Dropout as a Structured Shrinkage Prior

Eric Nalisnick [1]   **José Miguel Hernández-Lobato** [1 2 3]   **Padhraic Smyth** [4]

Abstract

Dropout regularization of deep neural networks has been a mysterious yet effective tool to prevent overfitting. Explanations for its success range from the prevention of "co-adapted" weights to it being a form of cheap Bayesian inference. We propose a novel framework for understanding multiplicative noise in neural networks, considering continuous distributions as well as Bernoulli noise (i.e. dropout). We show that multiplicative noise induces structured shrinkage priors on a network's weights. We derive the equivalence through reparametrization properties of scale mixtures and without invoking any approximations. Given the equivalence, we then show that dropout's Monte Carlo training objective approximates marginal MAP estimation. We leverage these insights to propose a novel shrinkage framework for resnets, terming the prior *automatic depth determination* as it is the natural analog of *automatic relevance determination* for network depth. Lastly, we investigate two inference strategies that improve upon the aforementioned MAP approximation in regression benchmarks.

1. Introduction

Dropout regularization (Hinton et al., 2012; Srivastava et al., 2014) has become an essential tool for fitting large neural networks. Due to its success, a number of variants have been proposed (Wan et al., 2013; Wang & Manning, 2013; Huang et al., 2016; Singh et al., 2016; Achille & Soatto, 2018), including versions for recurrent (Ji et al., 2016; Krueger et al., 2017; Gal & Ghahramani, 2016c; Zolna et al., 2018) and convolutional (Tompson et al., 2015; Gal & Ghahramani, 2016a) architectures. The narratives attempting to

explain dropout's inner-workings and success are also plentiful. To give a few examples, Srivastava et al. (2014) argue it prevents "conspiracies" between hidden units, Hinton et al. (2012) claim it serves a role similar to sex in evolution, Baldi & Sadowski (2013) show it ensembles by taking the geometric mean of sub-models, Wager et al. (2013) explain it as an adaptive ridge penalty, and Gal & Ghahramani (2016b) suggest dropout performs quasi-Bayesian uncertainty estimation. While some prior work has shown strict equivalences for simple models such as linear regression (Baldi & Sadowski, 2013; Wager et al., 2013; 2014; Helmbold & Long, 2015), the general case of dropout in deep neural networks is analytically intractable, which is likely why no one narrative has come to prominence.

In this paper we propose a novel Bayesian interpretation of regularization via multiplicative noise—with dropout being the special case of Bernoulli noise. Unlike previous frameworks, our method of analysis works through reparametrizations that are agnostic to network architecture. By assuming nothing more than a Gaussian prior (which could be diffuse) on the weights, we show that multiplicative noise induces (marginally) a Gaussian scale mixture. This result is exact and has been exploited previously in Bayesian modeling (Kuo & Mallick, 1998). Not only do we lay bare multiplicative noise's distributional assumptions, but we also reveal the structure it induces on the network's weights. We find that noise applied to hidden units ties the scale parameters in the same way as *automatic relevance determination* (Neal, 1994; MacKay, 1994; Tipping, 2001), a well-studied shrinkage prior. We propose an extension of this prior for residual networks (He et al., 2016), allowing Bayesian inference to select the number of layers.

We also address our framework's implications for posterior inference. We show that dropout's Monte Carlo objective is a lower bound on the scale mixture model's marginal MAP objective. Decoupling dropout's model from inference is a novel and useful contribution as previous Bayesian interpretations have been grounded in variational inference (Gal & Ghahramani, 2016b; Kingma et al., 2015) and hence gave no guidance on how dropout could be used in conjunction with Markov chain Monte Carlo (MCMC). We then make algorithmic contributions of our own, describing a computationally efficient importance weighted objective and EM algorithm. We test these algorithms on benchmark regression

[1]Department of Engineering, University of Cambridge, Cambridge, United Kingdom [2]Microsoft Research, Cambridge, United Kingdom [3]Alan Turing Institute [4]Department of Computer Science, University of California, Irvine, United States of America. Correspondence to: Eric Nalisnick <e.nalisnick@eng.cam.ac.uk>.

tasks from the UCI repository (Dheeru & Karra Taniskidou, 2017), showing our proposals for light-weight inference improve upon traditional Monte Carlo dropout and are competitive to other high-capacity Bayesian models such as deep Gaussian processes trained with expectation propagation. Lastly, we leave the reader with some directions for future work.

2. Background

We use the following notation throughout the paper. Matrices are denoted with upper-case and bold letters (e.g. $\boldsymbol{X}$), vectors with lower-case and bold (e.g. $\boldsymbol{x}$), and scalars with no bolding (e.g. x or X). Data are assumed to be row vectors $\boldsymbol{x} \in \mathbb{R}^D$, and N independently and identically distributed observations constitute the empirical data set $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$. We focus on supervised learning tasks in which $\boldsymbol{X}$ are covariates (features) that are predictive of another variable $\boldsymbol{y} = \{y_1, \ldots, y_N\}$, which we assume is a one-dimensional regression response or index denoting a class label. Throughout we use r to index the rows of a matrix and j to index its columns. We define an L-layer *neural network* (NN) (Goodfellow et al., 2016) recursively as

$$
\begin{aligned}
\mathbb{E}[y_n|\boldsymbol{x}_n] &= g^{-1}(\boldsymbol{h}_{n,L}\boldsymbol{W}_{L+1}), \\
\boldsymbol{h}_{n,l} &= f_l(\boldsymbol{h}_{n,l-1}\boldsymbol{W}_l), \quad \boldsymbol{h}_{n,0} = \boldsymbol{x}_n
\end{aligned}
\tag{1}
$$

where $g(\cdot)$ is a link function following the GLM framework (Nelder & Baker, 2004). $\{\boldsymbol{W}_1, \ldots, \boldsymbol{W}_l, \ldots, \boldsymbol{W}_{L+1}\}$ are the parameters, a set of $D_{l-1} \times D_l$-dimensional matrices. We omit the bias terms to reduce notational clutter as they can be subsumed into the weight matrices. The function $f(\cdot)$ acts element-wise and is known as the *activation* function.

Multiplicative Noise Regularization (Dropout) Dropout training (Hinton et al., 2012; Srivastava et al., 2014) introduces *multiplicative noise* (MN) into the hidden layer computation defined in Equation 1:

$$
\boldsymbol{h}_{n,l} = f_l(\boldsymbol{h}_{n,l-1}\boldsymbol{\Lambda}_l\boldsymbol{W}_l)
\tag{2}
$$

where $\boldsymbol{\Lambda}_l$ is a diagonal $D_{l-1} \times D_{l-1}$-dimensional matrix of random variables $\lambda_{j,j}$ drawn independently from a noise distribution $p(\lambda)$. Dropout corresponds to $p(\lambda)$ being Bernoulli. However, other noise distributions such as Gaussian (Srivastava et al., 2014; Shen et al., 2018), Beta (Tomczak, 2013; Liu et al., 2019), and uniform (Shen et al., 2018) have been shown to be equally effective. Training under MN is done by sampling a new $\boldsymbol{\Lambda}_l$ matrix for every forward propagation. This sampling can be viewed as Monte Carlo (MC) integration over the noise distribution, and therefore, the MN optimization objective is to maximize w.r.t $\{\boldsymbol{W}_l\}_{l=1}^{L+1}$ the

function

$$
\begin{aligned}
&\mathcal{L}_{\text{MN}}(\{\boldsymbol{W}_l\}_{l=1}^{L+1}) \\
&= \mathbb{E}_{p(\lambda)}[\log p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_l\}_{l=1}^{L+1}, \{\boldsymbol{\Lambda}_l\}_{l=1}^L)] \\
&\approx \frac{1}{S} \sum_{s=1}^{S} \log p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_l\}_{l=1}^{L+1}, \{\hat{\boldsymbol{\Lambda}}_{l,s}\}_{l=1}^L)
\end{aligned}
\tag{3}
$$

where the expectation is taken with respect to $p(\lambda)$ and $\hat{\boldsymbol{\Lambda}}_{l,s}$ denotes the sth set of samples for the lth layer.

