathcal{S}$). In summary, one can modify the aforementioned original PAC-Bayes bound through our data-dependent prior $\mathbb{P}_c$ as

$$\mathrm{err}_{\mathcal{D}}(\mathbb{Q}) \leq \mathrm{err}_{\mathcal{S}_{\mathbb{Q}}}(\mathbb{Q}) + \sqrt{\frac{\mathrm{KL}[\mathbb{Q}\|\mathbb{P}_{\mathcal{D}}] + \log(2\sqrt{N_{\mathbb{Q}}}/\delta)}{2N_{\mathbb{Q}}}} \tag{2}$$

where $N_{\mathbb{Q}}$ is the cardinality of $\mathcal{S}_{\mathbb{Q}}$. We denote sets of input and output of splitted datasets $(\mathcal{S}_{\mathbb{P}}, \mathcal{S}_{\mathbb{Q}})$ as $\mathcal{X}_{\mathbb{P}}, \mathcal{Y}_{\mathbb{P}}, \mathcal{X}_{\mathbb{Q}}, \mathcal{Y}_{\mathbb{Q}}$ for simplicity.

2.2 Design of PAC-Bayes prior and posterior

Our goal is to construct scale-invariant $\mathbb{P}_{\mathcal{D}}$ and $\mathbb{Q}$. To this end, we first assume a pre-trained parameter $\theta^* \in \mathbb{R}^P$ of the negative log-likelihood function with $\mathcal{S}_{\mathbb{P}}$. This parameter can be attained with standard NN optimization procedures (e.g., stochastic gradient descent (SGD) with momentum). Then, we consider linearized NN at the pre-trained parameter with the auxiliary variable $c \in \mathbb{R}^P$ as

$$g^{\mathrm{lin}}_{\theta^*}(x, c) := f(x, \theta^*) + \mathbf{J}_\theta(x, \theta^*)\mathrm{diag}(\theta^*)c \tag{3}$$

where diag is a vector-to-matrix diagonal operator. Note that equation 3 is the first-order Taylor approximation (i.e., linearization) of NN with perturbation $\theta^* \odot c$ given input x and parameter θ^*: $g^{\mathrm{pert}}_{\theta^*}(x, c) := f(x, \theta^* + \theta^* \odot c) = f(x, \theta^* + \mathrm{diag}(\theta^*)c) \approx g^{\mathrm{lin}}_{\theta^*}(x, c)$, where $\odot$ denotes element-wise multiplication (Hadamard product) of two vectors. Here we write the perturbation in parameter space as $\theta^* \odot c$ instead of single variable such as $\delta \in \mathbb{R}^P$. This design of linearization matches the scale of perturbation (i.e., $\mathrm{diag}(\theta^*) \odot c$) to the scale of θ^* in a component-wise manner. Similar idea was proposed in the context of pruning at initialization (Lee et al., 2019c;b) to measure the importance of each connection independently of its weight. In this context, our perturbation can be viewed as **perturbation in connectivity space by decomposing the scale and connectivity of parameter**.

Based on this, we define a data-dependent prior ($\mathbb{P}_{\mathcal{D}}$) over connectivity as

$$\mathbb{P}_{\theta^*}(c) := \mathcal{N}(c \,|\, \mathbf{0}_P, \alpha^2\mathbf{I}_p). \tag{4}$$

This distribution can be translated to a distribution over parameter by considering the distribution of perturbed parameter ($\psi := \theta^* + \theta^* \odot c$): $\mathbb{P}_{\theta^*}(\psi) := \mathcal{N}(\psi \,|\, \theta^*, \alpha^2\mathrm{diag}(\theta^*)^2)$. We now define the

PAC-Bayes posterior over connectivity $\mathbb{Q}(c)$ as follows:

$$\mathbb{Q}_{\theta^*}(c) := \mathcal{N}(c|\mu_{\mathbb{Q}}, \Sigma_{\mathbb{Q}}) \tag{5}$$

$$\mu_{\mathbb{Q}} := \frac{\Sigma_{\mathbb{Q}}\mathbf{J}_c^\top\left(\mathcal{Y} - f(\mathcal{X}, \theta^*)\right)}{\sigma^2} = \frac{\Sigma_{\mathbb{Q}}\mathrm{diag}(\theta^*)\mathbf{J}_\theta^\top\left(\mathcal{Y} - f(\mathcal{X}, \theta^*)\right)}{\sigma^2} \tag{6}$$

$$\Sigma_{\mathbb{Q}} := \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_c^\top\mathbf{J}_c}{\sigma^2}\right)^{-1} = \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathrm{diag}(\theta^*)\mathbf{J}_\theta^\top\mathbf{J}_\theta\mathrm{diag}(\theta^*)}{\sigma^2}\right)^{-1} \tag{7}$$

where $\mathbf{J}_c \in NK \times P$ is a concatenation of $\mathbf{J}_c(x, \mathbf{0}_P) := \mathbf{J}_\theta(x, \theta^*)\mathrm{diag}(\theta^*) \in \mathbb{R}^{K\times P}$ (i.e., Jacobian of perturbed NN $g^{\mathrm{pert}}_{\theta^*}(x, c)$ w.r.t. c at input x and connectivity $\mathbf{0}_P$) along training input $\mathcal{X}$. Our PAC-Bayes posterior $\mathbb{Q}_{\theta^*}$ is *the posterior of Bayesian linear regression problem w.r.t. connectivity* $c : y_i = f(x_i, \theta^*) + \mathbf{J}_\theta(x_i, \theta^*)\mathrm{diag}(\theta^*)c + \epsilon_i$ where $(x_i, y_i) \in \mathcal{S}$ and ϵ_i are i.i.d. samples of $\mathcal{N}(\epsilon_i|\mathbf{0}_K, \sigma^2\mathbf{I}_K)$. Again, it is equivalent to the posterior distribution over parameter $\mathbb{Q}_{\theta^*}(\psi) = \mathcal{N}\left(\psi|\theta^* + \theta^* \odot \mu_{\mathbb{Q}}, (\mathrm{diag}(\theta^*)^{-2}/\alpha^2 + \mathbf{J}_\theta^\top\mathbf{J}_\theta/\sigma^2)^{-1}\right)$ where $\mathrm{diag}(\theta^*)^{-2} := (\mathrm{diag}(\theta^*)^{-1})^2$ by assuming that all components of θ^* are non-zero. Note that this assumption can be easily satisfied by considering the prior and posterior distribution of non-zero components of NNs only. Although we choose this restriction for theoretical tractability, future work can modify this choice to achieve diverse predictions by considering the distribution of zero coordinates. We refer to Appendix C for detailed derivations of $\mathbb{Q}_{\theta^*}(c)$ and $\mathbb{Q}_{\theta^*}(\psi)$.

Remark 2.1 (Two-phase training)**.** *The prior distribution in equation 4 is data-dependent priors since they depend on the pre-trained parameter θ^* optimized on training dataset $\mathcal{S}_{\mathbb{P}}$. On the other hand, posterior distribution in equation 5 depend on both $\mathcal{S}_{\mathbb{P}}$ (through θ^*) and $\mathcal{S}_{\mathbb{Q}}$ (through Bayesian linear regression). Intuitively, one attain the PAC-Bayes posterior $\mathbb{Q}$ with two-phase training: pre-train with $\mathcal{S}_{\mathbb{P}}$ and Bayesian fine-tuning with $\mathcal{S}_{\mathbb{Q}}$. A similar idea of linearized fine-tuning was proposed in the context of transfer learning in Achille et al. (2021); Maddox et al. (2021).*

Now we provide an invariance property of prior and posterior distributions w.r.t. function-preserving scale transformations as follows: The main intuition behind this proposition is that **Jacobian w.r.t. connectivity is invariant to the function-preserving scaling transformation**, i.e., $\mathbf{J}_\theta(x, \mathcal{T}(\theta^*))\mathrm{diag}(\mathcal{T}(\theta^*)) = \mathbf{J}_\theta(x, \theta^*)\mathrm{diag}(\theta^*)$.

Proposition 2.2 (Scale-invariance of PAC-Bayes prior and posterior)**.** *Let $\mathcal{T} : \mathbb{R}^P \to \mathbb{R}^P$ is a invertible diagonal linear transformation such that $f(x, \mathcal{T}(\psi)) = f(x, \psi)$, $\forall x \in \mathbb{R}^D, \forall \psi \in \mathbb{R}^P$. Then, both PAC-Bayes prior and posterior are invariant under $\mathcal{T}$:*

$$\mathbb{P}_{\mathcal{T}(\theta^*)}(c) \overset{d}{=} \mathbb{P}_{\theta^*}(c), \quad \mathbb{Q}_{\mathcal{T}(\theta^*)}(c) \overset{d}{=} \mathbb{Q}_{\theta^*}(c).$$

Furthermore, generalization and empirical errors are also invariant to $\mathcal{T}$.

2.3 Resulting PAC-Bayes bound

Now we plug in prior and posterior into the modified PAC-Bayes generalization error bound in equation 2. Accordingly, we obtain a novel generalization error bound, named **PAC-Bayes-CTK**, which is guaranteed to be invariant to scale transformations (hence without the contradiction of FM hypothesis mentioned in Sec. 1).

Theorem 2.3 (PAC-Bayes-CTK and its invariance)**.** *Let us assume pre-trained parameter θ^* with data $\mathcal{S}_{\mathbb{P}}$. By applying $\mathbb{P}_{\theta^*}, \mathbb{Q}_{\theta^*}$ to data-dependent PAC-Bayes bound (equation 2), we get*

$$\mathrm{err}_{\mathcal{D}}(\mathbb{Q}_{\theta^*}) \le \mathrm{err}_{\mathcal{S}_{\mathbb{Q}}}(\mathbb{Q}_{\theta^*}) + \sqrt{\overbrace{\underbrace{\frac{\mu_{\mathbb{Q}}^\top\mu_{\mathbb{Q}}}{4\alpha^2 N_{\mathbb{Q}}}}_{\text{(average) perturbation}} + \underbrace{\sum_{i=1}^{P}\frac{h(\beta_i)}{4N_{\mathbb{Q}}}}_{\text{sharpness}}}^{\text{KL divergence}} + \frac{\log(2\sqrt{N_{\mathbb{Q}}}/\delta)}{2N_{\mathbb{Q}}}} \tag{8}$$

where $\{\beta_i\}_{i=1}^P$ are eigenvalues of $(\mathbf{I}_P + \frac{\alpha^2}{\sigma^2}\mathbf{J}_c^\top\mathbf{J}_c)^{-1}$ and $h(x) := x - \log(x) - 1$. ***This upper bound is invariant to*** *$\mathcal{T}$ for the function-preserving scale transformation in Proposition 2.2.*

Note that recent works on solving FM contradiction only focused on the scale-invariance of sharpness metric: Indeed, their generalization bounds are not invariant to scale transformations due to the

scale-dependent terms (equation (34) in Tsuzuku et al. (2020) and equation (6) in Kwon et al. (2021)). On the other hand, generalization bound in Petzka et al. (2021) (Theorem 11 in their paper) only holds for single-layer NNs, whereas ours has no restrictions for network structure. Therefore, to the best of our knowledge, our PAC-Bayes bound is the first scale-invariant PAC-Bayes bound. To highlight our theoretical implications, we show the representative cases of $\mathcal{T}$ in Proposition 2.2 in Appendix D (e.g., weight decay for network with BN), where the generalization bounds of the other studies are variant, but ours is invariant, resolving the FM contradiction on bound level.

The following corollary explains why we name PAC-Bayes bound in Theorem 2.3 PAC-Bayes-CTK.

Corollary 2.4 (Relation between CTK and PAC-Bayes-CTK)**.** *Let us define empirical **Connectivity Tangent Kernel** (CTK) of $\mathcal{S}$ as $\mathbf{C}_{\mathcal{X}}^{\theta^*} := \mathbf{J}_c\mathbf{J}_c^\top = \mathbf{J}_\theta \mathrm{diag}(\theta^*)^2\mathbf{J}_\theta^\top \in \mathbb{R}^{NK\times NK}$ by removing below term? Note that empirical CTK has $T(\leq NK)$ non-zero eigenvalues of $\{\lambda_i\}_{i=1}^T$, then following holds for $\{\beta\}_{i=1}^P$ in Theorem 2.3: (i) $\beta_i = \sigma^2/(\sigma^2+\alpha^2\lambda_i) < 1$ for $i = 1,\dots,T$ and (ii) $\beta_i = 1$ for $i = T+1,\dots,P$. Since $h(1) = 0$, this means $P-T$ terms of summation in sharpness part of PAC-Bayes-CTK vanishes to 0. Furthermore, this sharpness term of PAC-Bayes-CTK is a monotonically increasing function for each eigenvalue of empirical CTK.*

Note that Corollary 2.4 clarifies why $\sum_{i=1}^P h(\beta_i)/4N_{\mathbb{Q}}$ in Theorem 2.3 is called the sharpness term of PAC-Bayes-CTK. As eigenvalues of CTK measures the sensitivity of output w.r.t. perturbation on connectivity, a sharp pre-trained parameter would have large CTK eigenvalues, so increasing the sharpness term and the generalization gap by according to Corollary 2.4.

Finally, Proposition 2.5 shows that empirical CTK is also scale-invariant.

