Structured Dropout Variational Inference for Bayesian Neural Networks

Anonymous Author(s)
Affiliation
Address
email

Abstract

Approximate inference in deep Bayesian networks exhibits a dilemma of how to yield high fidelity posterior approximations while maintaining computational efficiency and scalability. We tackle this challenge by introducing a novel variational structured approximation inspired by the Bayesian interpretation of Dropout regularization. Concretely, we focus on the inflexibility of the factorized structure in Dropout posterior and then propose an improved method called *Variational Structured Dropout* (VSD). VSD employs an orthogonal transformation to learn a structured representation on the variational noise and consequently induces statistical dependencies in the approximate posterior. Theoretically, VSD successfully addresses the pathologies of previous Variational Dropout methods and thus offers a standard Bayesian justification. We further show that VSD induces an adaptive regularization term with several desirable properties which contribute to better generalization. Finally, we conduct extensive experiments on standard benchmarks to demonstrate the effectiveness of VSD over state-of-the-art variational methods on predictive accuracy, uncertainty estimation, and out-of-distribution detection.

1 Introduction

Bayesian Neural Networks (BNNs) [42, 55] offer a probabilistic interpretation for deep learning models by imposing a prior distribution on the weight parameters and aiming to infer a posterior distribution instead of only point estimates. By marginalizing over this posterior for prediction, BNNs perform a procedure of ensemble learning. These principles improve the model generalization, robustness and allow for uncertainty quantification. However, computing exactly the posterior of non-linear BNNs is infeasible and approximate inference has been devised. The core challenge is how to construct an expressive approximation for the true posterior while maintaining computational efficiency and scalability, especially for modern deep learning architectures.

Variational inference is a popular deterministic approximation approach to deal with this challenge. The first practical methods were proposed in [18, 6, 32], in which the approximate posterior is assumed to be a fully factorized distribution, also called mean-field variational inference. In general, the mean-field approximation family encourages several advantages in inference including computational tractability and effective optimization with the stochastic gradient-based methods. However, it ignores the strong statistical dependencies among random weights of neural nets, leading to the inability to capture the complicated structure of the true posterior and to estimate the true model uncertainty.

To overcome this limitation, many extensive studies proposed to approximate the true posterior with richer expressiveness. For instance, [40] treats the weight matrix as a whole via a matrix variate Gaussian [20] and approximates the posterior based on this parametrization. Several later works have exploited this distribution to investigate different structured representations for the variational Gaussian posterior, such as Kronecker-factored [76, 62, 63], k-tied distribution [68], non-

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

centered or rank-1 parameterization [17, 11]. Another original idea to represent the true covariance matrix of Gaussian posterior is by employing the low-rank approximation [58, 28, 70]. For robust approximation with multimodality, [41] adopted hierarchical variational model framework [60] for inferring an implicit marginal distribution in high dimensional Bayesian setting. Despite significant improvements in both predictive accuracy and uncertainty estimation, some of these methods incur a large computational complexity and are difficult to integrate into deep convolutional networks.

Motivations. In this paper, we approach the structured posterior approximation in Bayesian neural nets from a different perspective which has been inspired by the Bayesian interpretation of Dropout training [65, 44]. More specifically, the methods proposed in [32, 16] reinterpret Dropout regularization as approximate inference in deep Bayesian models and base on this connection to learn a variational Dropout posterior over the weight parameters. From the literature, inference approaches based on Bayesian Dropout have shown competitive performances in terms of predictive accuracy on various tasks, even compared to the structured Bayesian methods aforementioned, but with much cheaper computational complexity. Moreover, with the solid and intriguing theories on effective regularization [71, 22, 73], generalization bound [45, 52], convergence rate or robust optimization [48, 47], Dropout principle offers several potentials to further improve approximate inference in deep Bayesian nets. However, as these Bayesian Dropout methods also employed simple structures of the mean-field family, their approximations often fails to obtain satisfactory uncertainty estimates [13]. In addition, the Variational Dropout methods based on multiplicative Gaussian noise also suffer from theoretical pathologies, including improper prior leading to ill-posed true posterior, and singularity of the approximate posterior making the variational objective undefined [26].

Contributions. With the above insights, we propose a novel structured variational inference framework based on Dropout principle. Our method adopts an orthogonal approximation called Householder transformation to learn a structured representation for multiplicative Gaussian noise in Variational Dropout method [32, 50]. As a consequence of the Bayesian interpretation, we go beyond the mean-field family and obtain a variational Dropout posterior with structured covariance. Furthermore, to make our approximation more expressive, we deploy a hierarchical Dropout procedure, which is equivalent to inferring a joint posterior in a hierarchical Bayesian nets. We name the proposed method as *Variational Structured Dropout* (VSD) and summarize its advantages as follows:

1. Our structured approximation is implemented on low dimensional space of variational noise with considerable computational efficiency. VSD can be employed for deep CNNs in a direct way while maintaining the backpropagation in parallel and optimizing efficiently with gradient-based methods.

2. VSD has a standard Bayesian justification, in which our method can overcome the critiques from the non-Bayesian perspective of previous Variational Dropout methods. Our inference framework uses a proper prior, non-singular approximate posterior and derives a tractable variational lower bound without further simplified approximation.

3. We investigate the inductive bias induced by the adaptive regularization of structured Dropout noise. We also provide an interpretation that VSD implicitly facilitates the networks to converge to a local minima with smaller spectral norms and stable rank. This properties suggests better generalization and we present empirical results to support this implication.

4. Finally, we carry out extensive experiments with standard datasets and network architectures to validate the effectiveness of our method on many criteria, including scalability, predictive accuracy, calibration, and out-of-distribution detection, in comparison to popular variational inference methods.

Notation. For a matrix $\mathbf{A}$, $\|\mathbf{A}\|_F$ and $\mathbf{A}^\top$ denotes the Frobenius norm and the transpose matrix, $\mathbf{A}_{i:}$ and $\mathbf{A}_{:j}$ denote the i-th row and the j-th column. For an interger i, $\mathbf{e}_i$ is the i-th standard basis, $\mathbf{1}_i \in \mathbb{R}^i$ is the vector of all ones. The diagonal matrix with diagonal entries as the elements of a vector $\mathbf{x}$ is denoted by diag($\mathbf{x}$). The inner product between two matrices $\mathbf{A}$ and $\mathbf{B}$ is denoted by $\langle \mathbf{A}, \mathbf{B} \rangle$.

2 Background

Variational inference for Bayesian neural networks: Given a dataset $\mathcal{D}$ consisting of input-output pairs $(\mathbf{X}, \mathbf{Y}) = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$. In BNNs, we impose a prior distribution over random weights $p(\mathbf{W})$ whose form is in a tractable parametric family and aim to infer the intractable true posterior $p(\mathbf{W}|\mathcal{D})$. Variational inference (VI) [25, 27] can do this by specifying a variational distribution $q_\phi(\mathbf{W})$ with free parameter ϕ and then minimizing the Kullback-Leibler (KL) divergence $\mathbb{D}_{KL}(q_\phi(\mathbf{W})\|p(\mathbf{W}|\mathcal{D}))$.