Automatic Relevance Determination *Automatic relevance determination* (ARD) (MacKay, 1994; Neal, 1994; Tipping, 2001) is a Bayesian regularization framework that consists of placing (usually) Gaussian priors on the NN's weights and then structured hyper-priors on the Gaussian scales. The scales of the weights in the same row (assuming the matrix orientation in Equation 1) are tied so they grow or shrink together in a form of group regularization. The end result is feature / hidden unit selection since if all of a unit's outgoing weights are near zero, then the unit is inconsequential to the model output. We can write the ARD prior as

$$
w_{l,r,j} \sim \text{N}(0, \sigma_{l,r}^2), \quad \sigma_{l,r} \sim p(\sigma)
\tag{4}
$$

where l is the index on layers, r is the index on rows in the weight matrix, and j is the index on its columns. Writing $\sigma_{l,r}$ without a column index signifies that all of the weights in the rth row share the same scale. Although we have defined ARD using a first-level Gaussian prior, other distributions could be used as long as they can be given the same scale structure.

3. Multiplicative Noise as Automatic Relevance Determination

We now discuss our first contribution: showing that regularization via MN induces, under mild assumptions, an ARD prior. The key observation is that if we assume the weights to be Gaussian random variables, the product $\boldsymbol{\Lambda}_l\boldsymbol{W}_l$ defines a Gaussian scale mixture (GSM) with ARD structure (*ARD-GSM prior* for short). We then show how the MC training objective in Equation 3 can be derived from this framework.

3.1. Gaussian Scale Mixtures

A random variable θ is a *Gaussian scale mixture* (GSM) if (and only if) it can be expressed as the product of a Gaussian random variable–call it z–with zero mean and some variance σ_0^2 and an independent scalar random variable α (Beale & Mallows, 1959; Andrews & Mallows, 1974):

$$
\theta \overset{d}{=} \alpha z, \quad z \sim \text{N}(0, \sigma_0^2), \quad \alpha \sim p(\alpha)
\tag{5}
$$

where $\overset{d}{=}$ denotes equality in distribution. The RHS is known as the GSM's *expanded parametrization* (Kuo

& Mallick, 1998). While it may not be obvious from Equation 5 that θ is a *scale* mixture, the result follows from the Gaussian's closure under linear transformations: $\alpha z \sim N(\alpha \cdot 0, \alpha^2 \cdot \sigma_0^2)$. Integrating out the scale gives the marginal density of θ: $p(\theta) = \int N(0, \sigma_0^2 \alpha^2) p(\alpha) d\alpha$ where $p(\alpha)$ is now clearly the mixing distribution. We call the form $N(w; 0, \sigma_0^2 \alpha^2) p(\alpha)$ the *hierarchical parametrization*. Super-Gaussian distributions, such as the student-t, Laplace, and horseshoe (Carvalho et al., 2009), can be represented as GSMs, and the hierarchical parametrization is often used for its convenience when employing them as robust priors (Steel, 2000).

3.2. Equivalence Between MN and ARD-GSM Priors

Now that we have defined GSMs, we demonstrate their relationship to MN. Assume we have an L-layer Bayesian NN with an ARD-GSM prior:

$$y_n \sim p(y_n | \boldsymbol{x}_n, \{\boldsymbol{W}_l\}_{l=1}^{L+1})$$
$$w_{l,r,j} \sim N(0, \sigma_0^2 \xi_{l,r}^2), \quad \xi_{l,r} \sim p(\xi) \tag{6}$$

where $w_{l,r,j}$ denotes the NN weights, σ_0^2 a constant shared across all weights, and $\xi_{l,r}$ a local (row-wise) random scale. Accordingly we have given $\xi_{l,r}$ a layer index (l) and row index (r) but not a column index (j), following Equation 4.

As the prior on the weights is a GSM, we can reparametrize the model into the GSM's equivalent expanded form given in Equation 5:

$$y_n \sim p(y_n | \boldsymbol{x}_n, \{\boldsymbol{W}_l\}_{l=1}^{L+1}, \{\boldsymbol{\Xi}_l\}_{l=1}^{L})$$
$$w_{l,r,j} \sim N(0, \sigma_0^2), \quad \xi_{l,r} \sim p(\xi) \tag{7}$$

where the weights are still denoted $w_{l,r,j}$ and drawn from a Gaussian with a fixed variance. The reparametrization changes the NN's hidden layer computation to:

$$\boldsymbol{h}_{n,l} = f_l(\boldsymbol{h}_{n,l-1}\boldsymbol{W}_l) \stackrel{\text{reparametrization}}{\longrightarrow} f_l(\boldsymbol{h}_{n,l-1}\boldsymbol{\Xi}_l\boldsymbol{W}_l) \tag{8}$$

where $\boldsymbol{\Xi}_l$ is a diagonal $D_{l-1} \times D_{l-1}$-dimensional matrix of scale values $\xi_{l,r}$. Notice that this expression (Equation 8) is Equation 2 exactly—with ξ in the place of λ—and thus we have shown the equivalence to MN. We have used no approximations. Moreover, the Gaussian assumption is not a strong one, and it can be relaxed by making σ_0^2 sufficiently large so that the prior is diffuse and therefore negligible. Previous attempts at analyzing MN have been frustrated by the activation function, which introduces a composition of non-linear functions that makes expectations analytically intractable. Because the expanded-vs-hierarchical reparametrization acts through the inner products, the activation function is bypassed.

3.3. Monte Carlo Training as Marginal MAP Inference

Next we derive the dropout / MN optimization objective given in Equation 3 from the ARD-GSM perspective.