Proposition 2.5 (Scale-invariance of empirical CTK)**.** *Let $\mathcal{T}:\mathbb{R}^P \to \mathbb{R}^P$ be an function-preserving scale transformation in Proposition 2.2. Then empirical CTK at parameter ψ is invariant under $\mathcal{T}$:*

$$\mathbf{C}_{xy}^{\mathcal{T}(\psi)} := \mathbf{J}_\theta(x,\mathcal{T}(\psi))\mathrm{diag}(\mathcal{T}(\psi)^2)\mathbf{J}_\theta(y,\mathcal{T}(\psi))^\top = \mathbf{C}_{xy}^{\psi}\,,\ \forall x,y\in\mathbb{R}^D, \forall\psi\in\mathbb{R}^P. \tag{9}$$

Remark 2.6 (Connections to empirical NTK)**.** *The empirical CTK $\mathbf{C}_{xy}^{\psi}$ resembles the existing empirical Neural Tangent Kernel (NTK) at parameter ψ (Jacot et al., 2018): $\Theta_{xy}^{\psi} := \mathbf{J}_\theta(x,\psi)\mathbf{J}_\theta(y,\psi)^\top \in \mathbb{R}^{k\times k}$. Note that the deterministic NTK in Jacot et al. (2018) is the infinite-width limiting kernel at initialized parameters, while empirical NTK can be defined on any parameter of a finite-width NN. The main difference between empirical CTK and the existing empirical NTK is in the definition of Jacobian. In CTK, Jacobian is computed w.r.t. connectivity c while the empirical NTK uses Jacobian w.r.t. parameters θ. Therefore, another PAC-Bayes bound can be derived from the linearization of $f_{\theta^*}^{\mathrm{pert}}(x,\delta) := f(x,\theta^*+\delta) \approx f_{\theta^*}^{\mathrm{lin}}(x,\delta)$ where $f_{\theta^*}^{\mathrm{lin}}(x,\delta) := f(x,\theta^*) + \mathbf{J}_\theta(x,\theta^*)\delta$. As this bound is related to the eigenvalues of $\Theta_{\mathcal{X}}^{\theta^*}$, we call this bound **PAC-Bayes-NTK** and provide derivations in Appendix A. Note PAC-Bayes-NTK is not scale-invariant as $\Theta_{xy}^{\mathcal{T}(\psi)} \neq \Theta_{xy}^{\psi}$ in general.*

2.4 Computing approximate bound in real world problems

To verify that PAC-Bayes bound in Theorem 2.3 is non-vacuous, we compute it for real-world problems. We use CIFAR-10 and 100 datasets (Krizhevsky, 2009), where the 50K training instances are randomly partitioned into $\mathcal{S}_{\mathbb{P}}$ of cardinality 45K and $\mathcal{S}_{\mathbb{Q}}$ of cardinality 5K. We pre-train ResNet-18 (He et al., 2016) with a mini-batch size of 1K on $\mathcal{S}_{\mathbb{P}}$ with SGD of initial learning rate 0.4 and momentum 0.9. We use cosine annealing for learning rate scheduling (Loshchilov & Hutter, 2016) with a warmup for the initial 10% training step. We fix $\delta = 0.1$ and select α, σ based on the negative log-likelihood of $\mathcal{S}_{\mathbb{Q}}$.

To compute the equation 8, one need (i) $\mu_{\mathbb{Q}}$-based perturbation term, (ii) $\mathbf{C}_{\mathcal{X}}^{\theta^*}$-based sharpness term, and (iii) samples from PAC-Bayes posterior $\mathbb{Q}_{\theta^*}$. $\mu_{\mathbb{Q}}$ in equation 6 can be obtained by minimizing $\arg\min_{c\in\mathbb{R}^P} L(c) = \frac{1}{2N}\|\mathcal{Y} - f(\mathcal{X},\theta^*) - \mathbf{J}_c c\|^2 + \frac{\sigma^2}{2\alpha^2 N}c^\top c$ by first-order optimality condition. Note that this problem is a convex optimization problem w.r.t. c, since c is the parameter of the linear regression problem. We use Adam optimizer (Kingma & Ba, 2014) with fixed learning rate 1e-4 to solve this. For sharpness term, we apply the Lanczos algorithm to approximate the eigenspectrum of $\mathbf{C}_{\mathcal{X}}^{\theta^*}$ following Ghorbani et al. (2019). We use 100 Lanczos iterations based on the their setting. Lastly, we estimate empirical error and test error with 8 samples of CL/LL implementation of Randomize-Then-Optimize (RTO) framework (Bardsley et al., 2014; Matthews et al., 2017). We refer to Appendix E for the RTO implementation of CL/LL.

Table 1: Results for experiments on PAC-Bayes-CTK estimation.

	CIFAR-10				CIFAR-100			
Parameter scale	0.5	1.0	2.0	4.0	0.5	1.0	2.0	4.0
$\mathrm{tr}(\mathbf{C}_{\mathcal{X}}^{\theta^*})$	14515.0039	14517.7793	14517.3506	14518.4746	12872.6895	12874.4395	12873.9512	12875.541
Bias	13.9791	13.4685	13.3559	13.3122	25.3686	24.8064	24.9102	24.7557
Sharpness	24.6874	24.6938	24.6926	24.6941	26.0857	26.0894	26.0874	26.0916
KL	19.3332	19.0812	19.0243	19.0032	25.7271	25.4479	25.4988	25.4236
Test err.	0.0468 ± 0.0014	0.0463 ± 0.0013	0.0462 ± 0.0012	0.0460 ± 0.0013	0.2257 ± 0.0020	0.2252 ± 0.0017	0.2256 ± 0.0015	0.2253 ± 0.0017
PAC-Bayes-CTK	0.0918 ± 0.0013	0.0911 ± 0.0011	0.0909 ± 0.0011	0.0907 ± 0.0009	0.2874 ± 0.0034	0.2862 ± 0.0031	0.2860 ± 0.0032	0.2862 ± 0.0032

Table 1 provides the bound and related term of the resulting model. First of all, we found that our estimated PAC-Bayes-CTK is non-vacuous (Zhou et al., 2018): estimated bound is better than guessing at random. Note that the non-vacuous bound is not trivial in PAC-Bayes analysis: only a few PAC-Bayes literatures (Dziugaite & Roy, 2017; Zhou et al., 2018; Perez-Ortiz et al., 2021) verified the non-vacuous property of their bound, and other PAC-Bayes literatures (Foret et al., 2020; Tsuzuku et al., 2020) do not. To check the invariance property of our bound, we scale the scale-invariant parameters in ResNet-18 (i.e., parameters preceding BN layers) for fixed constants $\{0.5, 1.0, 2.0, 4.0\}$. Note that this scaling does not change the function represented by NN due to BN layers, and the error bound should be preserved. Table 1 shows that our bound and related terms are invariant to these transformations. On the other hand, PAC-Bayes-NTK is very sensitive to parameter scale, as shown in Table 7 in Appendix J.

2.5 Connectivity Sharpness and its efficient computation

Now, we focus on the fact that the trace of CTK is also invariant to the parameter scale by Proposition 2.5. Unlike PAC-Bayes-CTK/NTK, a trace of CTK/NTK does not require onerous hyper-parameter selection of δ, α, σ. Therefore, we simply define $\mathbf{CS}(\theta^*) := \mathrm{tr}(\mathbf{C}_{\mathcal{X}}^{\theta^*})$ as a practical sharpness measure at θ^*, named **Connectivity Sharpness** (CS) to detour the complex computation of PAC-Bayes-CTK. This metric can be easily applied to find NNs with better generalization, similar to other sharpness metrics (e.g., trace of Hessian), as shown in Jiang et al. (2020). We evaluate the detecting performance of CS in Sec. 4.1. The following corollary shows how CS can explain the generalization performance of NNs, conceptually.

Corollary 2.7 (Connectivity sharpness, Informal)**.** *Let us assume CTK and KL divergence term of PAC-Bayes-CTK as defined in Theorem 2.3. Then, if CS vanishes to zero or infinity, the KL divergence term of PAC-Bayes-CTK also does so.*

As the trace of a matrix can be efficiently estimated by Hutchinson's method (Hutchinson, 1989), one can compute the CS *without explicitly computing the entire CTK*. We refer to Appendix F for detailed procedures of computing CS. As CS is invariant to function-preserving scale transformations by Theorem 2.5, it also does not contradict the FM hypothesis.

3 Bayesian NNs with scale-invariance

This section provides the practical implications of the posterior distribution used in PAC-Bayes analysis. We interpret the PAC-Bayes posterior $\mathbb{Q}_{\theta^*}$ in equation 5 as a modified result of LA (MacKay, 1992). Then, we show this modification improves existing LA in the presence of weight decay regularization. Finally, we explain how one can efficiently construct a Bayesian NN from equation 5.

3.1 Pitfalls of weight decay with BN in Laplace Approximation

One can view parameter space version of $\mathbb{Q}_{\theta^*}$ as a modified version of $p_{\mathrm{LA}}(\psi|\mathcal{S})$ in equation 1 by (i) replacing isotropic damping term to the parameter scale-dependent damping term ($\mathrm{diag}(\theta^*)^{-2}$) and (ii) adding perturbation $\theta^* \odot \mu_{\mathbb{Q}}$ to the mean of Gaussian distribution. In this section, we focus on the effect of replacing the damping term of LA in the presence of weight decay of batch normalized NNs. We refer to Antoran et al. (2021; 2022) for the discussion on the effect of adding perturbation to the LA with linearized NNs.

The main difference between covariance term of LA equation 1 and equation 7 is the definition of Jacobian (i.e. parameter or connectivity) similar to the difference between empirical CTK and

Table 2: Correlation analysis of sharpness measures with generalization gap. We refer Sec. 4.1 for the details of sharpness measures (row) and correlation metrics for sharpness-generalization relationship (column).

	$\mathrm{tr}(\mathbf{H})$	$\mathrm{tr}(\mathbf{F})$	$\mathrm{tr}(\mathbf{\Theta}^{\theta^*})$	SO	PO	SM	PM	AS	FR	CS
τ (rank corr.)	0.706	0.679	0.703	0.490	0.436	0.473	0.636	0.755	0.649	**0.837**
network depth	0.764	0.652	**0.978**	-0.358	-0.719	0.774	0.545	0.756	0.771	**0.978**
network width	0.687	0.922	0.330	-0.533	-0.575	0.495	0.564	0.827	0.921	**0.978**
mini-batch size	0.976	0.810	**0.988**	0.859	0.893	0.909	0.750	0.829	0.685	0.905
learning rate	0.966	0.713	**1.000**	0.829	0.874	0.057	0.621	0.794	0.565	0.897
weight decay	-0.031	-0.103	0.402	0.647	0.711	0.168	0.211	0.710	0.373	**0.742**
Ψ (avg.)	0.672	0.599	0.739	0.289	0.237	0.481	0.538	0.783	0.663	**0.900**
$\mathcal{K}$ (cond. MI)	0.320	0.243	0.352	0.039	0.041	0.049	0.376	0.483	0.288	**0.539**

NTK in remark 2.6. Therefore, we name $p_{\mathrm{CL}}(\psi|\mathcal{S}) \sim \mathcal{N}(\psi|\theta^*, \left(\mathrm{diag}(\theta^*)^{-2}/\alpha^2 + \mathbf{J}_\theta^\top \mathbf{J}_\theta/\sigma^2\right)^{-1})$ as **Connectivity Laplace (CL)** approximated posterior.

To compare CL posterior and existing LA, we explain how weight decay regularization with BN produces unexpected side effects in existing LA. This side effect can be quantified if we consider linearized NN for LA, called Linearized Laplace (LL; Foong et al. (2019)). Note that LL is well known to be better calibrated than non-linearized LA for estimating 'in-between' uncertainty. By assuming $\sigma^2 \ll \alpha^2$, the predictive distribution of linearized NN for equation 1 and CL are

$$f^{\mathrm{lin}}_{\theta^*}(x,\psi)|p_{\mathrm{LA}}(\psi|\mathcal{S}) \sim \mathcal{N}(f(x,\theta^*), \alpha^2\Theta^{\theta^*}_{xx} - \alpha^2\Theta^{\theta^*}_{x\mathcal{X}}\Theta^{\theta^*}_{\mathcal{X}}{}^{-1}\Theta^{\theta^*}_{\mathcal{X}x}) \tag{10}$$

$$f^{\mathrm{lin}}_{\theta^*}(x,\psi)|p_{\mathrm{CL}}(\psi|\mathcal{S}) \sim \mathcal{N}(f(x,\theta^*), \alpha^2\mathbf{C}^{\theta^*}_{xx} - \alpha^2\mathbf{C}^{\theta^*}_{x\mathcal{X}}\mathbf{C}^{\theta^*}_{\mathcal{X}}{}^{-1}\mathbf{C}^{\theta^*}_{\mathcal{X}x}). \tag{11}$$

for any input $x \in \mathbb{R}^d$ where $\mathcal{X}$ in subscript of CTK/NTK means concatenation. We refer to Appendix G for the detailed derivations. The following proposition shows how weight decay regularization on scale-invariant parameters introduced by BN can amplify the predictive uncertainty of equation 10. Note that the primal regularizing effect of weight decay originates through regularization on scale-invariant parameters (Van Laarhoven, 2017; Zhang et al., 2019).