This optimization is equivalent to maximizing the Evidence Lower Bound (ELBO) with respect to variational parameters ϕ as follows:

$$\mathcal{L}(\phi) = \mathbb{E}_{q_\phi(\mathbf{W})} \log p(\mathcal{D}|\mathbf{W}) - \mathbb{D}_{KL}(q_\phi(\mathbf{W}) \| p(\mathbf{W})). \tag{1}$$

By leveraging the reparameterization trick [33] combined with the Monte Carlo integration, we can derive an unbiased differentiable estimation for the variational objective above. Then, this estimation can be effectively optimized using stochastic gradient methods with the variance reduction technique such as the *local* reparameterization trick [32].

Variational Bayesian inference with Dropout regularization: Given a deterministic neural net with the weight parameter Θ of size $K \times Q$, training this model with stochastic regularization techniques such as Dropout [24, 65] can be interpreted as approximate inference in Bayesian probabilistic models. This is because injecting a stochastic noise into the input layer is equivalent to multiplying the rows of the subsequent deterministic weight by the same random variable, namely with each datapoint $(\mathbf{x}_n, \mathbf{y}_n)$ and a noise vector ξ, we have: $\mathbf{y}_n = (\mathbf{x}_n \odot \xi)\Theta = \mathbf{x}_n \text{diag}(\xi)\Theta$. This induces a BNN with random weight matrix defined by $\mathbf{W} = \text{diag}(\xi)\Theta$. Applying VI to this Bayesian model, with some specific choices for prior and approximate posterior, the variational lower bound (1) can resemble the form of the Dropout objective in the original deterministic network. This principle is referred to as the KL condition [14]. Gal et al. [16], Kingma et al. [32] used this principle to propose Bayesian Dropout inference methods such as MC Dropout (MCD) and Variational Dropout (VD).

Dropout inference is practical approximate framework especially in high dimensional setting. However, the scope of Bayesian inference in these methods is restricted in terms of flexibility of the prior and approximate posterior. Concretely, the Dropout posteriors $q_\phi(\mathbf{W})$ in MCD and VD both have simple structures of mean-field approximation which often underestimate the variance of true posterior, possibly leading to a poor uncertainty representation [13]. Moreover, in theory, VD employed an improper log-uniform prior which can result in an ill-posed true posterior and generally push the parameters towards the penalized maximum likelihood solution [26]. Finally, VD also suffers from the singularity issue of approximate posterior that makes the KL divergence term undefined. Our work gains an efficient remedy to these pathologies.

3 Variational Structured Dropout

We focus on Bayesian Dropout methods using multiplicative Gaussian noise with correlated parameterization [32], namely $\mathbf{W} = \text{diag}(\xi)\Theta$ with the Dropout noise ξ is a multivariate Gaussian with diagonal covariance $q_\alpha(\xi) = \mathcal{N}(\mathbf{1}_K, \text{diag}(\alpha))$ and α is the droprate vector. Then, the Dropout posterior $q_\phi(\mathbf{W})$ exhibits a factorized structure on each column with the form of $q(\mathbf{W}_{:j}) = \mathcal{N}(\Theta_{:j}, \text{diag}(\alpha \odot \Theta^2_{:j}))$, whilst allowing a correlation on each row because each scalar noise ξ_i is shared across the row $\mathbf{W}_{i:}$ respectively. The parameters $\phi = (\alpha, \Theta)$ are then optimized via maximizing a variational lower bound as follows: $\mathcal{L}(\phi) := \mathbb{E}_{q_\phi(\mathbf{W})} \log p(\mathcal{D}|\mathbf{W}) - \mathbb{D}_{KL}(q_\phi(\mathbf{W}) \| p(\mathbf{W})) = \mathbb{E}_{q_\alpha(\xi)} \log p(\mathcal{D}|\xi, \Theta) - \mathbb{D}_{KL}(q_\phi(\mathbf{W}) \| p(\mathbf{W})$, where the later equation is derived from the change of variables formula.

3.1 The orthogonal approximation for variational structured noise

Intuitively, a richer representation for the noise distribution can enrich the expressiveness of Dropout posterior via the Bayesian interpretation. We implement this intuition with an assumption that the Dropout noise is sampled from a Gaussian distribution with a full covariance matrix instead of a diagonal structure, namely, $q_{\mathbf{\Sigma}}(\xi) = \mathcal{N}(\mathbf{1}_K, \mathbf{\Sigma})$ with $\mathbf{\Sigma}$ is a positive definite matrix of size $K \times K$. To make this covariance matrix learnable, we first represent $\mathbf{\Sigma}$ in the form of the spectral decomposition: $\mathbf{\Sigma} = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^T$, where $\mathbf{P}$ is an orthogonal matrix with its eigenvectors in columns, $\mathbf{\Lambda}$ is a diagonal matrix where diagonal elements are the eigenvalues. By the basis-kernel representation theorem [5, 67], we parameterize the orthogonal matrix $\mathbf{P}$ as a product of Householder matrices in the following form: $\mathbf{P} = \mathbf{H}_T \mathbf{H}_{T-1} ... \mathbf{H}_1$, where $\mathbf{H}_t = \mathbf{I} - 2\mathbf{v}_t\mathbf{v}_t^T / \|\mathbf{v}_t\|_2^2$, $\mathbf{v}_t$ is the Householder vector of size K, and T is the degree of $\mathbf{P}$. This parameterization relaxes the orthogonal constraint of matrix $\mathbf{P}$, and we can then directly optimize the covariance matrix $\mathbf{\Sigma}$ via gradient-based methods.

Notably, this transformation can be interpreted as a sequence of invertible mappings. More explicitly, we extract a zero-mean Gaussian noise $\eta^{(0)}$ from the original noise $\xi^{(0)} \sim \mathcal{N}(\mathbf{1}_K, \text{diag}(\alpha))$ in the form of $\xi^{(0)} = 1 + \eta^{(0)}$, and by successively transforming $\eta^{(0)}$ through a chain of T Householder

reflections, we obtain the induced noise and the corresponding density at each step t as follows:

$$\xi^{(t)} := 1 + \mathbf{H}_t\mathbf{H}_{t-1}...\mathbf{H}_1\eta^{(0)} = 1 + \mathbf{U}\eta^{(0)}, \quad \text{and} \quad q_t(\xi) := \mathcal{N}(\mathbf{1}_K, \mathbf{U}\text{diag}(\alpha)\mathbf{U}^T).$$

By injecting the structured noise $\xi^{(t)}$ into the deterministic weight Θ, we obtain a random weight $\mathbf{W}^{(t)} = \text{diag}(\xi^{(t)})\Theta$ and a structured Dropout posterior $q_t(.)$ with *fully correlated* representation, in which the marginal column distribution is given by: $q_t(\mathbf{W}_{:j}) = \mathcal{N}(\Theta_{:j}, \text{diag}(\Theta_{:j})\mathbf{U}\text{diag}(\alpha)\mathbf{U}^T\text{diag}(\Theta_{:j}))$. Detailed discussion about the expressiveness of this correlated structure is presented in Appendix A.1. With the above derivations, we use variational inference with approximate posterior $q_t(\mathbf{W})$ and optimize the variational lower bound as follows:

$$\begin{aligned}\mathcal{L}(\phi) &:= \mathbb{E}_{q_t(\mathbf{W})} \log p(\mathcal{D}|\mathbf{W}) - \mathbb{D}_{KL}(q_t(\mathbf{W})||p(\mathbf{W})) \\ &= \mathbb{E}_{q_\alpha(\xi)} \log p(\mathcal{D}|\Theta, \xi^{(t)}) - \mathbb{D}_{KL}(q_t(\mathbf{W})||p(\mathbf{W})). \end{aligned} \tag{2}$$