Specifically, $\mathcal{L}_{\text{MN}}(\{\boldsymbol{W}_l\}_{l=1}^{L+1})$ is equivalent to a lower bound on the marginal MAP objective—i.e. the objective that would be optimized to find the weights' MAP estimate (assuming the optimum could be found). We begin by writing the unnormalized, marginal log-posterior over the weights as

$$\log p(\{\boldsymbol{W}_l\}_{l=1}^{L+1} | \boldsymbol{y}, \boldsymbol{X})$$
$$\propto \log \mathbb{E}_{p(\xi)} \left[p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_l\}_{l=1}^{L+1}, \{\boldsymbol{\Xi}_l\}_{l=1}^{L}) \right]$$
$$+ \frac{-1}{2\sigma_0^2} \sum_{l=1}^{L+1} \sum_{r=1}^{D_{l-1}} \sum_{j=1}^{D_l} w_{l,r,j}^2 \tag{9}$$

$$= \mathcal{L}_{\text{MAP}}(\{\boldsymbol{W}_l\}_{l=1}^{L+1}).$$

Next we use Jensen's inequality to lower-bound the first term (i.e. $\log \mathbb{E}_{p(\xi)}[\cdot] \geq \mathbb{E}_{p(\xi)}[\log \cdot]$):

$$\mathcal{L}_{\text{MAP}} \geq \mathbb{E}_{p(\xi)} \left[\log p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_l\}_{l=1}^{L+1}, \{\boldsymbol{\Xi}_l\}_{l=1}^{L}) \right]$$
$$+ \frac{-1}{2\sigma_0^2} \sum_{l=1}^{L+1} \sum_{r=1}^{D_{l-1}} \sum_{j=1}^{D_l} w_{l,r,j}^2 \tag{10}$$

$$= \mathcal{L}_{\text{MN}}(\{\boldsymbol{W}_l\}_{l=1}^{L+1}) + \frac{-1}{2\sigma_0^2} \sum_{l=1}^{L+1} ||\boldsymbol{W}_l||_F^2$$

where $\mathcal{L}_{\text{MN}}$ is the MN objective defined in Equation 3 and $|| \cdot ||_F$ is the Frobenius norm. Thus, the lower bound is equivalent to the MN objective with an additional L2 penalty (a.k.a. weight decay). Using weight decay and MN regularization together is not uncommon; Srivastava et al. (2014) (see their Table 9) and Gal & Ghahramani (2016b) both do so. The L2 term can be removed by assuming that the Gaussian prior is sufficiently diffuse:

$$\frac{-1}{2\sigma_0^2} \sum_{l=1}^{L+1} ||\boldsymbol{W}_l||_F^2 \to 0 \quad \text{as} \quad \sigma_0^2 \to \infty.$$

Increasing the Gaussian's variance requires adapting $p(\xi)$ to ensure the same shrinkage level but presents no technical difficulty otherwise. From here forward we use ξ to denote both MN and random scales and use λ on its own to denote MN schemes proposed in prior work.

3.4. Corresponding Priors

Having shown the equivalence between ARD-GSM priors and MN, we now discuss some specific noise distributions and their corresponding priors. Starting with dropout, the noise distribution is $\xi \sim \text{Bernoulli}(\pi)$, and this implies the prior on the Gaussian's variance is also Bernoulli, i.e. $\xi^2 \sim \text{Bernoulli}(\pi)$, since the square of a Bernoulli random variable is still a Bernoulli of the same distribution. The marginal prior on the NN weights is then

$$p_{\text{DROPOUT}}(w) = \sum_{\xi \in \{0,1\}} N(w; 0, \xi \sigma_0^2) \, p(\xi)$$
$$= \pi \, N(w; 0, \sigma_0^2) + (1 - \pi) \, \delta[w] \tag{11}$$

Noise Model $p(\xi)$	Variance Prior $p(\xi^2)$	Marginal Prior $p(w)$
Bernoulli	Bernoulli	Spike-and-Slab
Gaussian	χ^2	Gen. Hyperbolic
Rayleigh	Exponential	Laplace
Inverse Nakagami	Γ^{-1}	Student-t
Half-Cauchy	Unnamed	Horseshoe

Table 1. Noise Models and their Corresponding GSM Prior.

where $\delta[\cdot]$ denotes the delta function located at zero. This is the *spike-and-slab* prior commonly used for Bayesian variable selection (Mitchell & Beauchamp, 1988; George & McCulloch, 1993; Kuo & Mallick, 1998). Interestingly, the expanded parametrization was used for linear regression by Kuo & Mallick (1998), and thus their work should be considered a precursor to dropout. However, Kuo & Mallick (1998) were interested in obtaining the feature inclusion probabilities $p(\xi = 1|\boldsymbol{y}, \boldsymbol{X})$ rather than deriving a regularization mechanism to improve predictive performance. When dropout is performed without weight decay, its prior becomes $p_{\text{DROP}}(w) \propto \pi \mathbb{1} + (1 - \pi)\,\delta[w]$ where the improper uniform distribution $\mathbb{1}$ is derived by taking $\sigma_0 \to \infty$.

In Table 1, we list several additional noise models, their corresponding priors on the Gaussian variance, and their marginal distribution on the NN weights. Gaussian MN corresponds to a χ^2-distribution on the variance and a generalized hyperbolic (Barndorff-Nielsen, 1977) marginal distribution. Other notable cases are Rayleigh noise, which corresponds to a Laplace marginal, inverse Nakagami noise (Nakagami, 1960), which corresponds to a student-t, and half-Cauchy noise, which corresponds to the horseshoe prior (Carvalho et al., 2009).

3.5. Equivalence to DropConnect

If we assume all weights have independent scales, thus removing the ARD structure, the hidden layer computation in the expanded parametrization changes to: $\boldsymbol{h}_{n,l} = f_l(\boldsymbol{h}_{n,l-1}(\boldsymbol{\Xi}_l \odot \boldsymbol{W}_l))$ where $\odot$ denotes an element-wise product and $\boldsymbol{\Xi}_l$ is now a dense matrix of scale variables. Following the same derivation from this point reveals an equivalence to *DropConnect* regularization (Wan et al., 2013), which applies MN to each weight instead of each hidden unit. This absence of the regularizing ARD structure may explain why DropConnect has not been used as widely as dropout.

4. Extension to Resnets

With the previous section's insights in mind, we turn our attention to other NN architectures. *Resnets* (He et al., 2016) are NNs with *residual connections* (a.k.a. skip connections)

(Lang & Witbrock, 1988; He et al., 2016; Srivastava et al., 2015) between their hidden layers. Residual connections simply add the previous hidden state to the usual non-linear transformation: $\boldsymbol{h}_l = f_l(\boldsymbol{h}_{l-1}\boldsymbol{W}_l) + \boldsymbol{h}_{l-1}$. Since a residual connection allows information to bypass the non-linear transformation, entire weight matrices can be shrunk to zero without obstructing the NN's forward propagation. Thus we can create a prior that selects for *layers* by tying the variance of all weights in the same matrix. By collectively shrinking all the weights in coordination, we can reduce the layer's influence, effectively pruning it in the case of absolute shrinkage to zero. We term this prior *automatic depth determination* (ADD) as it is the natural analog of ARD for network depth. ADD is specified as

$$w_{l,r,j} \sim \text{N}(0, \sigma_0^2 \tau_l^2), \quad \tau_l \sim p(\tau) \tag{12}$$

where we have introduced the variable τ_l that acts as a *per-layer* group variance. We denote this structure by giving τ a layer index l but not a row or column index. As $p(\tau_l)$ places more density near zero, Bayesian inference will increasingly prefer to prune whole weight matrices, i.e. $\boldsymbol{h}_l \approx f_l(\boldsymbol{h}_{l-1}\boldsymbol{0}) + \boldsymbol{h}_{l-1} = \boldsymbol{h}_{l-1}$ (assuming f_l is a ReLU), making the network effectively more shallow. In the supplementary materials, we show how ADD can be formulated as MN, which reveals equivalences to *stochastic depth* regularization (Huang et al., 2016).