Proposition 3.1 (Uncertainty amplifying effect for LL)**.** *Let us assume that $\mathcal{W}_\gamma : \mathbb{R}^P \to \mathbb{R}^P$ is a weight decay regularization on scale-invariant parameters (e.g., parameters preceding BN layers) by multiplying $\gamma < 1$ and all the non-scale-invariant parameters are fixed. Then, predictive uncertainty of LL is amplified by $1/\gamma^2 > 1$ while predictive uncertainty of CTK is preserved:*

$$\mathit{Var}_{\psi\sim p_{\mathrm{LA}}(\psi|\mathcal{S})}(f^{\mathrm{lin}}_{\mathcal{W}_\gamma(\theta^*)}(x,\psi)) = \mathit{Var}_{\psi\sim p_{\mathrm{LA}}(\psi|\mathcal{S})}(f^{\mathrm{lin}}_{\theta^*}(x,\psi))/\gamma^2$$
$$\mathit{Var}_{\psi\sim p_{\mathrm{CL}}(\psi|\mathcal{S})}(f^{\mathrm{lin}}_{\mathcal{W}_\gamma(\theta^*)}(x,\psi)) = \mathit{Var}_{\psi\sim p_{\mathrm{CL}}(\psi|\mathcal{S})}(f^{\mathrm{lin}}_{\theta^*}(x,\psi))$$

where $\mathit{Var}(\cdot)$ is variance of random variable.

Recently, Antoran et al. (2021; 2022) observed similar pitfalls in Proposition 3.1. However, their solution requires a more complicated hyper-parameter search: *independent prior selection for each normalized parameter group*. On the other hand, CL does not increase the hyper-parameter to be optimized compared to LL. We believe this difference will make CL more attractive to practitioners.

4 Experiments

Here we describe experiments that demonstrate (i) the effectiveness of Connectivity Sharpness (CS) as a generalization measurement metric and (ii) the usefulness of Connectivity Laplace (CL) as an efficient general-purpose Bayesian NN: CL resolves the contradiction of FM hypothesis and shows stable calibration performance to the selection of prior scale.

4.1 Connectivity Sharpness as a generalization measurement metric

To verify that the CS actually has a better correlation with generalization performance compared to existing sharpness measures, we evaluate the three correlation metrics on the CIFAR-10 dataset: (a) Kendall's rank-correlation coefficient (τ) (Kendall, 1938) (b) granulated Kendall's coefficient and their average (Ψ) (Jiang et al., 2020) (c) conditional independence test ($\mathcal{K}$) (Jiang et al., 2020). For all correlation metrics, a larger value means a stronger correlation between sharpness and generalization.

Table 3: Test negative log-likelihood on two UCI variants (Hernández-Lobato & Adams, 2015; Foong et al., 2019)

	Original (Hernández-Lobato & Adams, 2015)				GAP variants (Foong et al., 2019)			
	Deep Ensemble	MCDO	LL	CL	Deep Ensemble	MCDO	LL	CL
boston_housing	2.90 ± 0.03	2.63 ± 0.01	**2.85 ± 0.01**	2.88 ± 0.02	2.71 ± 0.01	2.68 ± 0.01	**2.74 ± 0.01**	2.75 ± 0.01
concrete_strength	3.06 ± 0.01	3.20 ± 0.00	3.22 ± 0.01	**3.11 ± 0.02**	4.03 ± 0.07	3.42 ± 0.00	**3.47 ± 0.01**	4.03 ± 0.02
energy_efficiency	0.74 ± 0.01	1.92 ± 0.01	2.12 ± 0.01	**0.83 ± 0.01**	0.77 ± 0.01	1.78 ± 0.01	2.02 ± 0.01	0.90 ± 0.02
kin8nm	-1.07 ± 0.00	-0.80 ± 0.01	-0.90 ± 0.00	**-1.07 ± 0.00**	-0.94 ± 0.00	-0.71 ± 0.00	-0.87 ± 0.00	**-0.93 ± 0.00**
naval_propulsion	-4.83 ± 0.00	-3.85 ± 0.00	-4.57 ± 0.00	**-4.76 ± 0.00**	-2.22 ± 0.33	-3.36 ± 0.01	-3.66 ± 0.11	**-3.80 ± 0.07**
power_plant	2.81 ± 0.00	2.91 ± 0.00	2.91 ± 0.00	**2.81 ± 0.00**	2.91 ± 0.00	2.97 ± 0.00	2.98 ± 0.00	**2.91 ± 0.00**
protein_structure	2.89 ± 0.00	2.96 ± 0.00	2.91 ± 0.00	**2.89 ± 0.00**	3.11 ± 0.00	3.07 ± 0.00	**3.07 ± 0.00**	3.13 ± 0.00
wine	1.21 ± 0.00	0.96 ± 0.01	**1.24 ± 0.01**	1.27 ± 0.01	1.48 ± 0.01	1.03 ± 0.00	1.45 ± 0.01	**1.43 ± 0.00**
yacht_hydrodynamics	1.26 ± 0.04	2.17 ± 0.06	**1.20 ± 0.04**	1.25 ± 0.04	1.71 ± 0.03	3.06 ± 0.02	1.78 ± 0.02	**1.74 ± 0.01**

We compare CS to following baseline sharpness measures: trace of Hessian ($\mathrm{tr}(\mathbf{H})$; Keskar et al. (2017)), trace of empirical Fisher ($\mathrm{tr}(\mathbf{F})$; Jastrzebski et al. (2021)), trace of empirical NTK at θ^* ($\mathrm{tr}(\mathbf{\Theta}^{\theta^*})$), Fisher-Rao (FR ;Liang et al. (2019)) metric, Adaptive Sharpness (AS; Kwon et al. (2021)), and four PAC-Bayes bound based measures: Sharpness-Orig. (SO), Pacbayes-Orig. (PO), Sharpness-Mag. (SM), and Pacbayes-Mag. (PM), which are eq. (52), (49), (62), (61) in Jiang et al. (2020). For computing granulated Kendall's correlation, we use 5 hyper-parameters (network depth, network width, learning rate, weight decay, and mini-batch size) and 3 options for each (thus we train models with $3^5 = 243$ different training configurations). We vary the depth and width of NN based on VGG-13 (Simonyan & Zisserman, 2015). We refer to Appendix H for experimental details.

In Table 2, CS shows the best results for τ, Ψ, and $\mathcal{K}$ compared to all other sharpness measures. Also, granulated Kendall of CS is higher than other sharpness measures for 3 out of 5 hyper-parameters and competitive to other sharpness measures with the leftover hyper-parameters. The main difference of CS with other sharpness measures is in the correlation with *network width* and *weight decay*: For *network width*, we found that sharpness measures except CS, $\mathrm{tr}(\mathbf{F})$, AS and FR fail to capture strong correlation. While SO/PO can capture the correlation with *weight decay*, we believe it is due to the weight norm term of SO/PO. However, this term would interfere in capturing the sharpness-generalization correlation related to the number of parameters (i.e., width/depth), while CS/AS does not suffer from such a problem. Also, it is notable that FR fails to capture this correlation despite its invariant property.

4.2 Connectivity Laplace as an efficient general-purpose Bayesian NN

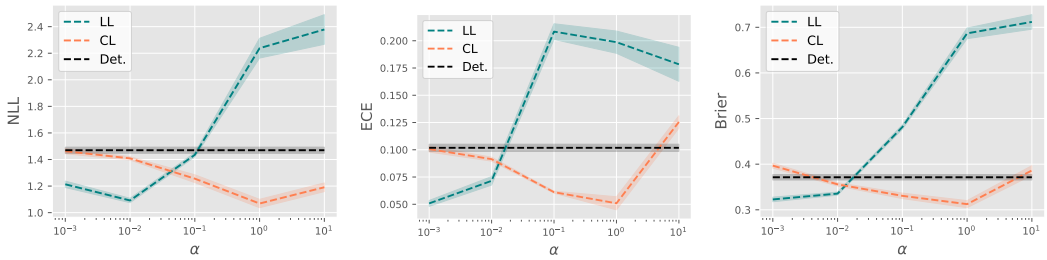
To assess the effectiveness of CL as a general-purpose Bayesian NN, we consider uncertainty calibration on UCI dataset and CIFAR-10/100.

UCI regression datasets We implement full-curvature versions of LL and CL and evaluate these to the 9 UCI regression datasets (Hernández-Lobato & Adams, 2015) and its GAP-variants (Foong et al., 2019) to compare calibration performance on *in-between* uncertainty. We train MLP with a single hidden layer. We fix $\sigma = 1$ and choose α from $\{0.01, 0.1, 1, 10, 100\}$ using log-likelihood of validation dataset. We use 8 random seeds to compute the average and standard error of the test negative log-likelihoods. The following two tables show test NLL for LL/CL and 2 baselines (Deep Ensemble (Lakshminarayanan et al., 2017) and Monte-Carlo DropOut (MCDO; Gal & Ghahramani (2016))). Eight ensemble members are used in Deep Ensemble, and 32 MC samples are used in LL, CL, and MCDO. Table 3 show that CL performs better than LL for 6 out of 9 datasets. Although LL shows better calibration results for 3 datasets in both settings, we would like to point out that the performance gaps between LL and CL were not severe as in the other 6 datasets, where CL performs better.

Image Classification We evaluate the uncertainty calibration performance of CL on CIFAR-10/100. As baseline methods, we consider Deterministic network, Monte-Carlo Dropout (MCDO; (Gal & Ghahramani, 2016)), Monte-Carlo Batch Normalization (MCBN; (Teye et al., 2018)), and Deep Ensemble (Lakshminarayanan et al., 2017), Batch Ensemble (Wen et al., 2020), LL (Khan et al., 2019; Daxberger et al., 2021a). We use Randomize-Then-Optimize (RTO) implementation of LL/CL in Appendix E. We measure Expected Calibration Error (ECE; Guo et al. (2017)), negative log-likelihood (NLL), and Brier score (Brier.) for ensemble predictions. We also measure the area under receiver operating curve (AUC) for OOD detection, where we set the SVHN (Netzer et al., 2011) dataset as an OOD dataset. For more details on the experimental setting, please refer to Appendix I.

Table 4: Uncertainty calibration results on CIFAR-100 (Krizhevsky, 2009) for ResNet-18 (He et al., 2016)

	CIFAR-100			
	NLL (↓)	ECE (↓)	Brier. (↓)	AUC (↑)
Deterministic	1.5370 ± 0.0117	0.1115 ± 0.0017	0.3889 ± 0.0031	-
MCDO	1.4264 ± 0.0110	0.0651 ± 0.0008	0.3925 ± 0.0020	0.6907 ± 0.0121
MCBN	1.4689 ± 0.0106	0.0998 ± 0.0016	0.3750 ± 0.0028	0.7982 ± 0.0210
Batch Ensemble	1.4029 ± 0.0031	0.0842 ± 0.0005	0.3582 ± 0.0010	0.7887 ± 0.0115
Deep Ensemble	1.0110	0.0507	0.2740	0.7802
Linearized Laplace	1.1673 ± 0.0093	0.0632 ± 0.0010	0.3597 ± 0.0020	0.8066 ± 0.0120
Connectivity Laplace (Ours)	1.1307 ± 0.0042	0.0524 ± 0.0009	0.3319 ± 0.0005	0.8423 ± 0.0204



(a) NLL (b) ECE (c) Brier Score

Figure 1: Sensitivity to α. Expected calibration error (ECE), Negative Log-likelihood (NLL), and Brier score results on corrupted CIFAR-100 for ResNet-18. Showing the mean (line) and standard deviation (shaded area) across four different seeds.

Table 4 shows uncertainty calibration results on CIFAR-100. We refer to Appendix I for results on other settings, including CIFAR-10 and VGGNet (Simonyan & Zisserman, 2015). Our CL shows better results than baselines for all uncertainty calibration metrics (NLL, ECE, Brier., and AUC) except Deep Ensemble. This means scale-invariance of CTK improves Bayesian inference, which is consistent with the results in toy examples. Although the Deep Ensemble presents the best results in 3 out of 4 metrics, it requires full training from initialization for each ensemble member, while LL/CL requires only a post-hoc training upon the pre-trained NN for each member. Particularly noteworthy is that CL presents competitive results with Deep Ensemble, even with much smaller computations.

Robustness to the selection of prior scale Figure 1 shows the uncertainty calibration (i.e. NLL, ECE, Brier) results over various α values for LL, CL, and Deterministic (Det.) baseline. As mentioned in previous works (Ritter et al., 2018; Kristiadi et al., 2020), the uncertainty calibration results of LL is extremely sensitive to the selection of α. Especially, LL shows severe under-fitting for large α (i.e. small damping) regime. On the other hand, CL shows stable performance in the various ranges of α.

5 Conclusion

This study introduced novel PAC-Bayes prior and posterior distributions to extend the robustness of generalization bound w.r.t. parameter transformation by *decomposing* the scale and connectivity of parameters. The resulting generalization bound is guaranteed to be invariant of any function-preserving scale transformations. This result successfully solved the problem that the contradiction of the FM hypothesis caused by the general scale transformation could not be solved in the existing generalization error bound, thereby allowing the theory to be much closer to reality. As a result of the theoretical enhancement, our posterior distribution for PAC-analysis can also be interpreted as an improved Laplace Approximation without pathological failures in weight decay regularization. Therefore, we expect this fact contributes to reducing the theory-practice gap in understanding the generalization effect of NN, leading to follow-up studies that interpret this effect more clearly.

REFERENCES

Alessandro Achille, Aditya Golatkar, Avinash Ravichandran, Marzia Polito, and Stefano Soatto. Lqf: Linear quadratic fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15729–15739, 2021.

