3rd Symposium on Advances in Approximate Bayesian Inference, 2020 1–10

Gradient Regularisation as Approximate Variational Inference

Ali Unlu[*] A.UNLU@SUSSEX.AC.UK
University of Sussex

Laurence Aitchison LAURENCE.AITCHISON@BRISTOL.AC.UK
University of Bristol

Abstract

Variational inference in Bayesian neural networks is usually performed using stochastic sampling which gives very high-variance gradients, and hence slow learning. Here, we show that it is possible to obtain a deterministic approximation of the ELBO for a Bayesian neural network by doing a Taylor-series expansion around the mean of the current variational distribution. The resulting approximate ELBO is the training-log-likelihood plus a squared gradient regulariser. In addition to learning the approximate posterior variance, we also consider a uniform-variance approximate posterior, inspired by the stationary distribution of SGD. The corresponding approximate ELBO has a simple form, as the log-likelihood plus a simple squared-gradient regulariser. We argue that this squared-gradient regularisation may at the root of the excellent empirical performance of SGD.

1. Introduction

Neural networks are increasingly being used in safety-critical settings such as self-driving cars (Bojarski et al., 2016) and medical diagnosis (Amato et al., 2013). In these settings, it is critical to be able to reason about uncertainty in the parameters of the network, for instance so that the system is able to call for additional human input when necessary (McAllister et al., 2017). Several approaches to Bayesian inference in neural networks are available, including from stochastic gradient Langevin dynamics (Welling and Teh, 2011) but here we focus on variational inference (Blundell et al., 2015).

Despite the theoretical advantages of Bayesian neural networks (e.g. their relationship with PAC-Bayesian methods that provide bounds on generalisation error Germain et al., 2016; Rivasplata et al., 2019), they are rare amongst networks claiming to give state-of-the-art performance on tasks such as image classification. Instead, much simpler techniques such as stochastic gradient descent (SGD) (Krizhevsky et al., 2012) are found to give excellent performance in practice. Indeed, SGD is also found to give better performance than adaptive optimizers such as Adam (Kingma and Ba, 2014; Keskar and Socher, 2017; Loshchilov and Hutter, 2017; Wilson et al., 2017). This is usually understood as regularisation that is implicitly embodied in SGD (Keskar et al., 2016; Wu et al., 2017; Lei et al., 2018; Roberts, 2018). While SGD's implicit regularisation gives excellent practical performance, it becomes problematic when trying to improve the optimizer used in deep learning. In the ideal case, we would be able to separate the objective (including regularisation terms) from the optimizer, such that e.g. improvements in the optimizer's convergence rate can always be

[*] Work done while at University of Bristol

expected to improve performance. In contrast, at the moment, methods that converge faster (such as Adam; Kingma and Ba, 2014) are generally believed to have worse performance at convergence, due to the lack of the implicit regularisation embodied in SGD (Wilson et al., 2017; Keskar and Socher, 2017). Therefore, it is important to understand the implicit regularisation embodied in SGD, such that we can use that regularisation in combination with other optimizers.

Here, we start by noting that variational inference for Bayesian neural networks typically involves stochastic sampling, which can give rise to high-variance gradients. We note that we can form a deterministic approximation of the ELBO by doing a second-order Taylor expansion around the mode of the approximate posterior. This gives an approximate ELBO that is composed of the log-likelihood, standard weight-decay regularisation and a squared-gradient regulariser, which is weighted by the variance of the approximate posterior. Next, we noted that the implicit regularisation effects of SGD can be understood using the *isotropic* Gaussian stationary distribution under locally quadratic loss functions (Mandt et al., 2017). Inspired by this stationary distribution, we considered uniform-variance approximate posteriors, which correspond to a simpler squared-gradient regulariser. Incorporating this regulariser improved performance on standard benchmark tasks, and provided a potential explanation for the excellent performance of stochastic gradient descent.

2. Background

2.1. Variational inference for Bayesian neural networks

Following the usual convention (Blundell et al., 2015), we use independent Gaussian priors and approximate posteriors for all parameters,

$$\mathrm{P}\left(w_\lambda\right) = \mathcal{N}\left(w_\lambda; 0, s_\lambda^2\right) \tag{1}$$

$$\mathrm{Q}\left(w_\lambda\right) = \mathcal{N}\left(w_\lambda; \mu_\lambda, \sigma_\lambda^2\right) \qquad \text{equivalently} \qquad \mathrm{Q}\left(\mathbf{w}\right) = \mathcal{N}\left(\mathbf{w}; \boldsymbol{\mu}, \boldsymbol{\Sigma}\right), \tag{2}$$

where μ_λ and σ_λ^2 are learned parameters of the approximate posterior, and where $\boldsymbol{\Sigma}$ is a diagonal matrix, with $\Sigma_{\lambda\lambda} = \sigma_\lambda^2$. Please see Appendix A for more information.