If Bayesian inference does not decide to prune an entire layer under the ADD prior, regularization still may be necessary. The natural progression is to then select the layer's number of hidden units—as ARD does. Therefore it makes sense to combine ARD and ADD so that the former takes effect when the latter imposes little to no regularization. The joint ARD-ADD prior is specified as

$$w_{l,i,j} \sim \text{N}(w_{l,i,j}; 0, \sigma_0^2 \xi_{l,r}\tau_l), \ \xi_{l,r} \sim p(\xi), \ \tau_l \sim p(\tau). \tag{13}$$

The priors remain essentially unchanged from their original definitions in Equations 4 and 12. The two multiplicatively interact in the first-level prior's variance, and therefore when $\tau_l \to 0$, the product $\lambda_{l,i}\tau_l \approx 0$, effectively turning off the influence of ARD. Conversely, when $\tau_l > 0$, then ARD will act as usual but have its effect modified by a factor of τ_l. Switching to the expanded parametrization, the hidden layer computation for ARD-ADD is $\boldsymbol{h}_{n,l} = f_l(\tau_l \boldsymbol{h}_{n,l-1}\boldsymbol{\Xi}_l\boldsymbol{W}_l) + \boldsymbol{h}_{n,l-1}$. Implementing the prior as MN would involve sampling τ_l and $\boldsymbol{\Xi}_l$ for each forward pass.

5. Rethinking Dropout Inference

We next turn towards training and inference, speculating: are there ways to make dropout 'more Bayesian'? Can we optimize a bound that is closer to the true marginal map
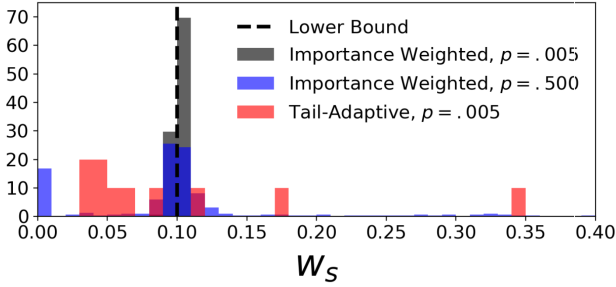
(a) Distribution of Importance Weights
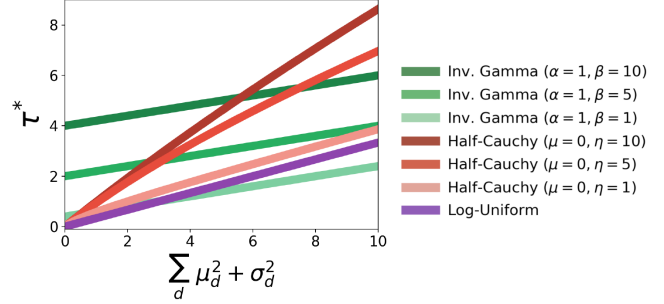
(b) EM Updates for Variance

Figure 1. Subfigure (a) shows the empirical distribution of importance weights observed when training on `Energy`. Subfigure (b) shows the EM updates for the posterior variance as a function of $q(\boldsymbol{W})$'s parameters.

objective? Or better yet, can we improve the posterior approximation while not incurring much additional cost in memory or computation? In the next two subsections, we address these questions and propose two solutions. Firstly, for traditional MC dropout we propose using a rank-based scheme for setting the importance weights, resulting in an objective that better covers posterior mass. Secondly, for applications in which there is room to perform a little more computation, we detail a light-weight variational EM algorithm for ARD-type priors.

5.1. Expanded vs Hierarchical Parametrization

Returning to Equation 10, recall that the MC dropout objective is a lower bound on the true marginal MAP objective. We are concerned with the gap between the two—$\mathcal{J}_{\mathrm{GAP}} = \mathcal{L}_{\mathrm{MAP}} - \mathcal{L}_{\mathrm{MN}}$—and wonder if it can be improved by changing parametrizations. MN is always applied in the expanded parametrization, but we could alternatively apply noise in the hierarchical parametrization:

$$\mathcal{L}_{\mathrm{MN\text{-}HP}} = \log p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_l\}_{l=1}^{L+1})$$
$$+ \frac{-1}{2\sigma_0^2} \sum_{l=1}^{L+1} \sum_{r=1}^{D_{l-1}} \mathbb{E}_{p(\xi)} \left[\frac{1}{\xi_{l,r}^2} \right] \sum_{j=1}^{D_l} w_{l,r,j}^2 \quad (14)$$

where the expectation would be computed with an MC approximation or solved for in closed-form. At first glance the above objective would seem to be better behaved as the NN is no longer being perturbed. However, in the supplementary materials we show that the expanded parametrization (i.e. likelihood noise) relaxes the MAP estimate as a function of $Var[\xi]$ whereas the alternative above has a $Var[\xi^{-2}]$-order gap. When $\mathbb{E}[\xi^2]$ is near zero, $Var[\xi^{-2}]$ explodes for all noise distributions considered in Table 1, resulting in an impractical objective. The intuition behind these results is easy to see in the case of Bernoulli noise: multiplying the NN's weights by $\xi \in \{0, 1\}$ is not a problem but plugging $\xi = 0$ into Equation 14 results in the second term becoming

undefined. The implication is that, of the two parametrizations considered, the expanded form provides the tightest bound to the true MAP objective under moderate to strong shrinkage.

5.2. Importance Weighted Objectives

The following MC importance weighted objective is attractive as it is tighter than the $\mathcal{L}_{\mathrm{MN}}$ lower bound (Burda et al., 2016; Noh et al., 2017):

$$\log \mathbb{E}_{p(\xi)}[p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_l\}_{l=1}^{L+1}, \{\boldsymbol{\Xi}_l\}_{l=1}^{L})]$$
$$\geq \log \frac{1}{S} \sum_{s=1}^{S} p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_l\}_{l=1}^{L+1}, \{\hat{\boldsymbol{\Xi}}_{l,s}\}_{l=1}^{L}) \quad (15)$$
$$= \mathcal{L}_{\mathrm{IW\text{-}MN}}(\{\boldsymbol{W}_l\}_{l=1}^{L+1})$$

where $\{\hat{\boldsymbol{\Xi}}_{l,s}\}_{l=1}^{L}$ is the sth sample from $p(\xi)$. While this objective is also a lower bound, it is guaranteed to become tighter with every additional sample, eventually converging to $\mathcal{L}_{\mathrm{MAP}}$ as $S \to \infty$ (Burda et al., 2016; Noh et al., 2017). The benefits of the IW objective can be seen in its gradient updates:

$$\nabla_{\boldsymbol{W}} \mathcal{L}_{\mathrm{IW\text{-}MN}}(\{\boldsymbol{W}_l\}_{l=1}^{L+1}) =$$
$$\sum_{s=1}^{S} w_s \nabla_{\boldsymbol{W}} \log p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_l\}_{l=1}^{L+1}, \{\hat{\boldsymbol{\Xi}}_{l,s}\}_{l=1}^{L}) \quad (16)$$

where $\tilde{w}_s = p(\boldsymbol{y}|\boldsymbol{X}, \{\boldsymbol{W}_{l,s}\}_{l=1}^{L+1}, \{\hat{\boldsymbol{\Xi}}_{l,s}\}_{l=1}^{L})$ and $w_s = \tilde{w}_s / \sum_k \tilde{w}_k$. Samples that result in a higher likelihood exhibit more influence on the aggregated gradient. Noh et al. (2017) showed that this IW objective improves dropout's performance in several vision tasks.