Overcoming the singularity issue of approximate posterior. In Variational Dropout with correlated parameterization, there is a mismatch in support between the approximate posterior and the prior, thus making the KL term $\mathbb{D}_{KL}(q(\mathbf{W})||p(\mathbf{W}))$ undefined [26]. Specifically, the form $\mathbf{W} = \text{diag}(\xi)\Theta$ is equivalent to multiplying each row $\mathbf{W}_{i:}$ by the same scalar noise ξ_i, namely $\mathbf{W}_{i:} = \xi_i\Theta_{i:}$ with $q(\xi_i) = \mathcal{N}(1, \alpha_i)$. This means that the approximate distribution always assigns all its mass on subspaces defined by the directions aligned with the rows of Θ. These subspaces have Lebesgue measure zero causing the singularities in approximate posterior and the KL term will be undefined whenever the prior $p(\mathbf{W})$ puts zero mass to these subspaces. However, in VSD, the scalar noises are not treated separately due to the structured correlation. In other words, VSD injects each ξ_i into the whole matrix Θ instead of some individual directions. Indeed, we have: $\mathbf{W}^{(VD)} = \text{diag}(\xi^{(0)})\Theta = \Theta + \text{diag}(\eta^{(0)})\Theta = \Theta + \sum_{i=1}^K \eta_i^{(0)}(\text{diag}(\mathbf{e}_i)\Theta) = \Theta + \sum_{i=1}^K \eta_i^{(0)}\Theta_{(i)}$, and $\mathbf{W}^{(VSD)} = \text{diag}(\xi^{(t)})\Theta = \Theta + \text{diag}(\mathbf{U}\eta^{(0)})\Theta = \Theta + \sum_{i=1}^K \eta_i^{(0)}(\text{diag}(\mathbf{U}_{i:})\Theta)$, where $\Theta_{(i)}$ is the matrix Θ with only the i-th row retained. While VD causes *singular* components represented by $\{\Theta_{(i)}\}_{i=1}^K$, VSD maintains a trainable orthogonal matrix $\mathbf{U}$ which prevents the approximate posterior from having degenerate supports with measure zero, thereby avoiding the singularity issue. In the next section, we will make the KL term well-defined with an appropriate prior.

3.2 Derivation of tractable variational lower bound

We consider employing an isotropic Gaussian as the prior distribution, namely $p(\mathbf{W}) = \prod_{j=1}^Q p(\mathbf{W}_{:j})$ with $p(\mathbf{W}_{:j}) = \mathcal{N}(0, \text{diag}(\beta_{:j}^{-1}))$ and β is a hyper-parameter matrix of the same size with $\mathbf{W}$. With the analysis in previous section, our approximate posterior $q_t(\mathbf{W})$ is absolutely continuous w.r.t the prior $p(\mathbf{W})$, and thus the KL term $\mathbb{D}_{KL}(q_t(\mathbf{W})||p(\mathbf{W}))$ is defined. Furthermore, the Gaussian prior helps our proposal avoid the pathologies of improper true posterior and ill-posed inference in VD.

Note that, the prior $p(\mathbf{W})$ is a fully factorized Gaussian which usually facilitates simple analysis and efficient computation. This factorized structure is chosen also because we have no reason for the correlation between non-identical weights at first. Moreover, several arguments indicate that a simple prior over parameter $p(\mathbf{W})$, when interacts with neural nets architecture $f(\mathcal{D}; \mathbf{W})$, induces a sophisticated prior over function $p(f(\mathcal{D}; \mathbf{W}))$, with desirable properties and useful inductive biases [74]. However, when we are interested in structured approximations in parameter space, the factorized prior may raise some contradictions. By relative entropy decomposition, we have:

$$\mathbb{D}_{KL}(q_t(\mathbf{W})||p(\mathbf{W})) = \sum_{j=1}^Q \mathbb{D}_{KL}(q_t(\mathbf{W}_{:j})||p(\mathbf{W}_{:j})) + \mathbf{I}(\mathbf{W}_{:1}, \mathbf{W}_{:2}, ..., \mathbf{W}_{:Q}), \tag{3}$$

where $\mathbf{I}(.)$ is the mutual information which is completely defined in VSD. Maximizing the variational lower bound tends to encourage smaller KL term, and hence constrains the components in RHS of equation (3). Intuitively, a relatively small mutual information can break the strong correlations between the columns of $\mathbf{W}$. Several studies have focused on this limitation and suggested using richer priors such as matrix variate Gaussian [66, 77], doubly semi-implicit distribution [51]. Our solution for this scenario is derived naturally from equation (3), in which we leverage the mutual information as an additional regularization term. Concretely, we maximize an alternative variational objective as follows:

$$\begin{aligned}\mathcal{L}_{MI}(\phi) &:= \mathbb{E}_{q_\alpha(\xi)} \log p(\mathcal{D}|\Theta, \xi^{(t)}) - \mathbb{D}_{KL}(q_t(\mathbf{W})||p(\mathbf{W})) + \mathbf{I}(\mathbf{W}_{:1}, \mathbf{W}_{:2}, ..., \mathbf{W}_{:Q}) \\ &= \mathbb{E}_{q_\alpha(\xi)} \log p(\mathcal{D}|\Theta, \xi^{(t)}) - \mathbb{D}_{KL}(q_t^\star(\mathbf{W})||p(\mathbf{W})) \end{aligned} \tag{4}$$

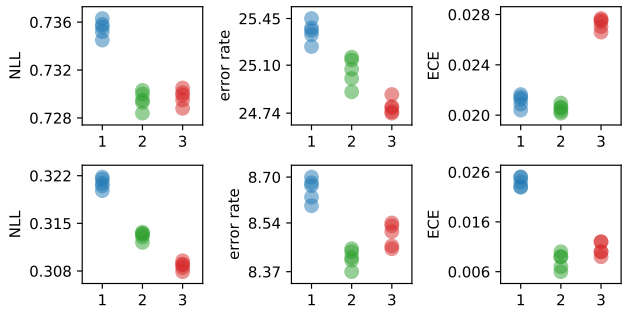

Figure 1: The performance of VSD when using the number of transformations $T \in \{1, 2, 3\}$. Evaluation over 5 runs on CIFAR10 (above) and SVHN (below) with LeNet architecture.

where $q_t^\star(\mathbf{W}) = \prod_{j=1}^{Q} q_t(\mathbf{W}_{:j})$ is the product of marginal column distributions. From the information-theoretic perspective, augmenting the mutual information is a standard principle for structure learning in Bayesian networks [34]. Interestingly, this technique is utilized reasonably in our method. This is because our dependence structure allows to specify explicitly the marginal distribution on each column of $\mathbf{W}$, leading to a tractable objective in equation (4) with the KL divergence between two multivariate Gaussian that can be calculated analytically in closed-form. We note that a similar application to other structured approximations, such as low-rank, Kronecker-factored or matrix variate Gaussian, can be non-trivial.