Javier Antoran, James Urquhart Allingham, David Janz, Erik Daxberger, Eric Nalisnick, and José Miguel Hernández-Lobato. Linearised laplace inference in networks with normalisation layers and the neural g-prior. In *Fourth Symposium on Advances in Approximate Bayesian Inference*, 2021.

Javier Antoran, David Janz, James U Allingham, Erik Daxberger, Riccardo Rb Barbano, Eric Nalisnick, and José Miguel Hernández-Lobato. Adapting the linearised laplace model evidence for modern deep learning. In *International Conference on Machine Learning*, pp. 796–821. PMLR, 2022.

Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 254–263. PMLR, 10–15 Jul 2018.

Johnathan M Bardsley, Antti Solonen, Heikki Haario, and Marko Laine. Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems. *SIAM Journal on Scientific Computing*, 36(4):A1895–A1910, 2014.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

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. *github*, 2018.

Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux - effortless bayesian deep learning. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021a.

Erik Daxberger, Eric Nalisnick, James U Allingham, Javier Antoran, and Jose Miguel Hernandez-Lobato. Bayesian deep learning via subnetwork inference. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 2510–2521. PMLR, 18–24 Jul 2021b.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1019–1028. PMLR, 06–11 Aug 2017.

Gintare Karolina Dziugaite and Daniel M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In *Proceedings of the 33rd Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, 2017.

Andrew YK Foong, Yingzhen Li, José Miguel Hernández-Lobato, and Richard E Turner. 'in-between'uncertainty in bayesian neural networks. *arXiv preprint arXiv:1906.11537*, 2019.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

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.

Behrooz Ghorbani, Shankar Krishnan, and Ying Xiao. An investigation into neural net optimization via hessian eigenvalue density. In *International Conference on Machine Learning*, pp. 2232–2241. PMLR, 2019.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330. PMLR, 06–11 Aug 2017.

Bobby He, Balaji Lakshminarayanan, and Yee Whye Teh. Bayesian deep ensembles via the neural tangent kernel. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1010–1022. Curran Associates, Inc., 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

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.

Sepp Hochreiter and Jürgen Schmidhuber. Simplifying neural nets by discovering flat minima. In G. Tesauro, D. Touretzky, and T. Leen (eds.), *Advances in Neural Information Processing Systems*, volume 7. MIT Press, 1995.

Like Hui and Mikhail Belkin. Evaluation of neural architectures trained with square loss vs cross-entropy in classification tasks. In *International Conference on Learning Representations*, 2021.

Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 18(3):1059–1076, 1989.

Alexander Immer, Maciej Korzepa, and Matthias Bauer. Improving predictions of bayesian neural nets via local linearization. In *AISTATS*, pp. 703–711, 2021.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 448–456, Lille, France, 07–09 Jul 2015. PMLR.

Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Stanislaw Jastrzebski, Devansh Arpit, Oliver Astrand, Giancarlo B Kerg, Huan Wang, Caiming Xiong, Richard Socher, Kyunghyun Cho, and Krzysztof J Geras. Catastrophic fisher explosion: Early phase fisher matrix impacts generalization. In *International Conference on Machine Learning*, pp. 4772–4784. PMLR, 2021.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*, 2020.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

Nitish Shirish Keskar, Jorge Nocedal, Ping Tak Peter Tang, Dheevatsa Mudigere, and Mikhail Smelyanskiy. On large-batch training for deep learning: Generalization gap and sharp minima. In *5th International Conference on Learning Representations, ICLR 2017*, 2017.

Mohammad Emtiyaz E Khan, Alexander Immer, Ehsan Abedi, and Maciej Korzepa. Approximate inference turns deep networks into gaussian processes. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Agustinus Kristiadi, Matthias Hein, and Philipp Hennig. Being bayesian, even just a bit, fixes overconfidence in relu networks. In *International conference on machine learning*, pp. 5436–5446. PMLR, 2020.

A Krizhevsky. Learning multiple layers of features from tiny images. *Master's thesis, University of Toronto*, 2009.

Jungmin Kwon, Jeongseop Kim, Hyunseo Park, and In Kwon Choi. Asam: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. *arXiv preprint arXiv:2102.11600*, 2021.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Jaehoon Lee, Lechao Xiao, Samuel Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019a.

Jaehoon Lee, Samuel Schoenholz, Jeffrey Pennington, Ben Adlam, Lechao Xiao, Roman Novak, and Jascha Sohl-Dickstein. Finite versus infinite neural networks: an empirical study. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 15156–15172. Curran Associates, Inc., 2020.

Namhoon Lee, Thalaiyasingam Ajanthan, Stephen Gould, and Philip HS Torr. A signal propagation perspective for pruning neural networks at initialization. *arXiv preprint arXiv:1906.06307*, 2019b.

Namhoon Lee, Thalaiyasingam Ajanthan, and Philip Torr. SNIP: SINGLE-SHOT NETWORK PRUNING BASED ON CONNECTION SENSITIVITY. In *International Conference on Learning Representations*, 2019c.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Tengyuan Liang, Tomaso Poggio, Alexander Rakhlin, and James Stokes. Fisher-rao metric, geometry, and complexity of neural networks. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 888–896. PMLR, 2019.

Ekaterina Lobacheva, Maxim Kodryan, Nadezhda Chirkova, Andrey Malinin, and Dmitry P Vetrov. On the periodic behavior of neural network training with batch normalization and weight decay. *Advances in Neural Information Processing Systems*, 34:21545–21556, 2021.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

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

Wesley Maddox, Shuai Tang, Pablo Moreno, Andrew Gordon Wilson, and Andreas Damianou. Fast adaptation with linearized neural networks. In *International Conference on Artificial Intelligence and Statistics*, pp. 2737–2745. PMLR, 2021.

Alexander G de G Matthews, Jiri Hron, Richard E Turner, and Zoubin Ghahramani. Sample-then-optimize posterior sampling for bayesian linear models. In *NIPS Workshop on Advances in Approximate Bayesian Inference*, 2017.

David A McAllester. Pac-bayesian model averaging. In *Proceedings of the twelfth annual conference on Computational learning theory*, pp. 164–170, 1999.

Raphael A Meyer, Cameron Musco, Christopher Musco, and David P Woodruff. Hutch++: Optimal stochastic trace estimation. In *Symposium on Simplicity in Algorithms (SOSA)*, pp. 142–155. SIAM, 2021.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011.

Behnam Neyshabur, Russ R Salakhutdinov, and Nati Srebro. Path-sgd: Path-normalized optimization in deep neural networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015a.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. In *ICLR (Workshop)*, 2015b.

Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nathan Srebro. Exploring generalization in deep learning. *arXiv preprint arXiv:1706.08947*, 2017.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Maria Perez-Ortiz, Omar Risvaplata, John Shawe-Taylor, and Csaba Szepesvári. Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22(227):1–40, 2021.

Henning Petzka, Michael Kamp, Linara Adilova, Cristian Sminchisescu, and Mario Boley. Relative flatness and generalization. *Advances in Neural Information Processing Systems*, 34, 2021.

Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Hippolyt Ritter, Aleksandar Botev, and David Barber. A scalable laplace approximation for neural networks. In *International Conference on Learning Representations*, 2018.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Mattias Teye, Hossein Azizpour, and Kevin Smith. Bayesian uncertainty estimation for batch normalized deep networks. In *International Conference on Machine Learning*, pp. 4907–4916. PMLR, 2018.

Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Normalized flat minima: Exploring scale invariant definition of flat minima for neural networks using PAC-Bayesian analysis. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9636–9647. PMLR, 13–18 Jul 2020.

Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9690–9700. PMLR, 13–18 Jul 2020.

Twan Van Laarhoven. L2 regularization versus batch and weight normalization. *arXiv preprint arXiv:1706.05350*, 2017.

Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. *arXiv preprint arXiv:2002.06715*, 2020.

M.A. Woodbury. *Inverting Modified Matrices*. Memorandum Report / Statistical Research Group, Princeton. Statistical Research Group, 1950.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization, 2017.

Guodong Zhang, Chaoqi Wang, Bowen Xu, and Roger Grosse. Three mechanisms of weight decay regularization. In *International Conference on Learning Representations*, 2019.

Wenda Zhou, Victor Veitch, Morgane Austern, Ryan P Adams, and Peter Orbanz. Non-vacuous generalization bounds at the imagenet scale: a pac-bayesian compression approach. *arXiv preprint arXiv:1804.05862*, 2018.

A PROOFS

A.1 PROOF OF PROPOSITION 2.2

Proof. Since the prior $\mathbb{P}_{\theta^*}(c)$ is independent to the parameter scale, $\mathbb{P}_{\theta^*}(c) \overset{d}{=} \mathbb{P}_{\mathcal{T}(\theta^*)}(c)$ is trivial. For Jacobian w.r.t. parameters, we have

$$\mathbf{J}_\theta(x, \mathcal{T}(\psi)) = \frac{\partial}{\partial \mathcal{T}(\psi)} f(x, \mathcal{T}(\psi)) = \frac{\partial}{\partial \mathcal{T}(\psi)} f(x, \psi) = \mathbf{J}_\theta(x, \psi)\mathcal{T}^{-1}$$

Then, the Jacobian of NN w.r.t. connectivity at $\mathcal{T}(\psi)$ holds

$$\begin{aligned} \mathbf{J}_\theta(x, \mathcal{T}(\psi))\mathrm{diag}(\mathcal{T}(\psi)) &= \mathbf{J}_\theta(x, \psi)\mathcal{T}^{-1}\mathcal{T}\mathrm{diag}(\psi) && (12) \\ &= \mathbf{J}_\theta(x, \psi)\mathrm{diag}(\psi) && (13) \end{aligned}$$

where the first equality holds from the above one and the fact that $\mathcal{T}$ is a diagonal linear transformation. Therefore, the covariance of posterior is invariant to $\mathcal{T}$.

$$\begin{aligned} &\left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathrm{diag}(\mathcal{T}(\theta^*))\mathbf{J}_\theta^\top(\mathcal{X}, \mathcal{T}(\theta^*))\mathbf{J}_\theta(\mathcal{X}, \mathcal{T}(\theta^*))\mathrm{diag}(\mathcal{T}(\theta^*))}{\sigma^2}\right)^{-1} \\ &= \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathrm{diag}(\theta^*)\mathbf{J}_\theta^\top(\mathcal{X}, \theta^*)\mathbf{J}_\theta(\mathcal{X}, \theta^*)\mathrm{diag}(\theta^*)}{\sigma^2}\right)^{-1} \\ &= \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathrm{diag}(\theta^*)\mathbf{J}_\theta^\top\mathbf{J}_\theta\mathrm{diag}(\theta^*)}{\sigma^2}\right)^{-1} \end{aligned}$$

Moreover, the mean of posterior is also invariant to $\mathcal{T}$.

$$\begin{aligned} &\frac{\Sigma_{\mathbb{Q}}\mathrm{diag}(\mathcal{T}(\theta^*))\mathbf{J}_\theta^\top(\mathcal{X}, \mathcal{T}(\theta^*))\left(\mathcal{Y} - f(\mathcal{X}, \mathcal{T}(\theta^*))\right)}{\sigma^2} \\ &= \frac{\Sigma_{\mathbb{Q}}\mathrm{diag}(\mathcal{T}(\theta^*))\mathbf{J}_\theta^\top(\mathcal{X}, \mathcal{T}(\theta^*))\left(\mathcal{Y} - f(\mathcal{X}, \theta^*)\right)}{\sigma^2} \\ &= \frac{\Sigma_{\mathbb{Q}}\mathrm{diag}(\theta^*)\mathbf{J}_\theta^\top(\mathcal{X}, \theta^*)\left(\mathcal{Y} - f(\mathcal{X}, \theta^*)\right)}{\sigma^2} \\ &= \frac{\Sigma_{\mathbb{Q}}\mathrm{diag}(\theta^*)\mathbf{J}_\theta^\top\left(\mathcal{Y} - f(\mathcal{X}, \theta^*)\right)}{\sigma^2} \end{aligned}$$

Therefore, equation 6 and equation 7 are invariant to function-preserving scale transformations. The remain part of the theorem is related to the definition of function-preserving scale transformation $\mathcal{T}$. For generalization error, following holds

$$\begin{aligned} \mathrm{err}_{\mathcal{D}}(\mathbb{Q}_{\mathcal{T}(\theta^*)}) &= \mathbb{E}_{(x,y)\sim\mathcal{D}, \psi\sim\mathbb{Q}_{\mathcal{T}(\theta^*)}}[\mathrm{err}(f(x, \psi), y)] \\ &= \mathbb{E}_{(x,y)\sim\mathcal{D}, c\sim\mathbb{Q}_{\mathcal{T}(\theta^*)}}[\mathrm{err}(g_{\theta^*}^{\mathrm{pert}}(x, c), y)] \\ &= \mathbb{E}_{(x,y)\sim\mathcal{D}, c\sim\mathbb{Q}_{\theta^*}}[\mathrm{err}(g_{\theta^*}^{\mathrm{pert}}(x, c), y)] \\ &= \mathbb{E}_{(x,y)\sim\mathcal{D}, \psi\sim\mathbb{Q}_{\theta^*}}[\mathrm{err}(f(x, \psi), y)] \\ &= \mathrm{err}_{\mathcal{D}}(\mathbb{Q}_{\theta^*}) \end{aligned}$$