Tail-Adaptive Weights   Yet, if the motivation for using dropout is to perform cheap uncertainty quantification (Gal & Ghahramani, 2016b), we should be optimizing an objective that covers as much posterior mass as possible. This is

even more desirable when using parameter point estimates as hopefully better posterior exploration will mitigate the effect of such a poor posterior approximation. We propose using Wang et al. (2018)'s *tail-adaptive* method for setting the importance weights. This modifies the weights such that the objective optimizes an implicit mass-covering f-divergence. Due to space constraints, we refer the reader to Wang et al. (2018) for more details. Other strategies for modifying the weights could be employed (Ionides, 2008; Vehtari et al., 2015), but the tail-adaptive method is easy to implement, preserving the simplicity of dropout. Specifically, the weights are set according to the rank of each sample:

$$\gamma_s = \frac{S}{\sum_k \mathbb{1}[\tilde{w}_k \geq \tilde{w}_s]}, \quad w_s = \frac{\gamma_s}{\sum_k \gamma_k} \quad (17)$$

where $\tilde{w}_s$ is the likelihood under the sth sample as defined in Equation 16. These weights do not depend on the precise value of the likelihood and are only a function of the sample size S.

We visualize the importance weights produced by each method—the lower bound ($w \propto 1$), importance weighted ($w \propto p(\boldsymbol{y}|\cdot)$), and tail-adaptive ($w \propto \gamma_s$)—in Figure 1(a) for an experiment on the Energy UCI data set. The gray histogram shows the importance weights observed when using Gal & Ghahramani (2016b)'s dropout rate: $p = .005$, 10 samples. We see that the distribution is strongly peaked at $1/10 = .1$, i.e. the uniform weight used by the lower bound. The blue histogram shows when the dropout rate is increased to $p = .5$. Now there is less of a mode at .1, but optimization does not converge under this dropout setting without a significant increase in the number of hidden units. Hence, there is a fragile interdependence between the noise parameters, the network architecture, and the distribution of the importance weights. The red histogram shows the weight distribution for the tail-adaptive method. Even though the dropout rate is $p = .005$, there is still diversity in the weights, speaking to the method's ability to better explore the posterior.

5.3. Light-Weight Inference via Variational EM

Ideally we would like to go beyond MAP point estimates and obtain some form of posterior *distribution*. Unfortunately, even approximate inference for scale mixture priors can be challenging—especially when the hyper-priors are half-Cauchy or log-uniform. Previous work performing variational inference for these and similar priors has had to incorporate truncated approximations (Neklyudov et al., 2017), auxiliary variables (Ghosh et al., 2018; Louizos et al., 2017), non-centered parametrizations (Ingraham & Marks, 2017; Ghosh et al., 2018; Louizos et al., 2017), and quasi-divergences (Hron et al., 2018) for the sake of tractability. To preserve the simplicity of dropout, we propose

a light-weight inference procedure derived through *variational expectation-maximization* (EM) (Beal & Ghahramani, 2003). For most variational Bayesian NN implementations, using variational EM with ADD or ARD priors can be implemented in a few lines of code—possibly in as little as one. Below we detail the inference procedure for ADD, leaving the description for the ARD-ADD prior to supplementary materials.

Following Wu et al. (2019), we assume the posterior approximation $p(\boldsymbol{W}, \tau | \boldsymbol{y}, \boldsymbol{X}) \approx q(\boldsymbol{W}; \phi)q(\tau) = \mathrm{N}(\boldsymbol{W}; \boldsymbol{\mu}_\phi, \mathrm{diag}\{\boldsymbol{\Sigma}_\phi\})\delta[\bar{\tau}_l]$ where $\phi = \{\boldsymbol{\mu}_\phi, \mathrm{diag}\{\boldsymbol{\Sigma}_\phi\}\}$ and $\bar{\tau}_l$ are the variational parameters. We assume $\delta[\bar{\tau}_l]$ is a *pseudo-Dirac delta* (Nakajima & Sugiyama, 2014) so that the distribution has finite entropy. The evidence lower bound (ELBO) for this approximation is

$$\begin{aligned}
\log p(\boldsymbol{y}|\boldsymbol{X}) &\geq \mathbb{E}_{q(\boldsymbol{W})}\left[\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{W})\right] \\
&\quad - \mathbb{E}_{q(\tau)}\mathrm{KL}\left[q(\boldsymbol{W}; \phi)||p(\boldsymbol{W}|\tau)\right] - \mathrm{KL}\left[q(\tau)||p(\tau)\right] \\
&= \mathbb{E}_{\mathrm{N}(\boldsymbol{W})}\left[\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{W})\right] \\
&\quad - \mathrm{KL}\left[\mathrm{N}(\boldsymbol{W}; \phi)||\mathrm{N}(\boldsymbol{W}|\bar{\tau}_l)\right] - \mathrm{KL}\left[\delta[\bar{\tau}_l]||p(\tau)\right] \\
&= \mathbb{E}_{\mathrm{N}(\boldsymbol{W})}\left[\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{W})\right] \\
&\quad - \mathrm{KL}\left[\mathrm{N}(\boldsymbol{W}; \phi)||\mathrm{N}(\boldsymbol{W}|\bar{\tau}_l)\right] + \log p(\bar{\tau}_l) + \mathcal{C},
\end{aligned} \quad (18)$$

where $\mathcal{C} = \mathbb{H}[\delta[\bar{\tau}_l]]$ is a constant. For inverse-gamma, half-Cauchy, and log-uniform hyper-priors (and possibly others), $\bar{\tau}_l$ has a closed-form solution that can be found by differentiating the ELBO (Equation 18) and setting to zero. We denote the optimal solution as $\bar{\tau}_l^*$ and give its formula for each hyper-prior in the supplementary materials. No closed-form exists for updating $q(\boldsymbol{W}; \phi)$, and hence we perform gradient ascent updates using

$$\begin{aligned}
\frac{\partial}{\partial \phi}\mathcal{J}_{\mathrm{ELBO}}(\phi, \bar{\tau}_l^*) &= \frac{\partial}{\partial \phi}\mathbb{E}_{\mathrm{N}(\boldsymbol{W}; \phi)}\left[\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{W})\right] \\
&\quad - \frac{\partial}{\partial \phi}\mathrm{KL}\left[\mathrm{N}(\boldsymbol{W}; \phi)||\mathrm{N}(\boldsymbol{W}|\bar{\tau}_l^*)\right].
\end{aligned} \quad (19)$$

Figure 1 (b) shows the value for $\bar{\tau}_l^*$ as a function of the variational parameters $\boldsymbol{\mu}_\phi$ and $\boldsymbol{\sigma}_\phi^2$. The slope and intercept of each line convey the prior's shrinkage properties. Only the log-uniform and half-Cauchy provide true sparsity, allowing for $\bar{\tau}_l^* = 0$ when $\boldsymbol{\mu}_\phi^2 + \boldsymbol{\sigma}_\phi^2 = 0$, no matter the setting of the prior's scale. The inverse Gamma, on the other hand, can set $\bar{\tau}_l^* = 0$ only in the limit as $\alpha \to 0$ and $\beta \to 0$.