The KL condition in VSD. With the new variational objective in equation (4), to benefit VSD from the Bayesian statistics, we need to ensure $\mathbb{D}_{KL}(q_t^\star(\mathbf{W})||p(\mathbf{W}))$ satisfies the KL condition. We solve this prerequisite by determining the precision parameter β the prior $p(\mathbf{W})$ via the Empirical Bayes approach. Then, the KL term has the form independent of the weight parameter Θ as follows:

$$\mathbb{D}_{KL}(q_t^\star(\mathbf{W})||p(\mathbf{W})) = \frac{Q}{2} \sum_{i=1}^{K} \log\big((1 + \sum_{j=1}^{K} \alpha_j U_{ij}^2)/\alpha_i\big). \tag{5}$$

A formal proof of equation (5) is in Appendix B. We further improve the efficiency of Householder transformation in VSD by utilizing the idea from [69, 4] where this technique was deployed to approximate a full covariance Gaussian posterior on latent space of VAEs model [33]. In particular, they made the approximation more flexible by using fully connected layers between the Householder vectors. The low dimension of variational noise of VSD leads to a good adaptation, but with a little trade-off in computational complexity when increasing the number of transformation steps T. We show the performance of VSD when using this neural approximate technique with $T \in \{1, 2, 3\}$ in Figure 1, where a larger T could potentially improve the results on predictive measures. However, to maintain computational efficiency, we recommend using $T = 1$ or 2 in large-scale experiments.

3.3 Joint inference with hierarchical prior

We further promote our proposal by introducing a prior hierarchy in VSD framework and then obtain a joint approximation for the Dropout posterior. This will facilitate the scalability of Bayesian inference in our method in terms of the expressiveness of both prior distribution and approximate posterior. We design a two-level hierarchical prior given by: $p(\mathbf{W}, \mathbf{z}) = p(\mathbf{W}|\mathbf{z}, \beta)p(\mathbf{z})$, where $p(\mathbf{W}|\mathbf{z}, \beta) = \prod_{j=1}^{Q} p(\mathbf{W}_{:j}|\mathbf{z}, \beta_{:j})$ and $p(\mathbf{W}_{:j}|\mathbf{z}, \beta_{:j}) = \mathcal{N}(0, \text{diag}(\mathbf{z} \odot \beta_{:j}^{-1}))$; the hyperprior $p(\mathbf{z})$ is a distribution with positive support such as Gamma or half-Cauchy distribution; the latent $\mathbf{z}$ has the size of the number of rows and is shared across columns of the weight matrix $\mathbf{W}$; the hyper-parameter matrix β is treated as a scaling factor. This prior family has a center parameterization and induces several compelling properties such as facilitating feature sparsity [39, 9], model selection [17] or improving robustness, uncertainty calibration [11].

We implement variational inference with a joint approximate posterior, also referred to as the joint Dropout posterior, which is parameterized as follows:

$$q_\phi(\mathbf{W}, \mathbf{z}) = q_\psi(\mathbf{z}) q_\phi(\mathbf{W}|\mathbf{z}) \quad \text{with} \quad q_\phi(\mathbf{W}_{:j}|\mathbf{z}) = \mathcal{N}(\mathbf{z} \odot \Theta_{:j}, \text{diag}(\mathbf{z}^2 \odot \alpha \odot \Theta_{:j}^2)),$$

where $q_\phi(\mathbf{W}|\mathbf{z})$ is the conditional Dropout posterior, $q_\psi(\mathbf{z})$ is chosen depending on the family of prior $p(z)$ so that the reparametrization trick can be utilized. Sampling the random weight $\mathbf{W}$ from

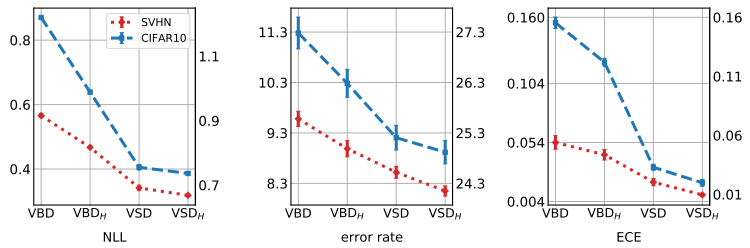

Figure 2: The performance of VBD and VSD when using the prior hierarchy, with the labels are VBD$_H$ and VSD$_H$ respectively.

the joint variational posterior $q_\phi(\mathbf{W}, \mathbf{z})$ includes two steps: $\mathbf{z}^* \sim q_\psi(\mathbf{z})$ and $\mathbf{W}^* \sim q_\phi(\mathbf{W}|\mathbf{z}^*)$, in which the second one can be reparameterized as: $\mathbf{W}^* = \text{diag}(\mathbf{z}^*)\text{diag}(\xi)\Theta = \text{diag}(\mathbf{z}^* \odot \xi)\Theta$, with the noise $\xi \sim \mathcal{N}(\mathbf{1}_K, \text{diag}(\alpha))$. This new representation adapts to the vanilla Dropout procedure but allows our method to regularize each unit layer with different levels of stochasticity. We derive some insights about the role of hierarchical prior in our framework in Appendix A.2. We apply the Householder transformation to the variational noise ξ and obtain a new joint approximate posterior:

$$q_t(\mathbf{W}, \mathbf{z}) = q_\psi(\mathbf{z}) q_t(\mathbf{W}|\mathbf{z}) \quad \text{with} \quad q_t(\mathbf{W}_{:j}|\mathbf{z}) = \mathcal{N}(\mathbf{z} \odot \Theta_{:j}, \mathbf{V}_j \mathbf{U} \text{diag}(\alpha)(\mathbf{V}_j \mathbf{U})^T),$$

where $\mathbf{V}_j = \text{diag}(\mathbf{z} \odot \Theta_{:j})$. Similarly, we define $q_t^\star(\mathbf{W}|\mathbf{z}) = \prod_{j=1}^{Q} q_t(\mathbf{W}_{:j}|\mathbf{z})$ and then optimize an alternate variational objective given by:

$$\begin{aligned}\mathcal{L}(\phi, \psi) :&= \mathbb{E}_{q_\psi(\mathbf{z}) q_t(\mathbf{W}|\mathbf{z})} \log p(\mathcal{D}|\mathbf{W}) - \\ &\mathbb{E}_{q_\psi(\mathbf{z})}(\mathbb{D}_{KL}(q_t^\star(\mathbf{W}|\mathbf{z}, \phi)||p(\mathbf{W}|\mathbf{z}, \beta)) - \mathbb{D}_{KL}(q_\psi(\mathbf{z})||p(\mathbf{z})).\end{aligned}$$

A full derivation of $\mathcal{L}(\phi, \psi)$ including the KL condition is given in Appendix C. The parameterization of the prior hierarchy in our method is flexible without any simplifying assumptions about hyperprior $p(\mathbf{z})$. We can directly apply it for Variational Bayesian Dropout framework (VBD) [29] (a derivation is in Appendix F). As the experimental results are shown in Figure 2, the hierarchical prior significantly improves the performance of both VBD and VSD on predictive metrics. Therefore, we aim to introduce VSD with hierarchical prior as a unified framework of our proposal in this paper.