This proof can be extended to the empirical error $\mathrm{err}_{\mathcal{S}_{\mathbb{Q}}}$. □

A.2 PROOF OF THEOREM 2.3

Proof. **(Construction of KL divergence)** To construct PAC-Bayes-CTK, we need to arrange KL divergence between posterior and prior as follows:

$$\begin{aligned}
\mathrm{KL}[\mathbb{Q}\|\mathbb{P}] &= \frac{1}{2}\left(\mathrm{tr}\left(\Sigma_{\mathbb{P}}^{-1}(\Sigma_{\mathbb{Q}} + (\mu_{\mathbb{Q}} - \mu_{\mathbb{P}})(\mu_{\mathbb{Q}} - \mu_{\mathbb{P}})^\top)\right) + \log|\Sigma_{\mathbb{P}}| - \log|\Sigma_{\mathbb{Q}}| - P\right) \\
&= \frac{1}{2}\mathrm{tr}(\Sigma_{\mathbb{P}}^{-1}(\mu_{\mathbb{Q}} - \mu_{\mathbb{P}})(\mu_{\mathbb{Q}} - \mu_{\mathbb{P}})^\top)) + \frac{1}{2}\left(\mathrm{tr}(\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}) + \log|\Sigma_{\mathbb{P}}| - \log|\Sigma_{\mathbb{Q}}| - P\right) \\
&= \frac{1}{2}(\mu_{\mathbb{Q}} - \mu_{\mathbb{P}})^\top\Sigma_{\mathbb{P}}^{-1}(\mu_{\mathbb{Q}} - \mu_{\mathbb{P}}) + \frac{1}{2}\left(\mathrm{tr}(\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}) - \log|\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}| - P\right) \\
&= \underbrace{\frac{\mu_{\mathbb{Q}}^\top\mu_{\mathbb{Q}}}{2\alpha^2}}_{\text{perturbation}} + \underbrace{\frac{1}{2}\left(\mathrm{tr}(\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}) - \log|\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}| - p\right)}_{\text{sharpness}}
\end{aligned}$$

where the first equality uses the KL divergence between two Gaussian distributions, the thrid equality uses trace property ($\mathrm{tr}(AB) = \mathrm{tr}(BA)$ and $\mathrm{tr}(a) = a$ for scalar a), and the last equality uses the definition of PAC-Bayes prior ($\mathbb{P}_{\theta^*}(c) = \mathcal{N}(c|\mathbf{0}_P, \alpha^2\mathbf{I}_P)$). For sharpness term, we first compute the $\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}$ term as

$$\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}} = \left(\mathbf{I}_P + \frac{\alpha^2}{\sigma^2}\mathbf{J}_c^\top\mathbf{J}_c\right)^{-1}$$

Since $\alpha^2, \sigma^2 > 0$ and $\mathbf{J}_c^\top\mathbf{J}_c$ is positive semi-definite, the matrix $\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}$ have non-zero eigenvalues of $\{\beta_i\}_{i=1}^P$. Since trace is the sum of eigenvalues and log-determinant is the sum of log-eigenvalues, we have

$$\begin{aligned}
\mathrm{KL}[\mathbb{Q}\|\mathbb{P}] &= \frac{\mu_{\mathbb{Q}}^\top\mu_{\mathbb{Q}}}{2\alpha^2} + \frac{1}{2}\sum_{i=1}^{P}(\beta_i - \log(\beta_i) - 1) \\
&= \frac{\mu_{\mathbb{Q}}^\top\mu_{\mathbb{Q}}}{2\alpha^2} + \frac{1}{2}\sum_{i=1}^{P}h(\beta_i)
\end{aligned}$$

where $h(x) = x - \log(x) - 1$. By plugging this KL divergence to the equation 2, we get equation 8.

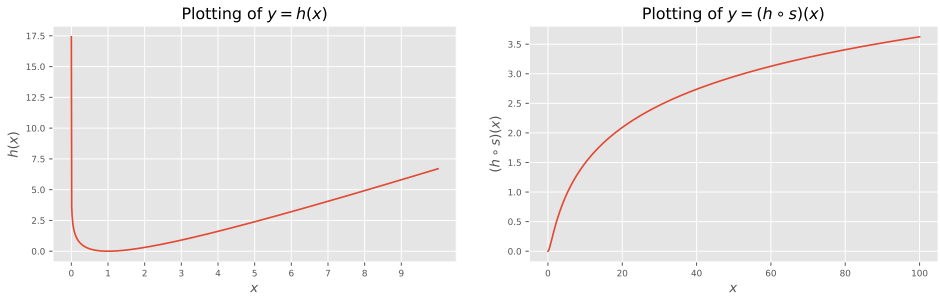
(Eigenvalues of $\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}$) To show the scale-invariance of PAC-Bayes-CTK, it is sufficient to show that KL divergence posterior and prior is scale-invariant: $\frac{\log(2\sqrt{N_{\mathbb{Q}}}/\delta)}{2N_{\mathbb{Q}}}$ is independent to KL PAC-Bayes prior/posterior and we already show the invariance property of empirical/generalization error term in Proposition 2.2. To show the invariance property of KL divergence, we consider the Connectivity Tangent Kernel (CTK) as defined in equation 2.4:

$$\mathbf{C}_{\mathcal{X}}^{\theta^*} := \mathbf{J}_c\mathbf{J}_c^\top = \mathbf{J}_\theta\mathrm{diag}(\theta^*)^2\mathbf{J}_\theta^\top \in \mathbb{R}^{NK\times NK}.$$

Since CTK is a real-symmetric matrix, one can assume the eigenvalue decomposition of CTK as $\mathbf{C}_{\mathcal{X}}^{\theta^*} = Q\Lambda Q^\top$ where $Q \in \mathbb{R}^{NK\times NK}$ is an orthogonal matrix and $\Lambda \in \mathbb{R}^{NK\times NK}$ is a diagonal matrix. Then following holds for $\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}$

$$\begin{aligned}
\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}} &= \left(\mathbf{I}_P + \frac{\alpha^2}{\sigma^2}\mathbf{J}_c^\top\mathbf{J}_c\right)^{-1} \\
&= \left(\mathbf{I}_P + \frac{\alpha^2}{\sigma^2}Q\Lambda Q^\top\right)^{-1} \\
&= Q\left(\mathbf{I}_P + \frac{\alpha^2}{\sigma^2}\Lambda\right)^{-1}Q^\top
\end{aligned}$$

Therefore, eigenvalues of $\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}$ are $\frac{1}{1+\alpha^2\lambda_i/\sigma^2} = \frac{\sigma^2}{\sigma^2+\alpha^2\lambda_i}$ where $\{\lambda_i\}_{i=1}^P$ are eigenvalues of CTK (and diagonal elements of Λ).



(a) $y = h(x)$ (b) $y = (h \circ s)(x)$ where $\sigma = \alpha = 1$

Figure 2: Functions used in proofs

(Scale invariance of CTK) The scale-invariance property of CTK is a simple application of equation 13:

$$\begin{aligned}
\mathbf{C}_{xy}^{\mathcal{T}(\psi)} &= \mathbf{J}_\theta(x, \mathcal{T}(\psi))\mathrm{diag}(\mathcal{T}(\psi)^2)\mathbf{J}_\theta(y, \mathcal{T}(\psi))^\top \\
&= \mathbf{J}_\theta(x, \psi)\mathcal{T}^{-1}\mathcal{T}\mathrm{diag}(\psi)\mathrm{diag}(\psi)\mathcal{T}\mathcal{T}^{-1}\mathbf{J}_\theta(x, \psi)^\top \\
&= \mathbf{J}_\theta(x, \psi)\mathrm{diag}(\psi)\mathrm{diag}(\psi)\mathbf{J}_\theta(x, \psi)^\top \\
&= \mathbf{C}_{xy}^{\psi}\ , \forall x, y \in \mathbb{R}^D, \forall \psi \in \mathbb{R}^P.
\end{aligned}$$

Therefore, CTK is invariant to any function-preserving scale transformation $\mathcal{T}$ and so do its eigenvalues. This guarantees the invariance of $\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}$ and its eigenvalues. In summary, we showed the scale-invariance property of sharpness term of KL divergence. Now all that remains is to show the invariance of the perturbation term. However, this is already proved in the proof of Proposition 2.2. Therefore, we show PAC-Bayes-CTK is invariant to any function-preserving scale transformation. □

A.3 Proof of Corollary 2.4

Proof. In proof of Theorem 2.3, we showed that eigenvalues of $\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}$ can be represented as

$$\frac{\sigma^2}{\sigma^2 + \alpha^2\lambda_i}$$

where $\{\lambda_i\}_{i=1}^P$ are eigenvalues of CTK. Now, we identify the eigenvalues of CTK. To this end, we assume the singular value decomposition (SVD) of Jacobian w.r.t. connectivity $\mathbf{J}_c \in \mathbb{R}^{NK \times P}$ as $\mathbf{J}_c = U\Sigma V^\top$ where $U \in \mathbb{R}^{NK \times NK}$ and $V \in \mathbb{R}^{P \times P}$ are orthogonal matrices and $\Sigma \in \mathbb{R}^{NK \times P}$ is a rectangular diagonal matrix. Then, CTK can be represented as $\mathbf{C}_{\mathcal{X}}^{\theta^*} = \mathbf{J}_c\mathbf{J}_c^\top = U\Sigma V^\top V\Sigma U^\top = U\Sigma^2 U^\top$. In summary, the column vectors of U are eigenvectors of CTK and eigenvalues of CTK are square of singular values of $\mathbf{J}_c$ and so $\lambda_i \geq 0, \forall i$. Therefore $\beta_i \leq 1$ for all $i = 1, \ldots, P$ for eigenvalues $\{\beta_i\}_{i=1}^P$ of $\Sigma_{\mathbb{P}}^{-1}\Sigma_{\mathbb{Q}}$ and equality holds for $\lambda_i = 0$. Now all that remains is to show that the sharpness term of PAC-Bayes-CTK is a monotonically increasing function on each eigenvalues of CTK. To show this, we first keep in mind that

$$s(x) := \frac{\sigma^2}{\sigma^2 + \alpha^2 x}$$

is a monotonically decreasing function for $x \geq 0$ and $h(x) := x - \log(x) - 1$ is a monotonically decreasing function for $x \in (0, 1]$. Since sharpness term of KL divergence is

$$\sum_{i=1}^{P} \frac{h(\beta_i)}{4N_{\mathbb{Q}}} = \sum_{i=1}^{P} \frac{(h \circ s)(\lambda_i)}{4N_{\mathbb{Q}}}$$

This is a monotonically increasing function for $x \geq 0$ since $s(x) \leq 1$ for $x \geq 0$. For your information, we plot $y = h(x)$ and $y = (h \circ s)(x)$ in Figure 2.

□

A.4 Proof of Proposition 2.5

We refer to **Scale invariance of CTK** part of proof of Theorem 2.3. This is a direct application of scale-invariance property of Jacobian w.r.t. connectivity.

A.5 Proof of Corollary 2.7

Proof. Since CS is trace of CTK, it is a sum of eigenvalues of CTK. As shown in the proof of Corollary 2.4, eigenvalues of CTK are square of singular values of Jacobian w.r.t. connectivity c. Therefore, eigenvalues of CTK are non-negative and vanishes to zero if CS vanishes to zero.

$$\sum_{i=1}^{P} \lambda_i = 0 \Rightarrow \lambda_i = 0 \Rightarrow \beta_i = s(\lambda_i) = 1 \Rightarrow h(\beta_i) = 0, \quad \forall i = 1, \dots, P$$

This means the sharpness term of KL divergence vanishes to zero. Furthermore, singular values of Jacobian w.r.t. c also vanishes to zero in this case. Therefore, $\mu_{\mathbb{Q}}$ vanishes to zero, also. Similarly, if CS diverges to infinity, this means (at least) one of eigenvalues of CTK diverges to infinity. In this case, following holds

$$\lambda_i \to \infty \Rightarrow \beta_i = s(\lambda_i) \to 0 \Rightarrow h(\beta_i) \to \infty, \quad \forall i = 1, \dots, P$$