2.2. The stationary distribution of SGD

When we solve for steady-state in which $\boldsymbol{\Sigma} = \mathbb{C}\left[\mathbf{w}(t+1)\right] = \mathbb{C}\left[\mathbf{w}(t)\right]$, we derive that $\boldsymbol{\Sigma} \approx \frac{\eta}{2S}\mathbf{I}$. For the derivation, please see Appendix B.

3. Methods

3.1. Deterministic approximations to variational inference

We begin by noting that the ELBO can be rewritten in terms of the KL divergence between the prior and approximate posterior,

$$\mathcal{L} = \textstyle\sum_i \underset{\mathrm{Q}(\mathbf{w})}{\mathbb{E}}\left[\log \mathrm{P}\left(y_i | x_i, \mathbf{w}\right)\right] - \textstyle\sum_\lambda \mathrm{D}_{\mathrm{KL}}\left(\mathrm{Q}\left(w_\lambda\right) || \, \mathrm{P}\left(w_\lambda\right)\right). \tag{3}$$

And the KL-divergence can be evaluated analytically,

$$D_{\text{KL}}\left(\text{Q}\left(w_\lambda\right)||\,\text{P}\left(w_\lambda\right)\right) = \frac{1}{2}\left(\frac{\sigma_\lambda^2+\mu_\lambda^2}{s_\lambda^2} - 1 + \log\frac{s_\lambda^2}{\sigma_\lambda^2}\right). \tag{4}$$

As such, the only term we need to approximate is the expected log-likelihood.

To evaluate this expectation, we begin by taking a second-order Taylor series expansion of the log-likelihood around the current setting of the mean parameters, $\boldsymbol{\mu}$,

$$\begin{aligned}\tfrac{1}{S}\log\text{P}\left(\{y_i\}_{i\in\mathcal{S}_j}|\{x_i\}_{i\in\mathcal{S}_j},\mathbf{w}\right) &= \\ &\log\text{P}\left(\{y_i\}_{i\in\mathcal{S}_j}|\{x_i\}_{i\in\mathcal{S}_j},\mathbf{w}{=}\mathbf{m}\right) + \mathbf{g}_j^T\left(\mathbf{w}-\boldsymbol{\mu}\right) + \tfrac{1}{2}\left(\mathbf{w}-\boldsymbol{\mu}\right)^T\mathbf{H}\left(\mathbf{w}-\boldsymbol{\mu}\right)\end{aligned} \tag{5}$$

where

$$g_{j;\lambda} = \frac{\partial}{\partial w_\lambda}\left[\tfrac{1}{S}\log\text{P}\left(\{y_i\}_{i\in\mathcal{S}_j}|\{x_i\}_{i\in\mathcal{S}_j},\mathbf{w}\right)\right] \qquad H_{\lambda,\nu} = \frac{\sum_i \log\text{P}\left(y_i|x_i,\mathbf{w}\right)}{\partial w_\lambda \partial w_\nu} \tag{6}$$

Now we consider the expectation of each of these terms under the approximate posterior, $\text{Q}\left(\mathbf{w}\right)$. The first term is constant and independent of $\mathbf{w}$. The second (linear) term is zero, because the expectation of $\left(\mathbf{w}-\mathbf{m}\right)$ under the approximate posterior is zero

$$\underset{\text{Q}(\mathbf{w})}{\mathbb{E}}\left[\mathbf{g}_j^T\left(\mathbf{w}-\mathbf{m}\right)\right] = \mathbf{g}_j^T \underset{\text{Q}(\mathbf{w})}{\mathbb{E}}\left[\left(\mathbf{w}-\mathbf{m}\right)\right] = 0 \tag{7}$$

the third (quadratic) term is difficult to evaluate because it involves $\mathbf{H}$, the $N\times N$ matrix of second derivatives, where N is the number of parameters in the model. Nonetheless, we begin by using properties of the trace, and noting that the expectation is the covariance of the approximate posterior,

$$\underset{\text{Q}(\mathbf{w})}{\mathbb{E}}\left[\tfrac{1}{2}\left(\mathbf{w}-\boldsymbol{\mu}\right)^T\mathbf{H}\left(\mathbf{w}-\boldsymbol{\mu}\right)\right] = \underset{\text{Q}(\mathbf{w})}{\mathbb{E}}\left[\tfrac{1}{2}\operatorname{Tr}\left(\mathbf{H}\left(\mathbf{w}-\boldsymbol{\mu}\right)\left(\mathbf{w}-\boldsymbol{\mu}\right)^T\right)\right] = \tfrac{1}{2}\operatorname{Tr}\left(\mathbf{H}\boldsymbol{\Sigma}\right) \tag{8}$$

writing the trace in index notation, and substituting for the (diagonal) posterior covariance, $\boldsymbol{\Sigma}$,

$$\tfrac{1}{2}\operatorname{Tr}\left(\mathbf{H}\boldsymbol{\Sigma}\right) = \tfrac{1}{2}\sum_{\lambda\nu} H_{\lambda\nu}\Sigma_{\lambda\nu} = \tfrac{1}{2}\sum_{\lambda} H_{\lambda\lambda}\sigma_\lambda^2 \tag{9}$$