3.4 Scalability of Variational Structured Dropout

Approximating a structured posterior directly on the random weights of deep convolution models is challenging. Besides expensive computation, it is difficult to employ the local reparameterization trick [32], leading to the high variance issue in training. We apply VSD to convolutional layer by learning a structured noise with the size of the number of kernels and imposing it to convolutional weights: $\xi \sim \mathcal{N}(\mathbf{1}_K, \mathbf{U}\text{diag}(\alpha)\mathbf{U}^T)$ and $\mathbf{W}_{ijk} = \xi_k \Theta_{ijk}$, with i, j, k are the indexes representing height, width, and kernel respectively. This simple solution greatly reduces computational complexity while being able to captures the dependencies among kernels of the convolutional layer.

We present the complexity of MAP and VI methods in terms of computational cost and memory storage in Table 4 in Appendix D with detailed analysis. VSD adopts the advantage of Dropout training and maintains an efficiency on both criteria, We also give more results in Table 5 about the empirical computation time of VSD and some other methods. Based on these tables, VSD shows more effective running time than the mean-field BBNs (BBB-LTR). Although there is a trade-off when using a larger number of T, VSD does not incur much extra computation time compared to VD.

3.5 On explicit regularization of Variational Structured Dropout

There are compelling theories to explain the tremendous success of Dropout, of which regularization-based is one of the most active approaches [71, 22, 48, 46, 73, 7]. To characterize the regularization of VSD, we consider a deep linear neural net with L layers parameterized by $\{\Theta^{(i)}\}_{i=1}^L$, and define some notations as: $\mathbf{x}$ is an input data, $\mathcal{B}$ is the data batch; $\mathbf{h}_i$ is the i-th hidden layer; $\mathbf{J}_i(\mathbf{x})$ denotes the Jacobian of network output w.r.t $\mathbf{h}_i(\mathbf{x})$; $\mathbf{H}_i(\mathbf{x})$ and $\mathbf{H}_{\text{out}}(\mathbf{x})$ denotes the Hessian of the loss w.r.t $\mathbf{h}_i(\mathbf{x})$ and the network output, respectively. Then we have $\mathbf{J}_i = (\prod_{l=i}^{L} \Theta^{(l)})^T \triangleq \Theta^{[i:L]}$ the transposition of linear multiplication of weight matrices from i-th layer to the last one. From a detailed derivation

Table 1: Results for VSD and baselines on vectorized MNIST, CIFAR10 and SVHN. Results are averaged over 5 random seeds. For all metrics, lower is better.

Method	MNIST						CIFAR10			SVHN		
	FC 400x2			FC 750x3			CNN 32x64x128					
	NLL	err. rate	ECE	NLL	err. rate	ECE	NLL	err. rate	ECE	NLL	err. rate	ECE
MAP	0.098	1.32	0.011	0.109	1.27	0.011	2.847	34.04	0.272	0.855	12.26	0.086
BBB	0.109	1.59	0.011	0.140	1.50	0.013	1.202	30.11	0.098	0.545	10.57	0.017
MCD	0.049	1.26	0.007	0.057	1.22	0.007	0.794	26.91	0.024	0.365	9.23	0.013
VD	0.051	1.21	0.007	0.061	1.17	0.008	1.176	27.45	0.156	0.534	9.47	0.055
ELRG	0.053	1.54	-	-	-	-	0.871	29.43	-	-	-	-
VSD	**0.042**	**1.08**	**0.006**	**0.048**	**1.09**	**0.006**	**0.730**	**24.92**	**0.020**	**0.299**	**8.39**	**0.008**
D.E	0.057	1.29	0.009	0.063	1.21	0.009	1.815	26.44	0.042	0.783	9.31	0.070
SWAG	0.044	1.27	0.008	**0.043**	1.25	0.007	0.799	26.94	**0.012**	0.312	8.42	0.021

presented in Appendix E, VSD induces an explicit regularization given by:

$$R_{VSD} = \mathbb{E}_{(\mathbf{x}\sim\mathcal{B})} \sum_{i=1}^{L} \left\langle \mathbf{H}_i, \text{diag}(\mathbf{h}_i)\mathbf{U}\text{diag}(\alpha)\mathbf{U}^T\text{diag}(\mathbf{h}_i) \right\rangle .$$

Note that, $\mathbf{H}_i$ can be approximated by $\mathbf{J}_i^T\mathbf{H}_{\text{out}}\mathbf{J}_i$ after ignoring the non-PSD term which is less important empirically [64]. This regularizer offers some intriguing but popular interpretations related to the flatness of local optima or the curvature of loss landscape [73, 7] (see a detailed explanation in Appendix E). We now show novel properties of VSD induced by the orthogonal matrix $\mathbf{U}$.

VSD imposes a Tikhonov-like regularization and reshapes the gradient of network weights: We rewrite our regularization corresponding to layer i-th by: $R_{VSD}^{(i)} = \mathbb{E}_{(\mathbf{x}\sim\mathcal{B})}\|\mathbf{H}_i^{1/2}\text{diag}(\mathbf{h}_i)\mathbf{U}\text{diag}(\alpha^{1/2})\|_F^2$. This form can be interpreted as the Tikhonov-*like* regularization imposed on the square root of Hessian matrix $\mathbf{H}_i$, in which the Tikhonov matrix $\Gamma := \text{diag}(\mathbf{h}_i)\mathbf{U}\text{diag}(\alpha^{1/2})$ is automatically learned during training. This principle can improve the conditioning of the estimation problem. Furthermore, when considering the case of regression problem, we have:

$$R_{VSD}^{(i)} = \mathbb{E}_{(\mathbf{x}\sim\mathcal{B})} \left[\Theta^{[i:L]}\text{diag}(\mathbf{h}_i)\mathbf{U}\text{diag}(\alpha)\mathbf{U}^T\text{diag}(\mathbf{h}_i)\Theta^{[i:L].T} \right] . \tag{6}$$

This is a data-dependent regularization with adaptive structure determined by the matrix $\Gamma\Gamma^T = \text{diag}(\mathbf{h}_i)\mathbf{U}\text{diag}(\alpha)\mathbf{U}^T\text{diag}(\mathbf{h}_i)$. From the algorithmic perspective, this regularizer allows VSD to reshape the gradient of network weights according to the geometry of the data based on both scale and direction information [10, 21]. Meanwhile $\Gamma\Gamma^T = \text{diag}(\mathbf{h}_i)\text{diag}(\alpha)\text{diag}(\mathbf{h}_i)$ only plays as a scaling factor in VD.

VSD penalizes implicitly the spectral norm of weight matrices: Let $\Omega_i := \text{diag}(\mathbf{h}_i)\mathbf{J}_i^T\mathbf{H}_{\text{out}}\mathbf{J}_i\text{diag}(\mathbf{h}_i)$, then our regularizer can be rewritten as: $R_{VSD}^{(i)} = \mathbb{E}_{(\mathbf{x}\sim\mathcal{B})}\sum_{k=1}^{K}\alpha_k^2\mathbf{U}_{:k}^T\Omega_i\mathbf{U}_{:k}$. Since the trainable matrix $\mathbf{U}$ satisfies $\mathbf{U}_{:k}^T\mathbf{U}_{:k} = 1$ for any k, a penalty on $\mathbf{U}_{:k}^T\Omega_i\mathbf{U}_{:k}$ implies that VSD likely prefers a solution with smaller spectral norms of the matrix $\mathbf{H}_{\text{out}}^{1/2}\mathbf{J}_i\text{diag}(\mathbf{h}_i)$ and thus of the network weights. This implication points us to the well-studied theories about generalization bound based on the spectral norm [3, 57]. Concretely, Neyshabur et al. [57] suggests that smaller spectral norm and stable rank can lead to better generalization. This expectation can be observed empirically in VSD through Table 7 in Appendix E. A more solid investigation about the generalization of VSD is of interest.