6. Related Work

Gal & Ghahramani (2016b;c)'s interpretation of dropout as a variational approximation is perhaps the best known work contextualizing dropout within the Bayesian paradigm. Their variational model is a spike-and-slab distribution and thus is equivalent to our generative model when $p(\xi)$ is

Bernoulli. However, there are crucial differences between their formulation and ours. Firstly, their framework does not separate the model from inference, providing no direction on how one could employ dropout if performing inference via MCMC, for instance. Since we work in terms of priors, MCMC can be applied as usual. Secondly, Gal & Ghahramani (2016b;c) define their variational model as having NN weights $W = M \operatorname{diag}[\mathbf{z}]$ where M is a Gaussian matrix and $\operatorname{diag}[\mathbf{z}]$ is a diagonal matrix with Bernoulli variables drawn from a *fixed* distribution. As far as we are aware, there is no reason for why the noise distribution must remain fixed, and in follow-up work Gal et al. (2017) relax this restriction, which had also been explored by Ba & Frey (2013), Wang & Manning (2013), Maeda (2014), Srinivas & Venkatesh Babu (2016), and Zhai & Wang (2018). Our work, on the other hand, derives their variational approximation from the perspective of structured priors. Since the noise distribution is considered *a prior*, it is natural that the dropout probability remains fixed and not optimized. Consequently, our framework withstands the criticism that variational dropout's posterior does not contract as more data arrives (Osband, 2016).

Kingma et al. (2015) also propose a variational interpretation of dropout, which again couples the model with the inference strategy. Their approach is derived by reparametrizing noise on the weights as uncertainty in the hidden units. They show that their variational framework implies a log-uniform prior: $p(w) \propto 1/|w|$. While our proposed GSM priors do not exactly match Kingma et al. (2015)'s prior, the log-uniform does have heavy tails and strong shrinkage behavior near the origin, and interestingly, many of the marginal priors in the GSM family such as the student-t and horseshoe have those same characteristics. However, recent work by Hron et al. (2018) illuminates flaws in the KL divergence term derived by Kingma et al. (2015) for their implicit prior—specifically, that it results in an improper posterior. Again, as our framework is removed from the variational paradigm and uses well-studied priors, this criticism does not apply. Molchanov et al. (2017) points out that Kingma et al. (2015)'s dropout has the ARD structure, but they do not consider how this structure is derived from MN a priori nor ARD's extension to other architectures, as we do.

Not all previous work has assumed a variational interpretation. General treatments of data and/or parameter corruption as a Bayesian prior were considered by Herlau et al. (2015) and Nalisnick & Smyth (2018). The connection between dropout and the spike-and-slab prior has been noted by several previous works (Louizos, 2015; Mohamed, 2015; Ingraham & Marks, 2017; Polson & Rockova, 2018). Ingraham & Marks (2017) even note that dropout corresponds to "scale noise," but their primary focus was inference for undirected graphical models, not NNs. Our work is distinct from these approaches in that none of them explicitly

showed the equivalence via Kuo & Mallick (1998)'s expanded parametrization, discussed other scale distributions, performed analysis of the MN objective, or generalized the interpretation to other architectures.

Regarding the inference algorithms discussed in Section 5, Noh et al. (2017) recognized that the importance weighted objective in Equation 15, which was first proposed by Burda et al. (2016) for variational inference, would provide a better estimator of the noise-marginalized likelihood. However, they do not draw any connections to Bayesian inference. Wang et al. (2018)'s tail-adaptive method was proposed for general variational inference and its application to MN regularization has not been previously investigated. The variational EM algorithm described in Section 5.3 was inspired by Wu et al. (2019)'s "empirical Bayes" procedure.[1] We have extended their framework by considering structured priors and deriving the scale updates for additional hyperpriors (e.g. half-Cauchy and log-uniform).

7. Experiments

We performed experiments to test the practicality of the tail-adaptive importance sampling scheme (Section 5.2) for MC dropout and the variational EM algorithm (Section 5.3) for ARD, ADD, and ARD-ADD priors (Section 4). For both cases we used the same experimental setup as Gal & Ghahramani (2016b), testing the models on regression tasks from the UCI repository (Dheeru & Karra Taniskidou, 2017). The supplementary materials include details of the model and optimization hyperparameters as well as Python implementations[2] of both experiments.

Tail-Adaptive Importance Sampling Table 2 reports test set root-mean-square error (RMSE) and log-likelihood for three MN objectives: the usual lower bound (Equation 3), the importance weighted objective (Equation 15), and the tail-adaptive method (Equation 17). The 'lower bound' columns are the results reported by Gal & Ghahramani (2016b).[3] The only difference between columns is in how the importance weights were set. We see that the tail-adaptive method results in the best RMSE in five out of seven data sets. However, the best log-likelihoods are achieved by regular importance sampling (five out of seven). We believe that this is due to $\mathcal{L}_{\text{IW-MN}}$ being a less-biased estimate of the exact likelihood (unbiased as $S \to \infty$). The tail-adaptive method, on the other hand, is most effective at regularizing for purposes of predictive accuracy. Notably,

---

[1] We do not follow Wu et al. (2019)'s "empirical Bayes" naming convention because the hyperprior's parameters remain fixed and are not chosen by the data (MacKay, 1992).

[2] Available at: `https://github.com/enalisnick/dropout_icml2019`

[3] We report Gal & Ghahramani (2016b)'s updated experiments that appear in Appendix Table 2 of the ArXiv version.

| | Test Set RMSE | | | Test Log-Likelihood | | |
	Lower Bound	Import. Weighted	Tail-Adaptive	Lower Bound	Import. Weighted	Tail-Adaptive
Boston	2.80 ±.19	2.450 ±.25	**2.369** ±.22	−2.39 ±.05	−2.352 ±.10	**−2.346** ±.01
Concrete	4.81 ±.14	4.052 ±.29	**3.935** ±.35	−2.94 ±.02	**−2.888** ±.08	−2.940 ±.13
Energy	1.09 ±.05	0.972 ±.06	**0.828** ±.05	−1.72 ±.02	**−1.339** ±.06	−1.349 ±.04
Kin8nm	0.09 ±.00	0.082 ±.00	**0.076** ±.00	0.97 ±.01	**1.119** ±.03	1.105 ±.03
Power	4.00 ±.04	**3.094** ±.08	3.286 ±.08	−2.79 ±.01	**−2.775** ±.04	−2.809 ±.05
Wine	0.61 ±.01	0.667 ±.02	**0.559** ±.03	**−0.92** ±.01	−0.996 ±.04	−0.962 ±.07
Yacht	0.72 ±.06	**0.577** ±.16	0.612 ±.14	−1.38 ±.01	**−1.274** ±.12	−1.290 ±.11

Table 2. Comparing Monte Carlo Objectives. We compare test set RMSE and test log-likelihood for UCI regression benchmarks. Gal & Ghahramani (2016b)'s results are reported in the *lower bound* column.