Therefore, KL divergence term of PAC-Bayes-CTK also diverges to infinity. □

A.6 Proof of Proposition 3.1

Proof. By assumption, we fixed all non-scale invariant parameters. This means we exclude these parameters in sampling procedure of CL and LL. In terms of predictive distribution, this can be translated as

$$f^{\text{lin}}_{\theta^*}(x, \psi) | p_{\text{LA}}(\psi|\mathcal{S}) \sim \mathcal{N}(f(x, \theta^*), \alpha^2 \hat{\Theta}^{\theta^*}_{xx} - \alpha^2 \hat{\Theta}^{\theta^*}_{x\mathcal{X}} \hat{\Theta} \mathcal{X}^{\theta^*-1} \hat{\Theta}^{\theta^*}_{\mathcal{X}x})$$
$$f^{\text{lin}}_{\theta^*}(x, \psi) | p_{\text{CL}}(\psi|\mathcal{S}) \sim \mathcal{N}(f(x, \theta^*), \alpha^2 \hat{\mathbf{C}}^{\theta^*}_{xx} - \alpha^2 \hat{\mathbf{C}}^{\theta^*}_{x\mathcal{X}} \hat{\mathbf{C}}^{\theta^*-1}_{\mathcal{X}} \hat{\mathbf{C}}^{\theta^*}_{\mathcal{X}x})$$

where $\hat{\Theta}^{\psi}_{xx'} := \sum_{i \in \mathcal{P}} \frac{\partial f(x,\psi)}{\partial \theta_i} \frac{\partial f(x',\psi)}{\partial \theta_i}$ and $\hat{\mathbf{C}}^{\psi}_{xx'} := \sum_{i \in \mathcal{P}} \frac{\partial f(x,\psi)}{\partial \theta_i} \frac{\partial f(x',\psi)}{\partial \theta_i} (\psi_i)^2$ for scale-invariant parameter set $\mathcal{P}$. Thereby, we mask the gradient of non scale-invariant parameter as zero. Therefore, this can be arrange as follows

$$\hat{\Theta}^{\psi}_{xx'} = \mathbf{J}_\theta(x, \psi) \text{diag}(\mathbf{1}_{\mathcal{P}}) \mathbf{J}_\theta(x, \psi)^\top, \quad \hat{\mathbf{C}}^{\psi}_{xx'} = \mathbf{J}_\theta(x, \psi) \text{diag}(\psi) \text{diag}(\mathbf{1}_{\mathcal{P}}) \text{diag}(\psi) \mathbf{J}_\theta(x, \psi)^\top$$

where $\mathbf{1}_{\mathcal{P}} \in \mathbb{R}^P$ is a masking vector (i.e., one for included components and zero for excluded components). Then, the weight decay regularization for scale-invariant parameters can be represented as

$$\mathcal{W}_\gamma(\psi)_i = \begin{cases} \gamma \psi_i, & \text{if} \quad \psi_i \in \mathcal{P}. \\ \psi_i, & \text{if} \quad \psi_i \notin \mathcal{P}. \end{cases}$$

Therefore, we get

$$\begin{aligned} \hat{\Theta}^{\mathcal{W}_\gamma(\psi)}_{xx'} &= \mathbf{J}_\theta(x, \mathcal{W}_\gamma(\psi)) \text{diag}(\mathbf{1}_{\mathcal{P}}) \mathbf{J}_\theta(x, \mathcal{W}_\gamma(\psi)))^\top \\ &= \mathbf{J}_\theta(x, \psi) \mathcal{W}_\gamma^{-1} \text{diag}(\mathbf{1}_{\mathcal{P}}) \mathcal{W}_\gamma^{-1} \mathbf{J}_\theta(x, \psi)^\top \\ &= \mathbf{J}_\theta(x, \psi) \text{diag}(\mathbf{1}_{\mathcal{P}}/\gamma^2) \mathbf{J}_\theta(x, \psi)^\top \\ &= 1/\gamma^2 \mathbf{J}_\theta(x, \psi) \text{diag}(\mathbf{1}_{\mathcal{P}}) \mathbf{J}_\theta(x, \psi)^\top \\ &= 1/\gamma^2 \hat{\Theta}^{\psi}_{xx'} \end{aligned}$$

for empirical NTK and

$$\begin{aligned} \hat{\mathbf{C}}^{\mathcal{W}_\gamma(\psi)}_{xx'} &= \mathbf{J}_\theta(x, \mathcal{W}_\gamma(\psi)) \text{diag}(\mathcal{W}_\gamma(\psi)) \text{diag}(\mathbf{1}_{\mathcal{P}}) \text{diag}(\mathcal{W}_\gamma(\psi)) \mathbf{J}_\theta(x, \mathcal{W}_\gamma(\psi)))^\top \\ &= \mathbf{J}_\theta(x, \psi) \mathcal{W}_\gamma^{-1} \mathcal{W}_\gamma \text{diag}(\psi) \text{diag}(\mathbf{1}_{\mathcal{P}}) \text{diag}(\psi) \mathcal{W}_\gamma \mathcal{W}_\gamma^{-1} \mathbf{J}_\theta(x, \psi)^\top \\ &= \mathbf{J}_\theta(x, \psi) \text{diag}(\psi) \text{diag}(\mathbf{1}_{\mathcal{P}}) \text{diag}(\psi) \mathbf{J}_\theta(x, \psi)^\top \\ &= \hat{\mathbf{C}}^{\psi}_{xx'} \end{aligned}$$

for empirical CTK. Therefore, we get

$$f^{\text{lin}}_{\mathcal{W}_\gamma(\theta^*)}(x,\psi)|p_{\text{LA}}(\psi|\mathcal{S}) \sim \mathcal{N}(f(x,\theta^*), \alpha^2/\gamma^2\hat{\Theta}^{\theta^*}_{xx} - \alpha^2/\gamma^2\hat{\Theta}^{\theta^*}_{x\mathcal{X}}\hat{\Theta}^{\theta^*}_{\mathcal{X}}{}^{-1}\hat{\Theta}^{\theta^*}_{\mathcal{X}x})$$
$$f^{\text{lin}}_{\mathcal{W}_\gamma(\theta^*)}(x,\psi)|p_{\text{CL}}(\psi|\mathcal{S}) \sim \mathcal{N}(f(x,\theta^*), \alpha^2\hat{\mathbf{C}}^{\theta^*}_{xx} - \alpha^2\hat{\mathbf{C}}^{\theta^*}_{x\mathcal{X}}\hat{\mathbf{C}}^{\theta^*}_{\mathcal{X}}{}^{-1}\hat{\mathbf{C}}^{\theta^*}_{\mathcal{X}x})$$

This gives us

$$\text{Var}_{\psi\sim p_{\text{LA}}(\psi|\mathcal{S})}(f^{\text{lin}}_{\mathcal{W}_\gamma(\theta^*)}(x,\psi)) = \text{Var}_{\psi\sim p_{\text{LA}}(\psi|\mathcal{S})}(f^{\text{lin}}_{\theta^*}(x,\psi))/\gamma^2$$
$$\text{Var}_{\psi\sim p_{\text{CL}}(\psi|\mathcal{S})}(f^{\text{lin}}_{\mathcal{W}_\gamma(\theta^*)}(x,\psi)) = \text{Var}_{\psi\sim p_{\text{CL}}(\psi|\mathcal{S})}(f^{\text{lin}}_{\theta^*}(x,\psi))$$

□

A.7 Derivation of PAC-Bayes-NTK

Theorem A.1 (PAC-Bayes-NTK)**.** *Let us assume pre-trained parameter θ^* with data $\mathcal{S}_{\mathbb{P}}$. Let us assume PAC-Bayes prior and posterior as*

$$\mathbb{P}'_{\theta*}(\delta) := \mathcal{N}(\delta|\mathbf{0}_P, \alpha^2\mathbf{I}_P) \tag{14}$$
$$\mathbb{Q}'_{\theta^*}(\delta) := \mathcal{N}(\delta|\mu_{\mathbb{Q}'}, \Sigma_{\mathbb{Q}'}) \tag{15}$$
$$\mu_{\mathbb{Q}'} := \frac{\Sigma_{\mathbb{Q}'}\mathbf{J}_\theta^\top(\mathcal{Y} - f(\mathcal{X},\theta^*))}{\sigma^2} \tag{16}$$
$$\Sigma_{\mathbb{Q}'} := \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_\theta^\top\mathbf{J}_\theta}{\sigma^2}\right)^{-1} \tag{17}$$

By applying $\mathbb{P}'_{\theta^*}, \mathbb{Q}'_{\theta^*}$ to data-dependent PAC-Bayes bound (equation 2), we get

$$\text{err}_{\mathcal{D}}(\mathbb{Q}'_{\theta^*}) \leq \text{err}_{\mathcal{S}_{\mathbb{Q}'}}(\mathbb{Q}'_{\theta^*}) + \sqrt{\overbrace{\underbrace{\frac{\mu_{\mathbb{Q}'}^\top\mu_{\mathbb{Q}'}}{4\alpha^2 N_{\mathbb{Q}'}}}_{\textit{(average) perturbation}} + \underbrace{\sum_{i=1}^{P}\frac{h(\beta'_i)}{4N_{\mathbb{Q}'}}}_{\textit{sharpness}}}^{\textit{KL divergence}} + \frac{\log(2\sqrt{N_{\mathbb{Q}'}}/\delta)}{2N_{\mathbb{Q}'}}} \tag{18}$$

where $\{\beta'_i\}_{i=1}^P$ are eigenvalues of $(\mathbf{I}_P + \frac{\alpha^2}{\sigma^2}\mathbf{J}_\theta^\top\mathbf{J}_\theta)^{-1}$ and $h(x) := x - \log(x) - 1$. This upper bound is not scale-invariant in general.

Proof. The main difference between PAC-Bayes-CTK and PAC-Bayes-NTK is the definition of Jacobian: PAC-Bayes-CTK use Jacobian w.r.t connectivity and PAC-Bayes-NTK use Jacobian w.r.t. parameter. Therefore, **Construction of KL divergence** of proof of Theorem 2.3 is preserved except

$$\Sigma_{\mathbb{P}'}^{-1}\Sigma_{\mathbb{Q}'} = (\mathbf{I}_P + \frac{\alpha^2}{\sigma^2}\mathbf{J}_\theta^\top\mathbf{J}_\theta)^{-1}$$

and β'_i are eigenvalues of $\Sigma_{\mathbb{P}'}^{-1}\Sigma_{\mathbb{Q}'}$. Note that these eigenvalues satisfies

$$\beta'_i = \frac{\sigma^2}{\sigma^2 + \alpha^2\lambda'_i}$$

where λ'_i are eigenvalues of NTK. □

Remark A.2 (Function-preserving scale transformation to NTK)**.** *On the contrary to the CTK, scale invariance property is not applicable to the NTK due to Jacobian w.r.t. parameter:*

$$\mathbf{J}_\theta(x,\mathcal{T}(\psi)) = \frac{\partial}{\partial\mathcal{T}(\psi)}f(x,\mathcal{T}(\psi)) = \frac{\partial}{\partial\mathcal{T}(\psi)}f(x,\psi) = \mathbf{J}_\theta(x,\psi)\mathcal{T}^{-1}$$

If we assume all parameters are scale-invariant (or equivalently masking the Jacobian for all non scale-invariant parameters as in the proof of Proposition 3.1), the scale of NTK is proportional to the inverse scale of parameters.

A.8 DETERMINISTIC LIMITING KERNEL OF CTK

Theorem A.3 (Deterministic limiting kernel of CTK)**.** *Let us assume L-layered network with Lipschitz activation function and NN with NTK initialization. Then the empirical CTK converges in probability to a deterministic limiting kernel $\mathbf{C}_{xy}$ as the layers width $n_1, \ldots, n_L \to \infty$ sequentially. Furthermore, $\mathbf{C}_{xy} = \Theta_{xy}$ holds.*

Proof. The proof is a modification to proof of convergence of NTK in Jacot et al. (2018) considering NTK initialization (i.e. standard Gaussian for all parameters). We provide proof by induction. For single layer network, The CTK is summed as:

$$(\mathbf{C}_{xx'})_{kk'} = \frac{1}{n_0} \sum_{i=1}^{n_0} \sum_{j=1}^{n_1} x_i x_i^{'} \delta_{jk} \delta_{jk'} W_{ik} W_{ik'} + \beta^2 \sum_{j=1}^{n_1} \delta_{jk} \delta_{jk'}$$
$$\to (\Theta_{xx})_{kk'}$$

since the weight is sampled from standard Gaussian distribution, whose variance is 1, and product of two (independent) random variable converges in probability converges to the product of converged values. If we assume CTK of l-th layer is converged to NTK of l-th layer in probability, then the convergence of the $(l+1)$-th layer is also satisfied since multiplication of two random weights, which converges to 1, is multiplied to the empirical NTK of $(l+1)$-th layer, which converges to the deterministic limiting NTK of $(l+1)$-th layer. Therefore, empirical CTK converges in probability to the deterministic limiting CTK, which is equivalent to the deterministic limiting NTK. □

B DETAILS OF SQUARED LOSS FOR CLASSIFICATION TASKS

For the classification tasks in Sec. 4.2, we use the squared loss instead of the cross-entropy loss since our theoretical results are built on the squared loss. Here, we describe how we use the squared loss to mimic the cross-entropy loss. There are several works (Lee et al., 2020; Hui & Belkin, 2021) that utilized the squared loss for the classification task instead of the cross-entropy loss. Specifically, Lee et al. (2020) used

$$\mathcal{L}(\mathcal{S}, \theta) = \frac{1}{2NK} \sum_{(x_i, y_i) \in \mathcal{S}} \|f(x_i, \theta) - y_i\|^2$$

where C is the number of classes, and Hui & Belkin (2021) used

$$\ell((x, c), \theta) = \frac{1}{2K} \left(k(f_c(x, \theta) - M)^2 + \sum_{i=1, i \neq c}^{K} f_i(x, \theta)^2 \right)$$

for single data loss, where $\ell((x, c), \theta)$ is sample loss given input x, target c and parameter θ, $f_i(x, \theta) \in \mathbb{R}$ is the i-th component of $f(x, \theta) \in \mathbb{R}^K$, k and M are dataset-specific hyper-parameters.

These works used the mean for reducing the vector-valued loss into a scalar loss. However, this can be problematic when the number of classes is large. When the number of classes increases, the denominator of the mean (the number of classes) increases while the target value remains 1 (one-hot label). As a result, the scale of a gradient for the target class becomes smaller. To avoid such an unfavorable effect, we just use the sum for reducing vector-valued loss into a scalar loss instead of taking the mean, i.e.,

$$\ell((x, c), \theta) = \frac{1}{2} \left((f_c(x, \theta) - 1)^2 + \sum_{i=1, i \neq c}^{K} f_i(x, \theta)^2 \right)$$

This analysis is consistent with the hyper-parameter selection in Hui & Belkin (2021). They used larger k and M as the number of classes increases (e.g., $k = 1, M = 1$ for CIFAR-10 (Krizhevsky, 2009), but $k = 15, M = 30$ for ImageNet (Deng et al., 2009)) which results in manually compensating the scale of gradient to the target class label.