4 Experiments

In this section, we provide experimental evaluations to show the effectiveness of our proposed methods compared with the existing methods in terms of both predictability and scalability. We focus mainly on variational inference methods of the following two directions: the first one is direct approximations of the posterior on the random weights of Bayesian nets, including Bayes by Backprop (BBB) [6], Variational Matrix Gaussian (VMG) [40], low-rank approximations (SLANG, ELRG) [49, 70]; and the other one is the Bayesian Dropout methods with MC Dropout (MCD) [15, 16], Variational Dropout (VD) [32], and our method-Variational Structured Dropout (VSD). In addition, we evaluate

Table 2: Image classification using AlexNet architecture. Results are averaged over 5 random seeds.

AlexNet	**CIFAR10**			**CIFAR100**			**SVHN**			**STL10**		
	NLL	ACC	ECE	NLL	ACC	ECE	NLL	ACC	ECE	NLL	ACC	ECE
MAP	1.038	69.58	0.121	4.705	40.23	0.393	0.418	87.56	0.033	2.532	65.70	0.267
BBB	0.994	65.38	0.062	2.659	32.41	**0.049**	0.476	87.30	0.094	1.707	65.46	0.222
MCD	0.717	75.22	0.023	2.503	42.91	0.151	0.401	88.03	0.023	1.059	63.65	**0.052**
VD	0.702	77.28	**0.028**	2.582	43.10	0.106	0.327	90.76	0.010	2.130	65.48	0.195
ELRG	0.723	76.87	0.065	2.368	42.90	0.099	0.312	90.66	**0.006**	1.088	59.99	**0.018**
VSD	**0.656**	**78.21**	**0.046**	**2.241**	**46.85**	0.112	**0.290**	**91.62**	**0.008**	**1.019**	**67.98**	0.079
D.E	0.872	77.56	0.115	3.402	46.42	0.314	0.319	90.30	**0.008**	2.229	**68.51**	0.241
SWAG	**0.651**	**78.14**	0.059	**1.958**	**49.81**	**0.028**	0.331	90.04	0.031	1.522	**68.41**	0.161

Table 3: Image classification using ResNet18 architecture. Results are averaged over 5 random seeds.

ResNet18	**CIFAR10**			**CIFAR100**			**SVHN**			**STL10**		
	NLL	ACC	ECE	NLL	ACC	ECE	NLL	ACC	ECE	NLL	ACC	ECE
MAP	0.644	86.34	0.093	2.410	55.38	0.243	0.232	95.32	0.028	1.401	71.26	0.199
BBB	0.697	76.63	0.071	2.239	41.07	0.100	0.218	94.53	0.047	1.290	71.55	0.179
MCD	0.534	87.47	0.084	2.121	59.28	0.227	0.207	95.78	0.026	1.333	72.28	0.188
VD	0.451	87.68	**0.024**	2.888	56.80	0.284	0.164	96.11	0.017	1.084	73.29	0.084
ELRG	**0.382**	87.24	**0.018**	1.634	58.14	**0.096**	0.145	96.03	**0.003**	0.811	73.66	**0.080**
VSD	0.464	87.44	0.061	**1.504**	**60.15**	0.116	**0.140**	**96.41**	**0.003**	**0.769**	**74.50**	**0.083**
D.E	0.488	**88.91**	0.069	1.913	**60.16**	0.203	0.171	**96.36**	0.020	1.197	73.16	0.177
SWAG	**0.330**	**88.77**	0.026	**1.417**	**62.45**	**0.028**	**0.130**	**96.72**	0.016	0.843	73.15	**0.069**

the performance of point estimate framework MAP and two standard non-variational baselines Deep Ensemble (D.E) [36] and SWAG [43]. Details about data descriptions, network architectures, hyper-parameter tuning are presented in Appendix I.

4.1 Image classification

We now compare the predictive performance of the aforementioned methods for classification tasks on three standard image datasets: MNIST [37], CIFAR10 [35], and SVHN [56]. We evaluate the predictive probabilities using negative log-likelihood (NLL), error rate, and expected calibration error (ECE) [54, 19]. Details on experimental settings are available in Appendix I.4.

The synthesis results of this experiment are in Table 1. On MNIST, VMG achieves err. rates of 1.17% and 1.27% with FC 400x2 and FC 750x3 respectively, while SLANG reports 1.72% err. rate with FC 400x2. In general, VSD outperforms consistently other variational methods in most settings. For D.E and SWAG, VSD exhibits competitive results on all three metrics. Especially, the figures on NLL and ECE indicate well-calibrated probabilities in our model. This is also a common but noteworthy behavior in structured approximations. On the other hand, for the remaining methods such as MAP and BBB, the error rates are worse by a large margin compared to VSD (respectively about 9% and 5% on CIFAR10, 4% and 2% on SVHN). On CIFAR10 and SVHN, these two methods and VD all show poor results on both NLL and ECE measures, implying that it will be difficult for them to reason properly about the model uncertainty especially in the out-of-distribution context. For MC Dropout, we observe a pretty good performance with the second-best result in variational methods that is similar to those reported of other works [49, 59, 70]. These results of the Bayesian Dropout methods are competitive with structured methods such as VMG, SLANG, ELRG. This further reinforces our motivation about the potential of Dropout methods for improving the predictive performance.

4.2 Scaling up Bayesian deep convolutional networks

We conduct additional experiments to integrate VSD into large-scale convolutional networks. We reproduce the experiments proposed in [70] (ELRG), in which we trained AlexNet and ResNet18 on 4 datasets CIFAR10, SVHN, CIFAR100 [35], and STIL10 [8] to evaluate the predictive performance of our proposal compared to other methods.

The final results are given in Table 2 and Table 3, in which the top two results will be highlighted in bold. For AlexNet, the performance of VSD is more consistent and higher than other variational methods. Modern deeper architectures facilitate deterministic estimates like MAP to better learn

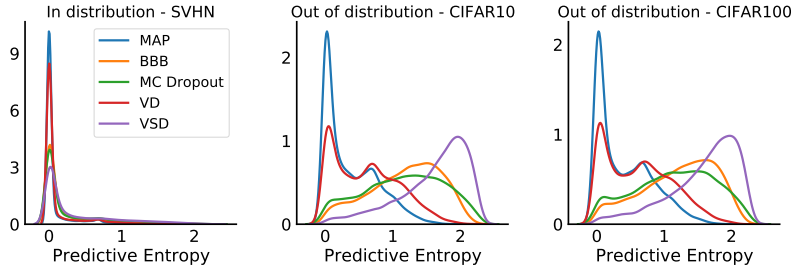

Figure 3: Histograms of predictive entropy for LeNet architecture trained on SVHN dataset.

discriminative information extracted from training data but also makes its predictions more confident when picking excessively on unique optima. Therefore, although MAP has comparable predictive accuracy on some settings, it comes at a trade-off with the worst results on both NLL and ECE. Meanwhile, ELRG with a low-rank structure on the variational posterior gains desirable properties on uncertainty metrics. Its performance on NLL and ECE are competitive to that of VSD, however, our method obtains significant improvements on accuracy metric. For the remaining methods, BBB performs poorly on CIFAR10 and CIFAR100 in predictive accuracy, but it still has better performance than MAP in terms of NLL and ECE.