| | Test Set RMSE | | | | | | | |
	Dropout	Prob. Backprop	Deep GP	ARD	ADD	ARD-ADD		
Boston	2.80 ±.13	2.795 ±.16	2.38 ±.12	**2.158** ±.20	2.343 ±.31	2.367 ±.18		
Concrete	4.50 ±.18	5.241 ±.12	4.64 ±.11	3.805 ±.28	4.084 ±.34	**3.761** ±.23		
Energy	**0.47** ±.01	0.903 ±.05	0.57 ±.02	0.852 ±.01	0.867 ±.11	0.853 ±.08		
Kin8nm	0.08 ±.00	0.071 ±.00	**0.05** ±.00	0.066 ±.01	0.064 ±.00	0.064 ±.00		
Power	3.63 ±.04	4.028 ±.03	3.60 ±.03	3.486 ±.10	3.290 ±.06	**3.236** ±.07		
Wine	0.60 ±.01	0.643 ±.01	**0.50** ±.01	0.561 ±.03	0.555 ±.01	0.538 ±.03		
Yacht	0.66 ±.06	0.848 ±.05	0.98 ±.09	0.691 ±.12	0.657 ±.14	**0.604** ±.16		
Avg. Rank	4.4 ±1.7	5.6 ±0.5	3.1 ±1.8	3.0 ±1.1	2.9 ±10	**2.0** ±1.1		

Figure 2. Posterior Structure.

Table 3. We compare test set RMSE for UCI regression benchmarks. As baselines, we use previously reported two-hidden-layer results for dropout (Gal & Ghahramani, 2016b), probabilistic backpropagation (Hernández-Lobato & Adams, 2015), and deep Gaussian processes (Bui et al., 2016). ARD, ADD, and ARD-ADD use the $\Gamma^{-1}(3, 3)$ scale prior in all cases.
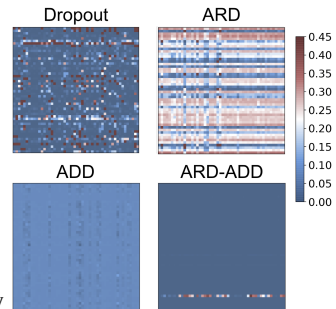
the tail-adaptive method performs worst on Power, the largest data set ($N = 9568$).

EM for Resnets We next report results for the variational EM algorithm applied to resnets with two hidden layers (one skip connection) and ARD, ADD, and ARD-ADD priors. We compare their test set RMSE with the two-layer results reported for dropout (Gal & Ghahramani, 2016b), probabilistic backpropagation (Hernández-Lobato & Adams, 2015), and deep Gaussian processes (GPs) trained with expectation propagation (EP) (Bui et al., 2016). Table 3 contains the results: the ARD-ADD resnet gives the best performance on three data sets, and the deep GP gives the best on two. The average rank of each model / algorithm is given in the last line with ARD-ADD coming in first (but there is substantial overlap in the error bars). This result is encouraging since the EM-trained ARD-ADD resnet can compete with the deep GP, a rich nonparametric model, trained with EP, a strong approximate inference algorithm (Vehtari et al., 2014).

Figure 2 shows heat maps of the hidden-to-hidden weight matrices for dropout and the three ARD/ADD priors when trained on Boston. The colors are determined by the summed posterior moments $\mu_\phi^2 + \sigma_\phi^2$ (just w^2 for dropout) as this quantifies each parameter's ability to deviate from zero. We see that, as expected, ARD induces row-structured

shrinkage, ADD induces matrix-wide shrinkage, and ARD-ADD allows some rows to grow (just one in this case) while preserving global shrinkage. While not strictly comparable to the others, MC dropout's heat map seems to balance having some row structure with strong global shrinkage.

8. Conclusions

We have presented a novel interpretation of MN, showing that it induces an ARD-GSM prior. This revelation uncouples dropout's generative assumptions from inference, unlike previously proposed Bayesian interpretations (Kingma et al., 2015; Gal & Ghahramani, 2016b). Our Bayesian framework then inspires an extension of ARD to resnets, a novel prior that we call *automatic depth determination*, and the application of two alternative inference algorithms. An exciting direction for future work is to extend the ARD framework to recurrent and convolutional networks. Bayesian inference for these architectures has proven to be challenging (Gal & Ghahramani, 2016a; Fortunato et al., 2017), and the structured priors and efficient inference algorithms we explore in this work may enable a notable jump in progress.

References

Achille, A. and Soatto, S. Information Dropout: Learning Optimal Representations Through Noisy Computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

Andrews, D. F. and Mallows, C. L. Scale Mixtures of Normal Distributions. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 99–102, 1974.

Ba, J. and Frey, B. Adaptive Dropout for Training Deep Neural Networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3084–3092, 2013.

Baldi, P. and Sadowski, P. J. Understanding Dropout. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2814–2822, 2013.

Barndorff-Nielsen, O. Exponentially Decreasing Distributions for the Logarithm of Particle Size. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 353:401–419, 1977.

Beal, M. J. and Ghahramani, Z. The Variational Bayesian EM Algorithm for Incomplete Data: with application to scoring graphical model structures. *Bayesian Statistics*, 7:453–464, 2003.

Beale, E. and Mallows, C. Scale Mixing of Symmetric Distributions with Zero Means. *The Annals of Mathematical Statistics*, 30(4):1145–1151, 1959.

Bui, T., Hernandez-Lobato, D., Hernandez-Lobato, J., Li, Y., and Turner, R. Deep Gaussian Processes for Regression using Approximate Expectation Propagation. In *Proceedings of The 33rd International Conference on Machine Learning (ICML)*, pp. 1472–1481, 2016.

Burda, Y., Grosse, R., and Salakhutdinov, R. Importance Weighted Autoencoders. *International Conference on Learning Representations (ICLR)*, 2016.

Carvalho, C. M., Polson, N. G., and Scott, J. G. Handling Sparsity via the Horseshoe. In *International Conference on Artificial Intelligence and Statistics (AIStats)*, pp. 73–80, 2009.

Dheeru, D. and Karra Taniskidou, E. UCI Machine Learning Repository, 2017. URL http://archive.ics.uci.edu/ml.

Fortunato, M., Blundell, C., and Vinyals, O. Bayesian Recurrent Neural Networks. *ArXiv e-prints*, 2017.

Gal, Y. and Ghahramani, Z. Bayesian Convolutional Neural Networks with Bernoulli Approximate Variational Inference. *ICLR Workshop Track*, 2016a.

Gal, Y. and Ghahramani, Z. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pp. 1050–1059, 2016b.

Gal, Y. and Ghahramani, Z. A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1019–1027, 2016c.

Gal, Y., Hron, J., and Kendall, A. Concrete Dropout. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3581–3590, 2017.

George, E. I. and McCulloch, R. E. Variable Selection via Gibbs Sampling. *Journal of the American Statistical Association*, 88(423):881–889, 1993.

Ghosh, S., Yao, J., and Doshi-Velez, F. Structured Variational Learning of Bayesian Neural Networks with Horseshoe Priors. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018.

Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. MIT press, 2016.

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Helmbold, D. P. and Long, P. M. On the Inductive Bias of Dropout. *The Journal of Machine Learning Research*, 16(1):3403–3454, 2015.

Herlau, T., Mørup, M., and Schmidt, M. N. Bayesian Dropout. *ArXiv e-prints*, 2015.