C Derivation of PAC-Bayes posterior

Derivation of $\mathbb{Q}_{\theta^*}(c)$

For Bayesian linear regression, we compute the posterior of $\beta \in \mathbb{R}^P$

$$y_i = x_i\beta + \epsilon_i, \quad \text{for } i = 1 \ldots, M$$

where $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ is i.i.d. sampled and the prior of β is given as $\beta \sim \mathcal{N}(\mathbf{0}_P, \alpha^2\mathbf{I}_P)$. By concatenating this, we get

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon$$

where $\mathbf{y} \in \mathbb{R}^M, \mathbf{X} \in \mathbb{R}^{M\times p}$ are concatenation of y_i, x_i, respectively and $\varepsilon \sim \mathcal{N}(\mathbf{0}_M, \sigma^2\mathbf{I}_M)$. It is well known (Bishop, 2006; Murphy, 2012) that the posterior of β for this problem is

$$\begin{aligned}
&\beta \sim \mathcal{N}\left(\beta|\mu, \Sigma\right) \\
&\mu := \frac{\Sigma\mathbf{X}^\top y}{\sigma^2} \\
&\Sigma := \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{X}^\top\mathbf{X}}{\sigma^2}\right)^{-1}.
\end{aligned}$$

Similarly, we define Bayesian linear regression problem as

$$y_i = f(x_i, \theta^*) + \mathbf{J}_\theta(x_i, \theta^*)\mathrm{diag}(\theta^*)c + \epsilon_i, \quad \text{for } i = 1 \ldots, NK$$

where $M = NK$ and the regression coefficient is $\beta = c$ in this case. Thus, we treat $y_i - f(x_i, \theta^*)$ as a target and $\mathbf{J}_\theta(x_i, \theta^*)\mathrm{diag}(\theta^*)$ as an input of linear regression problem. By concatenating this, we get

$$\mathcal{Y} = f(\mathcal{X}, \theta^*) + \mathbf{J}_c c + \varepsilon \Rightarrow (\mathcal{Y} - f(\mathcal{X}, \theta^*)) = \mathbf{J}_c c + \varepsilon.$$

By plugging this to the posterior of Bayesian linear regression problem, we get

$$\begin{aligned}
&\mathbb{Q}_{\theta^*}(c) := \mathcal{N}(c|\mu_{\mathbb{Q}}, \Sigma_{\mathbb{Q}}) \\
&\mu_{\mathbb{Q}} := \frac{\Sigma_{\mathbb{Q}}\mathbf{J}_c^\top\left(\mathcal{Y} - f(\mathcal{X}, \theta^*)\right)}{\sigma^2} = \frac{\Sigma_{\mathbb{Q}}\mathrm{diag}(\theta^*)\mathbf{J}_\theta^\top\left(\mathcal{Y} - f(\mathcal{X}, \theta^*)\right)}{\sigma^2} \\
&\Sigma_{\mathbb{Q}} := \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_c^\top\mathbf{J}_c}{\sigma^2}\right)^{-1} = \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathrm{diag}(\theta^*)\mathbf{J}_\theta^\top\mathbf{J}_\theta\mathrm{diag}(\theta^*)}{\sigma^2}\right)^{-1}
\end{aligned}$$

Derivation of $\mathbb{Q}_{\theta^*}(\psi)$ We define perturbed parameter ψ as follows

$$\psi := \theta^* + \theta^* \odot c.$$

Since ψ is affine to c, we get the distribution of ψ as

$$\begin{aligned}
&\mathbb{Q}_{\theta^*}(\psi) := \mathcal{N}(\psi|\mu_{\mathbb{Q}}^\psi, \Sigma_{\mathbb{Q}}^\psi) \\
&\mu_{\mathbb{Q}}^\psi := \theta^* + \theta^* \odot \mu_{\mathbb{Q}} \\
&\Sigma_{\mathbb{Q}}^\psi := \mathrm{diag}(\theta^*)\Sigma_{\mathbb{Q}}\mathrm{diag}(\theta^*) = \left(\frac{\mathrm{diag}(\theta^*)^{-2}}{\alpha^2} + \frac{\mathbf{J}_\theta^\top\mathbf{J}_\theta}{\sigma^2}\right)^{-1}
\end{aligned}$$

D Representative cases of function-preserving scaling transformations

Activation-wise rescaling transformation (Tsuzuku et al., 2020; Neyshabur et al., 2015a) For NNs with ReLU activations, following holds for $\forall x \in \mathbb{R}^d, \gamma > 0$: $f(x, \theta) = f(x, \mathcal{R}_{\gamma,l,k}(\theta))$, where rescale transformation $\mathcal{R}_{\gamma,l,k}(\cdot)$[1] is defined as

$$(\mathcal{R}_{\gamma,l,k}(\theta))_i = \begin{cases} \gamma \cdot \theta_i & \text{, if } \theta_i \in \{\text{param. subset connecting as input edges to } k\text{-th activation at } l\text{-th layer}\} \\ \theta_i/\gamma & \text{, if } \theta_i \in \{\text{param. subset connecting as output edges to } k\text{-th activation at } l\text{-th layer}\} \\ \theta_i & \text{, for } \theta_i \text{ in the other cases} \end{cases} \tag{19}$$

[1]For a simple two layer linear NN $f(x) := W_2\sigma(W_1x)$ with weight matrix W_1, W_2, the first case of equation 19 corresponds to k-th row of W_1 and the second case of equation 19 corresponds to k-th column of W_2.

Note that $\mathcal{R}_{\gamma,l,k}(\cdot)$ is a finer-grained rescaling transformation than layer-wise rescaling transformation (i.e. common γ for all activations in layer l) discussed in Dinh et al. (2017). Dinh et al. (2017) showed that even layer-wise rescaling transformations can sharpen pre-trained solutions in terms of trace of Hessian (i.e., contradicting the FM hypothesis). This contradiction also occurs to previous PAC-Bayes bounds (Tsuzuku et al., 2020; Kwon et al., 2021) due to the scale-dependent term.

Weight decay with BN layers (Ioffe & Szegedy, 2015) For parameters $W \in \mathbb{R}^{n_l \times n_{l-1}}$ preceding BN layer,

$$\mathrm{BN}((\mathrm{diag}(\gamma)W)u) = \mathrm{BN}(Wu) \tag{20}$$

for an input $u \in \mathbb{R}^{n_{l-1}}$ and a positive vector $\gamma \in \mathbb{R}_+^{n_l}$. This implies that scaling transformations on these parameters preserve function represented by NNs for $\forall x \in \mathbb{R}^d, \gamma \in \mathbb{R}_+^{n_l}$: $f(x,\theta) = f(x, \mathcal{S}_{\gamma,l,k}(\theta))$, where scaling transformation $\mathcal{S}_{\gamma,l,k}(\cdot)$ is defined for $i = 1, \ldots, P$

$$(\mathcal{S}_{\gamma,l,k}(\theta))_i = \begin{cases} \gamma_k \cdot \theta_i & \text{, if } \theta_i \in \{\text{param. subset connecting as input edges to } k\text{-th activation at } l\text{-th layer}\} \\ \theta_i & \text{, for } \theta_i \text{ in the other cases} \end{cases} \tag{21}$$

Note that the weight decay regularization (Loshchilov & Hutter, 2019; Zhang et al., 2019) can be implemented as a realization of $\mathcal{S}_{\gamma,l,k}(\cdot)$ (e.g., $\gamma = 0.9995$ for all activations preceding BN layers). Therefore, thanks to Theorem 2.2 and Theorem 2.5, our CTK-based bound is invariant to weight decay regularization applied to parameters before BN layers. We also refer to (Van Laarhoven, 2017; Lobacheva et al., 2021) for optimization perspective of weight decay with BN.

E Implementation of Connectivity Laplace

To estimate the empirical/generalization bound in Sec. 2.4 and calibrate uncertainty in Sec. 4.2, we need to sample c from the posterior $\mathbb{Q}_{\theta^*}(c)$. For this, we sample perturbations δ in connectivity space

$$\delta \sim \mathcal{N}\left(\delta | \mathbf{0}_P, \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_c^\top \mathbf{J}_c}{\sigma^2}\right)^{-1}\right)$$

so that $c = \mu_{\mathbb{Q}} + \delta$ for equation 6. To sample this, we provide a novel approach to sample from LA/CL without curvature approximation. To this end, we consider following optimization problem

$$\arg\min_c L(c) := \arg\min_c \frac{1}{2N\sigma^2}\|\mathcal{Y} - f(\mathcal{X},\theta^*) - \mathbf{J}_c c + \varepsilon\|^2 + \frac{1}{2N\alpha^2}\|c - c_0\|_2^2$$

where $\varepsilon \sim \mathcal{N}(\varepsilon | \mathbf{0}_{NK}, \sigma^2 \mathbf{I}_{NK})$ and $c_0 \sim \mathcal{N}(c_0 | \mathbf{0}_P, \alpha^2 \mathbf{I}_P)$. By first-order optimality condition, we have

$$N\nabla_c L(c) = -\frac{\mathbf{J}_c^\top(\mathcal{Y} - f(\mathcal{X},\theta^*) - \mathbf{J}_c c^* + \varepsilon)}{\sigma^2} + \frac{c^* - c_0}{\alpha^2} = \mathbf{0}_P.$$

By arranging this w.r.t. optimizer c^*, we get

$$\begin{aligned} c^* &= \left(\mathbf{J}_c^\top \mathbf{J}_c + \frac{\sigma^2}{\alpha^2}\mathbf{I}_P\right)^{-1}\left(\mathbf{J}_c^\top(\mathcal{Y} - f(\mathcal{X},\theta^*)) + \frac{\sigma^2}{\alpha^2}c_0 + \mathbf{J}_c\varepsilon\right) \\ &= \left(\mathbf{J}_c^\top \mathbf{J}_c + \frac{\sigma^2}{\alpha^2}\mathbf{I}_P\right)^{-1}\mathbf{J}_c^\top(\mathcal{Y} - f(\mathcal{X},\theta^*)) + \left(\mathbf{J}_c^\top \mathbf{J}_c + \frac{\sigma^2}{\alpha^2}\mathbf{I}_P\right)^{-1}\left(\frac{\sigma^2}{\alpha^2}c_0 + \mathbf{J}_c\varepsilon\right) \\ &= \underbrace{\left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_c^\top \mathbf{J}_c}{\sigma^2}\right)^{-1}\frac{\mathbf{J}_c^\top(\mathcal{Y} - f(\mathcal{X},\theta^*))}{\sigma^2}}_{\text{Deterministic}} + \underbrace{\left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_c^\top \mathbf{J}_c}{\sigma^2}\right)^{-1}\left(\frac{c_0}{\alpha^2} + \frac{\mathbf{J}_c^\top \varepsilon}{\sigma^2}\right)}_{\text{Stochastic}} \end{aligned}$$

Since both ε and c_0 are sampled from independent Gaussian distributions, we have

$$z := \left(\frac{c_0}{\alpha^2} + \frac{\mathbf{J}_c^\top \varepsilon}{\sigma^2}\right) \sim \mathcal{N}\left(z | \mathbf{0}_P, \frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_c^\top \mathbf{J}_c}{\sigma^2}\right)$$

Therefore, optimal solution of randomized optimization problem $\arg\min_c L(c)$ is

$$c \sim \mathcal{N}\left(c \,\middle|\, \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_c^\top \mathbf{J}_c}{\sigma^2}\right)^{-1} \frac{\mathbf{J}_c^\top(\mathcal{Y} - f(\mathcal{X}, \theta^*))}{\sigma^2}, \left(\frac{\mathbf{I}_P}{\alpha^2} + \frac{\mathbf{J}_c^\top \mathbf{J}_c}{\sigma^2}\right)^{-1}\right) = \mathcal{N}(c|\mu_{\mathbb{Q}}, \Sigma_{\mathbb{Q}})$$

Similarly, sampling from CL can be implemented as a following optimization problem.

$$\arg\min_c L(c) := \arg\min_c \frac{1}{2N\sigma^2}\|\mathbf{J}_c c - \varepsilon\|^2 + \frac{1}{2N\alpha^2}\|c - c_0\|_2^2$$

where $\varepsilon \sim \mathcal{N}(\varepsilon|\mathbf{0}_{NK}, \sigma^2\mathbf{I}_{NK})$ and $c_0 \sim \mathcal{N}(c_0|\mathbf{0}_P, \alpha^2\mathbf{I}_P)$. Since we sample the noise of data/perturbation and optimize the perturbation, this can be interpreted as a Randomize-Then-Optimize implementation of Laplace Approximation and Connectivity Laplace (Bardsley et al., 2014; Matthews et al., 2017).