For ResNet18 architecture, while MAP and BBB still exhibit the same behavior as mentioned above, VSD continues to achieve the convincing results, namely, it has the best performance on CIFAR100 and SVHN over all three metrics when compared to variational-based methods. On CIFAR10 and STL10, VSD also gets competitive statistics on NLL and accuracy. Overall, compared with MCD, VD, and ELRG, our method maintains good performance with better stability.

4.3 Predictive entropy performance

We now evaluate the predictive uncertainty of each model on out-of-distribution settings that have been implemented in previous works [36, 41, 59, 70]. We evaluate the entropy of predictive distribution $p(y^*|x^*, \mathcal{D})$ and use the density of this entropy to assess the quality of uncertainty estimates.

For LeNet, we train the model on SVHN dataset and then consider out-of-distribution data from CIFAR10 and CIFAR100. The results are shown in Figure 3. All methods work well on in-distribution data SVHN with the entropy value being distributed mostly around zero. However, the entropy densities of MAP and VD are concentrated excessively. This indicates that these methods tend to make overconfident predictions on out-of-distribution data. This claim is further supported by the quantitative results on CIFAR10 and CIFAR100 datasets. In contrast, MCD, BBB, and VSD are well-calibrated with a moderate level of confidence for in-distribution data. On CIFAR10 and CIFAR100 datasets, VSD gains better results with entropy values being distributed over a larger support, meaning that the predictions of VSD are closer to uniform on unseen classes.

We run a similar experiment, in which we train AlexNet on CIFAR10 and use SVHN, CIFAR100 as out-of-distribution data. The results are shown in the top row of Figure 4 in Appendix G.1. While MAP and VD still exhibit the same overconfident phenomenon as on LeNet, we observe the underconfident predictions of BBB and MC Dropout even on in-distribution data, which possibly leads to a high uncertainty on out-of-distribution data. We hypothesize that this is because the models trained with these methods are likely underfit with a low accuracy on the in-distribution training data. In contrast, VSD estimates reasonably the predictive entropy in both settings. The remaining scenario with ResNet18 trained on CIFAR100 is shown in the bottom row of Figure 4 with the same behaviors.

5 Conclusions

We proposed a novel approximate inference framework for deep Bayesian nets, named Variational Structured Dropout (VSD). In VSD, we learn a structured approximate posterior via the Dropout principle. VSD is able to yield a flexible inference while maintaining computational efficiency and scalability for deep convolutional models. The extensive experiments have evidenced the advantages of VSD such as well-calibrated prediction, better generalization, good uncertainty estimation. Given a consistent performance of VSD as presented throughout the paper, an extension of that method to other problems, such as Bayesian active learning or reinforcement learning, is of interest.

References

[1] O. Arenz, M. Zhong, and G. Neumann. Trust-region variational inference with Gaussian mixture models. *Journal of Machine Learning Research*, 21(163):1–60, 2020.

[2] A. Asuncion and D. Newman. UCI machine learning repository, 2007.

[3] P. Bartlett, D. J. Foster, and M. Telgarsky. Spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1706.08498*, 2017.

[4] R. v. d. Berg, L. Hasenclever, J. M. Tomczak, and M. Welling. Sylvester normalizing flows for variational inference. *arXiv preprint arXiv:1803.05649*, 2018.

[5] C. H. Bischof and X. Sun. On orthogonal block elimination. *Preprint MCS-P450-0794, Mathematics and Computer Science Division, Argonne National Laboratory*, 1994.

[6] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424*, 2015.

[7] A. Camuto, M. Willetts, U. Şimşekli, S. Roberts, and C. Holmes. Explicit regularisation in Gaussian noise injections. *arXiv preprint arXiv:2007.07368*, 2020.

[8] A. Coates, A. Ng, and H. Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

[9] T. Cui, A. Havulinna, P. Marttinen, and S. Kaski. Informative Gaussian scale mixture priors for Bayesian neural networks. *arXiv preprint arXiv:2002.10243*, 2020.

[10] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

[11] M. W. Dusenberry, G. Jerfel, Y. Wen, Y.-a. Ma, J. Snoek, K. Heller, B. Lakshminarayanan, and D. Tran. Efficient and scalable Bayesian neural nets with rank-1 factors. *arXiv preprint arXiv:2005.07186*, 2020.

[12] B. Efron. *Large-scale inference: empirical Bayes methods for estimation, testing, and prediction*, volume 1. Cambridge University Press, 2012.

[13] A. Y. Foong, D. R. Burt, Y. Li, and R. E. Turner. On the expressiveness of approximate inference in Bayesian neural networks. *arXiv preprint arXiv:1909.00719*, 2019.

[14] Y. Gal. Uncertainty in deep learning. *University of Cambridge*, 1(3), 2016.

[15] Y. Gal and Z. Ghahramani. Bayesian convolutional neural networks with bernoulli approximate variational inference. *arXiv preprint arXiv:1506.02158*, 2015.

[16] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016.

[17] S. Ghosh, J. Yao, and F. Doshi-Velez. Structured variational learning of Bayesian neural networks with horseshoe priors. *arXiv preprint arXiv:1806.05975*, 2018.

[18] A. Graves. Practical variational inference for neural networks. In *Advances in neural information processing systems*, pages 2348–2356, 2011.

[19] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

[20] A. K. Gupta and D. K. Nagar. *Matrix variate distributions*, volume 104. CRC Press, 2018.

[21] V. Gupta, T. Koren, and Y. Singer. A unified approach to adaptive regularization in online and stochastic optimization. *arXiv preprint arXiv:1706.06569*, 2017.

[22] D. P. Helmbold and P. M. Long. On the inductive bias of dropout. *The Journal of Machine Learning Research*, 16(1):3403–3454, 2015.

[23] J. M. Hernández-Lobato and R. Adams. Probabilistic backpropagation for scalable learning of Bayesian neural networks. In *International Conference on Machine Learning*, pages 1861–1869, 2015.

[24] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[25] G. E. Hinton and D. Van Camp. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the sixth annual conference on Computational learning theory*, pages 5–13, 1993.

[26] J. Hron, A. G. d. G. Matthews, and Z. Ghahramani. Variational Bayesian dropout: pitfalls and fixes. *arXiv preprint arXiv:1807.01969*, 2018.

[27] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[28] M. E. Khan, D. Nielsen, V. Tangkaratt, W. Lin, Y. Gal, and A. Srivastava. Fast and scalable Bayesian deep learning by weight-perturbation in adam. *arXiv preprint arXiv:1806.04854*, 2018.

[29] V. Kharitonov, D. Molchanov, and D. Vetrov. Variational dropout via empirical bayes. *arXiv preprint arXiv:1811.00596*, 2018.

[30] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[31] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling. Improving variational inference with inverse autoregressive flow. *arXiv preprint arXiv:1606.04934*, 2016.