Hernández-Lobato, J. M. and Adams, R. Probabilistic Backpropagation for Scalable Learning of Bayesian Neural Networks. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pp. 1861–1869, 2015.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors. *ArXiv e-prints*, 2012.

Hron, J., Matthews, A., and Ghahramani, Z. Variational Bayesian Dropout: Pitfalls and Fixes. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 1–8, 2018.

Huang, G., Sun, Y., Liu, Z., Sedra, D., and Weinberger, K. Q. Deep Networks with Stochastic Depth. In *European Conference on Computer Vision (ECCV)*, pp. 646–661. Springer, 2016.

Ingraham, J. and Marks, D. Variational Inference for Sparse and Undirected Models. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 1607–1616, 2017.

Ionides, E. L. Truncated Importance Sampling. *Journal of Computational and Graphical Statistics*, 17(2):295–311, 2008.

Ji, S., Vishwanathan, S., Satish, N., Anderson, M. J., and Dubey, P. Blackout: Speeding Up Recurrent Neural Network Language Models with Very Large Vocabularies. *International Conference on Learning Representations (ICLR)*, 2016.

Kingma, D. P., Salimans, T., and Welling, M. Variational Dropout and the Local Reparameterization Trick. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

Krueger, D., Maharaj, T., Kramár, J., Pezeshki, M., Ballas, N., Ke, N. R., Goyal, A., Bengio, Y., Courville, A., and Pal, C. Zoneout: Regularizing RNNs by Randomly Preserving Hidden Activations. *International Conference on Learning Representations (ICLR)*, 2017.

Kuo, L. and Mallick, B. Variable Selection for Regression Models. *Sankhyā: The Indian Journal of Statistics, Series B*, pp. 65–81, 1998.

Lang, K. J. and Witbrock, M. Learning to Tell Two Spirals Apart. In *1988 Connectionist Models Summer School*, 1988.

Liu, L., Luo, Y., Shen, X., Sun, M., and Li, B. β-Dropout: a Unified Dropout. *IEEE Access*, 2019.

Louizos, C. Smart Regularization of Deep Architectures. *Masters Thesis, University of Amsterdam*, 2015.

Louizos, C., Ullrich, K., and Welling, M. Bayesian Compression for Deep Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3288–3298, 2017.

MacKay, D. J. Bayesian Interpolation. *Neural Computation*, 4(3):415–447, 1992.

MacKay, D. J. Bayesian Non-Linear Modeling for the Prediction Competition. In *Maximum Entropy and Bayesian Methods*, pp. 221–234. Springer, 1994.

Maeda, S.-i. A Bayesian Encourages Dropout. *ArXiv e-prints*, 2014.

Mitchell, T. J. and Beauchamp, J. J. Bayesian Variable Selection in Linear Regression. *Journal of the American Statistical Association*, 83(404):1023–1032, 1988.

Mohamed, S. A Statistical View of Deep Learning. *ArXiv e-prints*, 2015.

Molchanov, D., Ashukha, A., and Vetrov, D. Variational Dropout Sparsifies Deep Neural Networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 2498–2507, 2017.

Nakagami, M. The M-Distribution: A General Formula of Intensity Distribution of Rapid Fading. In *Statistical Methods in Radio Wave Propagation*, pp. 3–36. Elsevier, 1960.

Nakajima, S. and Sugiyama, M. Analysis of Empirical MAP and Empirical Partially Bayes: Can They be Alternatives to Variational Bayes? In *International Conference on Artificial Intelligence and Statistics (AIStats)*, pp. 20–28, 2014.

Nalisnick, E. and Smyth, P. Learning Priors for Invariance. In *International Conference on Artificial Intelligence and Statistics (AIStats)*, pp. 366–375, 2018.

Neal, R. M. *Bayesian Learning for Neural Networks*. PhD thesis, University of Toronto, 1994.

Neklyudov, K., Molchanov, D., Ashukha, A., and Vetrov, D. P. Structured Bayesian Pruning via Log-Normal Multiplicative Noise. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 6778–6787, 2017.

Nelder, J. A. and Baker, R. J. Generalized Linear Models. *Encyclopedia of Statistical Sciences*, 4, 2004.

Noh, H., You, T., Mun, J., and Han, B. Regularizing Deep Neural Networks by Noise: Its Interpretation and Optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5109–5118, 2017.

Osband, I. Risk Versus Uncertainty in Deep Learning: Bayes, Bootstrap and the Dangers of Dropout. *NIPS Workshop on Bayesian Deep Learning*, 2016.

Polson, N. and Rockova, V. Posterior Concentration for Sparse Deep Learning. *ArXiv e-prints*, 2018.

Shen, X., Tian, X., Liu, T., Xu, F., and Tao, D. Continuous Dropout. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2018.

Singh, S., Hoiem, D., and Forsyth, D. Swapout: Learning an Ensemble of Deep Architectures. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 28–36, 2016.

Srinivas, S. and Venkatesh Babu, R. Generalized Dropout. *ArXiv e-prints*, 2016.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Srivastava, R. K., Greff, K., and Schmidhuber, J. Training Very Deep Networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2377–2385, 2015.

Steel, M. F. Bayesian Regression Analysis With Scale Mixtures of Normals. *Econometric Theory*, 16(01):80–101, 2000.

Tipping, M. E. Sparse Bayesian Learning and the Relevance Vector Machine. *The Journal of Machine Learning Research*, 1:211–244, 2001.

Tomczak, J. M. Prediction of Breast Cancer Recurrence Using Classification Restricted Boltzmann Machine with Dropping. *ArXiv e-prints*, 2013.

Tompson, J., Goroshin, R., Jain, A., LeCun, Y., and Bregler, C. Efficient Object Localization Using Convolutional Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 648–656, 2015.

Vehtari, A., Gelman, A., Sivula, T., Jylänki, P., Tran, D., Sahai, S., Blomstedt, P., Cunningham, J. P., Schiminovich, D., and Robert, C. Expectation Propagation as a Way of Life: A framework for Bayesian inference on partitioned data. *ArXiv e-prints*, 2014.

Vehtari, A., Gelman, A., and Gabry, J. Pareto Smoothed Importance Sampling. *ArXiv Preprint*, 2015.

Wager, S., Wang, S., and Liang, P. S. Dropout Training as Adaptive Regularization. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 351–359, 2013.

Wager, S., Fithian, W., Wang, S., and Liang, P. S. Altitude Training: Strong Bounds for Single-Layer Dropout. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 100–108, 2014.

Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., and Fergus, R. Regularization of Neural Networks Using DropConnect. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pp. 1058–1066, 2013.

Wang, D., Liu, H., and Liu, Q. Variational Inference with Tail-adaptive f-Divergence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Wang, S. and Manning, C. Fast Dropout Training. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pp. 118–126, 2013.

Wu, A., Nowozin, S., Meeds, E., Turner, R. E., Hernández-Lobato, J. M., and Gaunt, A. L. Deterministic Variational Inference for Robust Bayesian Neural Networks. *International Conference on Learning Representations (ICLR)*, 2019.

Zhai, K. and Wang, H. Adaptive Dropout with Rademacher Complexity Regularization. In *International Conference on Learning Representations (ICLR)*, 2018.

Zolna, K., Arpit, D., Suhubdy, D., and Bengio, Y. Fraternal Dropout. *International Conference on Learning Representations (ICLR)*, 2018.