Evaluating this term requires the diagonal of the Hessian, and as such our methods can be used with any drop-in estimate of this Hessian. Note that this regulariser, explicitly in terms of the Hessian, can also be justified by empirical work on generalisation in neural networks (Wu et al., 2017), which found that networks with smaller Hessian norm generalised better. In our case, we use the Fisher Information (Kunstner et al., 2019), for three reasons. First, it is extremely stable (the Fisher Information matrix is always positive definite, and hence $H_{\lambda\lambda}$ is always positive). Second the FI will allow us to relate to recently published work on implicit gradient regularisation. Third, use of the FI is standard practice in a variety of work (Khan and Lin, 2017; Khan et al., 2017, 2018; Aitchison, 2018). Nonetheless, it is interesting future work to establish whether any other estimates of the diagonal of the Hessian (e.g. from Dangel et al., 2019) improve performance. The Fisher Information identity allows us to approximate the Hessian,

$$\mathbf{H} \approx -S^2\sum_j \mathbf{g}_j\mathbf{g}_j^T \qquad\qquad H_{\lambda\lambda} \approx -S^2\sum_j g_{j;\lambda}^2 \tag{10}$$

Note that the S^2 arises because we defined $\mathbf{g}_j$ as the average gradient for the minibatch Eq. 6, whereas the Fisher Information requires the raw log probability, which is formed by the sum. Substituting this approximation of the diagonal of the Hessian we can write the quadratic term in the Taylor expansion of the expected log-likelihood as,

$$\underset{\mathrm{Q}(\mathbf{w})}{\mathbb{E}}\left[\tfrac{1}{2}\left(\mathbf{w}-\boldsymbol{\mu}\right)^T \mathbf{H}\left(\mathbf{w}-\boldsymbol{\mu}\right)\right] \approx -\tfrac{S^2}{2}\textstyle\sum_j\sum_\lambda \sigma_\lambda^2 g_{j;\lambda}^2. \tag{11}$$

Thus, our deterministic approximation to the expected log-likelihood is,

$$\underset{\mathrm{Q}(\mathbf{w})}{\mathbb{E}}\left[\log \mathrm{P}\left(\{y_i\}_{i\in\mathcal{S}_j}|\{x_i\}_{i\in\mathcal{S}_j},\mathbf{w}\right)\right] \approx \log \mathrm{P}\left(\{y_i\}_{i\in\mathcal{S}_j}|\{x_i\}_{i\in\mathcal{S}_j},\mathbf{w}{=}\mathbf{m}\right) - \tfrac{S^2}{2}\textstyle\sum_\lambda \sigma_\lambda^2 g_{j;\lambda}^2 \tag{12}$$

Critically, this quantity is a sum over datapoints, and we can therefore use minibatches of data to give unbiased estimates of the objective in stochastic gradient descent.

Our full approximation to the ELBO is thus,

$$\mathcal{L} \approx \sum_j \left(\log \mathrm{P}\left(\{y_i\}_{i\in\mathcal{S}_j}|\{x_i\}_{i\in\mathcal{S}_j},\mathbf{w}{=}\mathbf{m}\right) - \tfrac{S^2}{2}\textstyle\sum_\lambda \sigma_\lambda^2 g_{j;\lambda}^2\right) - \frac{1}{2}\sum_\lambda \left(\frac{\sigma_\lambda^2+\mu_\lambda^2}{s_\lambda^2} - 1 + \log\frac{s_\lambda^2}{\sigma_\lambda^2}\right). \tag{13}$$

We now have to be extremely careful when using minibatch estimates of the gradient. The easiest approach is to consider x_i and y_i as minibatches, where there are M minibatches, each with S examples in the full dataset. In that case, the objective for one minibatch can be written

$$\mathcal{L}_j \approx \tfrac{1}{S}\log \mathrm{P}\left(y_i|x_i,\mathbf{w}{=}\mathbf{m}\right) - \tfrac{S}{2}\textstyle\sum_\lambda \sigma_\lambda^2 g_{i;\lambda}^2 - \displaystyle\frac{1}{2P}\sum_\lambda \left(\frac{\sigma_\lambda^2+\mu_\lambda^2}{s_\lambda^2} - 1 + \log\frac{s_\lambda^2}{\sigma_\lambda^2}\right). \tag{14}$$

where, remember, S is the minibatch size and P is the total number of datapoints in the full dataset.