F Details of Computing Connectivity Sharpness

It is well known that empirical NTK or Jacobian is intractable in modern architecture of NNs (e.g., ResNet (He et al., 2016) or BERT (Devlin et al., 2018)). Therefore, one might wonder how Connectivity Sharpness can be computed for these architectures. However, Connectivity Sharpness in Sec. 2.5 is defined as trace of empirical CTK, thereby one can compute CS with Hutchison's method (Hutchinson, 1989; Meyer et al., 2021). According to Hutchison's method, trace of a matrix $A \in \mathbb{R}^{m\times m}$ is

$$\mathrm{tr}(A) = \mathrm{tr}(A\mathbf{I}_p) = \mathrm{tr}(A\mathbb{E}_z[zz^\top]) = \mathbb{E}_z[\mathrm{tr}(Azz^\top)] = \mathbb{E}_z[\mathrm{tr}(z^\top Az)] = \mathbb{E}_z[z^\top Az]$$

where $z \in \mathbb{R}^m$ is a random variable with $\mathrm{cov}(z) = \mathbf{I}_m$ (e.g., standard normal distribution or Rademacher distribution). Since $A = \mathbf{C}_{\mathcal{X}}^{\theta^*} = \mathbf{J}_c(\mathcal{M}, \mathbf{0}_p)\mathbf{J}_c(\mathcal{M}, \mathbf{0}_p)^\top \in \mathbb{R}^{Nk}$ in our case, we further use mini-batch approximation to compute $z^\top Az$: (i) Sample $z_M \in \mathbb{R}^{Mk}$ from Rademacher distribution for mini-batch $\mathcal{M}$ with size M and (ii) compute $v_{\mathcal{M}} := \mathbf{J}_c(\mathcal{M}, \mathbf{0}_p)^\top z_{\mathcal{M}} \in \mathbb{R}^P$ with Jacobian-vector product of JAX (Bradbury et al., 2018) and (iii) compute $x_{\mathcal{M}} = \|v_{\mathcal{M}}\|_2^2$. Then, the sum of x_M for all mini-batch in training dataset is a Monte-Carlo approximation of CS with sample size 1. Empirically, we found that this approximation is sufficiently stable to capture the correlation between sharpness and generalization as shown in Sec. 4.1.

G Predictive Uncertainty of Connectivity/Linearized Laplace

In this section, we derive predictive uncertainty of Linearized Laplace (LL) and Connectivity Laplace (CL). By matrix inversion lemma (Woodbury, 1950), the weight covariance of LL is

$$(\mathbf{I}_p/\alpha^2 + \mathbf{J}_\theta(\mathcal{X}, \theta^*)^\top \mathbf{J}_\theta(\mathcal{X}, \theta^*)/\sigma^2)^{-1} = \alpha^2\mathbf{I}_p - \alpha^2\mathbf{J}_\theta(\mathcal{X}, \theta^*)^\top(\frac{\sigma^2}{\alpha^2}\mathbf{I}_{Nk} + \Theta_{\mathcal{X}\mathcal{X}}^{\theta^*})^{-1}\mathbf{J}_\theta(\mathcal{X}, \theta^*).$$

Therefore, if $\sigma^2/\alpha^2 \to 0$, then the weight covariance of LL converges to

$$\alpha^2\mathbf{I}_p - \alpha^2\mathbf{J}_\theta(\mathcal{X}, \theta^*)^\top \Theta_{\mathcal{X}\mathcal{X}}^{\theta^*-1}\mathbf{J}_\theta(\mathcal{X}, \theta^*).$$

With this weight covariance and linearized NN, the predictive uncertainty of LL is

$$f_{\theta^*}^{\mathrm{lin}}(x, \theta)|p_{\mathrm{LA}}(\theta|\mathcal{S}) \sim \mathcal{N}(f(x, \theta^*), \alpha^2\Theta_{xx}^{\theta^*} - \alpha^2\Theta_{x\mathcal{X}}^{\theta^*}\Theta_{\mathcal{X}\mathcal{X}}^{\theta^*-1}\Theta_{\mathcal{X}x}^{\theta^*}).$$

Similarly, the predictive uncertainty of CL is

$$f_{\theta^*}^{\mathrm{lin}}(x, \theta)|\mathbb{Q}_{\theta^*}(\theta) \sim \mathcal{N}(f(x, \theta^*), \alpha^2\mathbf{C}_{xx}^{\theta^*} - \alpha^2\mathbf{C}_{x\mathcal{X}}^{\theta^*}\mathbf{C}_{\mathcal{X}\mathcal{X}}^{\theta^*-1}\mathbf{C}_{\mathcal{X}x}^{\theta^*}).$$

H Details on Sharpness-Generalization Experiments

To verify that the CS has a better correlation with generalization performance compared to existing sharpness measures, we evaluate the three metrics: (a) Kendall's rank-correlation coefficient τ

Table 5: Configuration of hyper-parameter

network depth	1, 2, 3
network width	32, 64, 128
learning rate	0.1, 0.032, 0.001
weight decay	0.0, 1e-4, 5e-4
mini-batch size	256, 1024, 4096

Table 6: Example of network configuration with respect to the depth 1, width 128 in (Simonyan & Zisserman, 2015)-style.

ConvNet Configuration
input (224 $\times$ 224 RGB image)
Conv3-128 BN ReLU
MaxPool
Conv3-256 BN ReLU
MaxPool
Conv3-512 BN ReLU
MaxPool
Conv3-1024 BN ReLU
MaxPool
Conv3-1024 BN ReLU
MaxPool
FC-4096 ReLU
FC-4096 ReLU
FC-1000

(Kendall, 1938) which considers the consistency of a sharpness measure with generalization gap (i.e., if one has higher sharpness, then so has higher generalization gap) (b) granulated Kendall's coefficient (Jiang et al., 2020) which examines Kendall's rank-correlation coefficient w.r.t. individual hyper-parameters to separately evaluate the effect of each hyper-parameter to generalization gap (c) conditional independence test (Jiang et al., 2020) which captures the causal relationship between measure and generalization.

Three metrics are compared with the following baselines: trace of Hessian ($\mathrm{tr}(\mathbf{H})$; (Keskar et al., 2017)), trace of Fisher information matrix ($\mathrm{tr}(\mathbf{F})$; (Jastrzebski et al., 2021)), trace of empirical NTK at θ^* ($\mathrm{tr}(\mathbf{\Theta}^{\theta^*})$), and four PAC-Bayes bound based measures, sharpness-orig (SO), pacbayes-orig (PO), $1/\alpha'$ sharpness mag (SM), and $1/\sigma'$ pacbayes mag (PM), which are eq. (52), (49), (62), (61) in Jiang et al. (2020).

For the granulated Kendall's coefficient, we use 5 hyper-parameters : network depth, network width, learning rate, weight decay and mini-batch size, along with 3 options for each hyper-parameters as in Table 5. We use the VGG-13 (Simonyan & Zisserman, 2015) as a base model and we adjust the depth and width of each conv block. We add BN layers after the convolution layer for each block. Specifically, the number of convolution layers of each conv block is the depth and the number of channels of convolution layers of the first conv block is the width. For the subsequent conv blocks, we follow the original VGG width multipliers ($\times 2$, $\times 4$, $\times 8$). An example with depth 1 and width 128 is depicted in Table 6.

We use SGD optimizer with a momentum 0.9. We train each model for 200 epochs and use cosine learning rate scheduler (Loshchilov & Hutter, 2016) with 30% of initial epochs as warm-up epochs. The standard data augmentations (padding, random crop, random horizontal flip, and normalization) for CIFAR-10 is used for training data. For the analysis, we only use models with above 99% training accuracy following Jiang et al. (2020). As a result, we use 200 out of 243 trained models for our correlation analysis. For every experiment, we use 8 NVIDIA RTX 3090 GPUs.

I Details and additional results on BNN experiments

I.1 Experimental Setting

Uncertainty calibration on image classification task We pre-train models for 200 epochs CIFAR-10/100 dataset (Krizhevsky, 2009) with ResNet-18(He et al., 2016) as mentioned in Section 2.4. We choose ensemble size M as 8 except Deep Ensemble (Lakshminarayanan et al., 2017) and Batch Ensemble (Wen et al., 2020). We use 4 ensemble members for Deep Ensemble and Batch Ensemble due to computational cost.

For evaluation, we define single member prediction as one-hot representation of network output with label smoothing. We select label smoothing coefficient as 0.01 for CIFAR-10, 0.1 for CIFAR-100. We define ensemble prediction as averaged prediction of single member predictions. For OOD detection, we use variance of prediction in output space, which is competitive to recent OOD detection methods (Ren et al., 2019; Van Amersfoort et al., 2020). We use 0.01 for σ and select best α with cross validation. For every experiment, we use 8 NVIDIA RTX 3090 GPUs.

J ADDITIONAL RESULTS ON BOUND ESTIMATION

Table 7: Results for experiments on PAC-Bayes-NTK estimation.

	CIFAR-10				CIFAR-100			
Parameter scale	0.5	1.0	2.0	4.0	0.5	1.0	2.0	4.0
$\mathrm{tr}(\Theta_X^{\theta^*})$	18746194.0	6206303.5	3335419.75	2623873.25	12688970.0	3916139.25	2819272.5	2662497.0
Bias	483.86	427.0042	299.0085	197.3149	476.9061	478.1776	440.284	329.8767
Sharpness	579.6815	472.0	402.8186	369.3761	547.2874	434.7583	398.5075	387.3265
KL divergence	531.7708	449.5021	350.9135	283.3455	512.0967	456.4679	419.3957	358.6016
Test err.	0.5617 ± 0.0670	0.4566 ± 0.0604	0.2824 ± 0.0447	0.1530 ± 0.0199	0.6210 ± 0.0096	0.6003 ± 0.0094	0.5499 ± 0.0100	0.4666 ± 0.0093
PAC-Bayes-NTK	0.7985 ± 0.0694	0.6730 ± 0.0626	0.4718 ± 0.0465	0.3186 ± 0.0202	0.8530 ± 0.0140	0.8162 ± 0.0136	0.7602 ± 0.0112	0.6617 ± 0.0114

K ADDITIONAL RESULTS ON IMAGE CLASSIFICATION

Table 8: Uncertainty calibration results on CIFAR-10 (Krizhevsky, 2009) for ResNet-18 (He et al., 2016).

	CIFAR-10			
	NLL (↓)	ECE (↓)	Brier. (↓)	AUC (↑)
Deterministic	0.3135 ± 0.0088	0.0350 ± 0.0014	0.0875 ± 0.0026	-
MCDO	0.2845 ± 0.0148	0.0157 ± 0.0012	0.1056 ± 0.0062	0.9172 ± 0.0074
MCBN	0.2922 ± 0.0074	0.0325 ± 0.0010	0.0838 ± 0.0022	0.9431 ± 0.0033
Batch Ensemble	0.2740 ± 0.0030	0.0286 ± 0.0005	0.0814 ± 0.0009	0.9376 ± 0.0036
Deep Ensemble	0.1753	0.0067	0.0594	0.8527
Linearized Laplace	0.2077 ± 0.0032	0.0180 ± 0.0009	0.0816 ± 0.0010	0.8991 ± 0.0198
Connectivity Laplace (Ours)	0.2089 ± 0.0023	0.0120 ± 0.0019	0.0720 ± 0.0011	0.9705 ± 0.0098

Table 9: Uncertainty calibration results on CIFAR-10 (Krizhevsky, 2009) for VGG-13 (Simonyan & Zisserman, 2015).

	CIFAR-10			
	NLL (↓)	ECE (↓)	Brier. (↓)	AUC (↑)
Deterministic	0.4086 ± 0.0018	0.0490 ± 0.0003	0.1147 ± 0.0005	-
MCDO	0.3889 ± 0.0049	0.0465 ± 0.0009	0.1106 ± 0.0015	0.7765 ± 0.0221
MCBN	0.3852 ± 0.0012	0.0462 ± 0.0002	0.1108 ± 0.0003	0.9051 ± 0.0065
Batch Ensemble	0.3544 ± 0.0036	0.0399 ± 0.0009	0.1064 ± 0.0012	0.9067 ± 0.0030
Deep Ensemble	0.2243	0.0121	0.0776	0.7706
Linearized Laplace	0.3366 ± 0.0013	0.0398 ± 0.0004	0.1035 ± 0.0003	0.8883 ± 0.0017
Connectivity Laplace (Ours)	0.2674 ± 0.0028	0.0234 ± 0.0011	0.0946 ± 0.0010	0.9002 ± 0.0033

Table 10: Uncertainty calibration results on CIFAR-100 (Krizhevsky, 2009) for VGG-13 (Simonyan & Zisserman, 2015).

	CIFAR-100			
	NLL (↓)	ECE (↓)	Brier. (↓)	AUC (↑)
Deterministic	1.8286 ± 0.0066	0.1544 ± 0.0010	0.4661 ± 0.0018	-
MCDO	1.7439 ± 0.0089	0.1363 ± 0.0008	0.4456 ± 0.0017	0.6424 ± 0.0099
MCBN	1.7491 ± 0.0075	0.1399 ± 0.0010	0.4488 ± 0.0015	0.7039 ± 0.0197
Batch Ensemble	1.6142 ± 0.0101	0.1077 ± 0.0020	0.4143 ± 0.0027	0.7232 ± 0.0021
Deep Ensemble	1.2006	0.0456	0.3228	0.6929
Linearized Laplace	1.5806 ± 0.0054	0.1036 ± 0.0004	0.4127 ± 0.0010	0.6893 ± 0.0221
Connectivity Laplace (Ours)	1.4073 ± 0.0039	0.0703 ± 0.0028	0.3827 ± 0.0012	0.7254 ± 0.0136