[32] D. P. Kingma, T. Salimans, and M. Welling. Variational dropout and the local reparameterization trick. In *Advances in neural information processing systems*, pages 2575–2583, 2015.

[33] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[34] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[35] A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[36] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *arXiv preprint arXiv:1612.01474*, 2016.

[37] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[38] Y. Liu, W. Dong, L. Zhang, D. Gong, and Q. Shi. Variational Bayesian dropout with a hierarchical prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7124–7133, 2019.

[39] C. Louizos, K. Ullrich, and M. Welling. Bayesian compression for deep learning. *arXiv preprint arXiv:1705.08665*, 2017.

[40] C. Louizos and M. Welling. Structured and efficient variational deep learning with matrix Gaussian posteriors. In *International Conference on Machine Learning*, pages 1708–1716, 2016.

[41] C. Louizos and M. Welling. Multiplicative normalizing flows for variational Bayesian neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2218–2227. JMLR. org, 2017.

[42] D. J. MacKay. A practical Bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992.

[43] W. J. Maddox, P. Izmailov, T. Garipov, D. P. Vetrov, and A. G. Wilson. A simple baseline for Bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 32:13153–13164, 2019.

[44] S.-i. Maeda. A Bayesian encourages dropout. *arXiv preprint arXiv:1412.7003*, 2014.

[45] D. McAllester. A pac-Bayesian tutorial with a dropout bound. *arXiv preprint arXiv:1307.2118*, 2013.

[46] P. Mianjy and R. Arora. On dropout and nuclear norm regularization. In *International Conference on Machine Learning*, pages 4575–4584. PMLR, 2019.

[47] P. Mianjy and R. Arora. On convergence and generalization of dropout training. *Advances in Neural Information Processing Systems*, 33, 2020.

[48] P. Mianjy, R. Arora, and R. Vidal. On the implicit bias of dropout. In *International Conference on Machine Learning*, pages 3537–3545, 2018.

[49] A. Mishkin, F. Kunstner, D. Nielsen, M. Schmidt, and M. E. Khan. Slang: Fast structured covariance approximations for Bayesian deep learning with natural gradient. In *Advances in Neural Information Processing Systems*, pages 6245–6255, 2018.

[50] D. Molchanov, A. Ashukha, and D. Vetrov. Variational dropout sparsifies deep neural networks. *arXiv preprint arXiv:1701.05369*, 2017.

[51] D. Molchanov, V. Kharitonov, A. Sobolev, and D. Vetrov. Doubly semi-implicit variational inference. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2593–2602, 2019.

[52] W. Mou, Y. Zhou, J. Gao, and L. Wang. Dropout training, data-dependent regularization, and generalization bounds. In *International Conference on Machine Learning*, pages 3645–3653, 2018.

[53] J. Mukhoti, P. Stenetorp, and Y. Gal. On the importance of strong baselines in Bayesian deep learning. *arXiv preprint arXiv:1811.09385*, 2018.

[54] M. P. Naeini, G. Cooper, and M. Hauskrecht. Obtaining well calibrated probabilities using Bayesian binning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[55] R. M. Neal. *Bayesian Learning for Neural Networks*. PhD thesis, University of Toronto, 1995.

[56] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[57] B. Neyshabur, S. Bhojanapalli, and N. Srebro. A pac-Bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017.

[58] V. M.-H. Ong, D. J. Nott, and M. S. Smith. Gaussian variational approximation with a factor covariance structure. *Journal of Computational and Graphical Statistics*, 27(3):465–478, 2018.

[59] K. Osawa, S. Swaroop, A. Jain, R. Eschenhagen, R. E. Turner, R. Yokota, and M. E. Khan. Practical deep learning with Bayesian principles. *arXiv preprint arXiv:1906.02506*, 2019.

[60] R. Ranganath, D. Tran, and D. Blei. Hierarchical variational models. In *International Conference on Machine Learning*, pages 324–333, 2016.

[61] D. Rezende and S. Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pages 1530–1538. PMLR, 2015.

[62] H. Ritter, A. Botev, and D. Barber. A scalable laplace approximation for neural networks. In *6th International Conference on Learning Representations, ICLR 2018-Conference Track Proceedings*, volume 6. International Conference on Representation Learning, 2018.

[63] S. Rossi, S. Marmin, and M. Filippone. Walsh-hadamard variational inference for Bayesian deep learning. *arXiv preprint arXiv:1905.11248*, 2019.

[64] L. Sagun, U. Evci, V. U. Guney, Y. Dauphin, and L. Bottou. Empirical analysis of the hessian of over-parametrized neural networks. *arXiv preprint arXiv:1706.04454*, 2017.

[65] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[66] S. Sun, C. Chen, and L. Carin. Learning structured weight uncertainty in Bayesian neural networks. In *Artificial Intelligence and Statistics*, pages 1283–1292, 2017.

[67] X. Sun and C. Bischof. A basis-kernel representation of orthogonal matrices. *SIAM journal on matrix analysis and applications*, 16(4):1184–1196, 1995.

[68] J. Swiatkowski, K. Roth, B. S. Veeling, L. Tran, J. V. Dillon, S. Mandt, J. Snoek, T. Salimans, R. Jenatton, and S. Nowozin. The k-tied normal distribution: A compact parameterization of Gaussian mean field posteriors in Bayesian neural networks. *arXiv preprint arXiv:2002.02655*, 2020.

[69] J. M. Tomczak and M. Welling. Improving variational auto-encoders using convex combination linear inverse autoregressive flow. *arXiv preprint arXiv:1706.02326*, 2017.

[70] M. Tomczak, S. Swaroop, and R. Turner. Efficient low rank Gaussian variational inference for neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.

[71] S. Wager, S. Wang, and P. S. Liang. Dropout training as adaptive regularization. In *Advances in Neural Information Processing Systems*, pages 351–359, 2013.

[72] S. Wang and C. Manning. Fast dropout training. In *international conference on machine learning*, pages 118–126. PMLR, 2013.

[73] C. Wei, S. Kakade, and T. Ma. The implicit and explicit regularization effects of dropout. In *International Conference on Machine Learning*, pages 10181–10192. PMLR, 2020.

[74] A. G. Wilson and P. Izmailov. Bayesian deep learning and a probabilistic perspective of generalization. *arXiv preprint arXiv:2002.08791*, 2020.

[75] M. Yin and M. Zhou. Semi-implicit variational inference. *arXiv preprint arXiv:1805.11183*, 2018.

[76] G. Zhang, S. Sun, D. Duvenaud, and R. Grosse. Noisy natural gradient as variational inference. In *International Conference on Machine Learning*, pages 5852–5861. PMLR, 2018.

[77] H. Zhao, Y.-H. H. Tsai, R. Salakhutdinov, and G. J. Gordon. Learning neural networks with adaptive regularization. *arXiv preprint arXiv:1907.06288*, 2019.

[78] O. Zobay et al. Variational Bayesian inference with Gaussian-mixture approximations. *Electronic Journal of Statistics*, 8(1):355–389, 2014.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [N/A]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...

(a) Did you state the full set of assumptions of all theoretical results? [Yes]
(b) Did you include complete proofs of all theoretical results? [Yes] See Appendix

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Appendix
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]