3.2. Connections to SGD

If we set the approximate posterior to that suggested by the stationary distribution of SGD,

$$\mathrm{Q}\left(\mathbf{w}\right) = \mathcal{N}\left(\mathbf{w};\boldsymbol{\mu},\tfrac{\eta}{2S}\mathbf{I}\right) \tag{15}$$

and we use fixed prior variance, s_λ^2, then the ELBO simplifies further, simply including a log-likelihood, a squared-gradient regulariser and weight-decay,

$$\mathcal{L}_j \approx \tfrac{1}{S}\log \mathrm{P}\left(\{y_i\}_{i\in\mathcal{S}_j}|\{x_i\}_{i\in\mathcal{S}_j},\mathbf{w}{=}\mathbf{m}\right) - \tfrac{\eta}{4}\textstyle\sum_\lambda g_{j;\lambda}^2 - \tfrac{1}{2P}\textstyle\sum_\lambda \tfrac{\mu_\lambda^2}{s_\lambda^2} + \mathrm{const.} \tag{16}$$

3.3. Network architecture

Importantly, here the regulariser is the squared gradient of the loss with respect to the parameters. As such, computing the loss implicitly involves a second-derivative of the loss, and we therefore should not use piecewise linear activation functions such as ReLU, which have pathological second derivatives. Instead, we used a softplus activation function, but any activation with well-behaved second derivatives is admissible.

4. Results

We trained a PreactResNets-18 (He et al., 2016b) on CIFAR-10 with fixed approximate posteriors inspired by SGD. We used the Adam optimizer (Kingma and Ba, 2014), with an initial learning rate of 1E-4, which decreased by a factor of 10 after 100 epochs and a batch size of 128 with all the other optimizer parameters set to their default values. We have also run experiments whereby Beta-annealing is used with $\beta = 0.1$ (Huang et al., 2018). We compared the performance of adding noise to the parameters using "MCVI" (i.e. using samples to evaluate the expectation in Eq. 18) against using our approximate ELBO, and against MAP inference. We found that our approximate ELBO gave superior performance both to MCVI and MAP. We also tried MCVI for four times as many epochs ("rescaled"), to compensate for the additional compute required to backpropagate gradients. Remarkably, this actually performed worse, than the standard settings of MCVI and we hypothesise that the additional training time allows it to overfit more strongly. Our approximate ELBO displayed optimal performance with a variance of around $\sigma^2 = 10^{-4}$ or 10^{-5} (Fig. 1). To understand whether this is sensible, in comparison with SGD learning rates, we solved for the implied learning rate,

$$\sigma^2 = \tfrac{\eta}{2S} \qquad\qquad \eta = 2S\sigma^2 \tag{17}$$

if we were to use, a batch size of $S = 50$. These settings correspond to an SGD learning rate of 10^{-2} or 10^{-3}, which is a common final learning rate (e.g. if we start at a learning rate of 10^{-1} and decay the learning rate by a factor of 10 once or twice; He et al., 2016a,b).

In addition, our approximate ELBO can be used to learn the approximate posterior variance. We used a single scalar posterior variance for each convolutional weight matrix, which would correspond to a using a different SGD learning rate for each convolutional weight matrix. Sampling-based MCVI performed poorly, even with small initial variances (initialized to $e^{-}6$ times their prior value). In past work on MCVI it was found that achieving even 80% performance on CIFAR-10 required choosing an unusually small network (Ober and Aitchison, 2020) with 32 channels in all layers. Here by contrast, we used a standard-size PreActResNet-18 with 64 channels in the first layers, increasing to 512 channels in the final layers. Remarkably, our approximate ELBO in these larger standard networks was gave superior performance even to MCVI in networks specifically tuned for variational inference (Table 1).

5. Related work

Recent work also showed that gradient-descent implies an implicit squared-gradient regularisation (Barrett and Dherin, 2020) with a very similar dependence of the strength of regularisation on the SGD step-size. Remarkably, this arose despite their taking a radically different approach. In particular, they showed that squared-gradient regularisation can arise through errors when using finite step size Euler integration, as compared to following the true underlying gradient flow. However, this approach was derived for gradient descent, and does not take into account the fundamental stochasticity of *stochastic* gradient descent. The emergence of such similar results from fundamentally very different analytical approaches would suggest that gradient regularisation is a fundamental component of the implicit regularisation inherent in SGD.

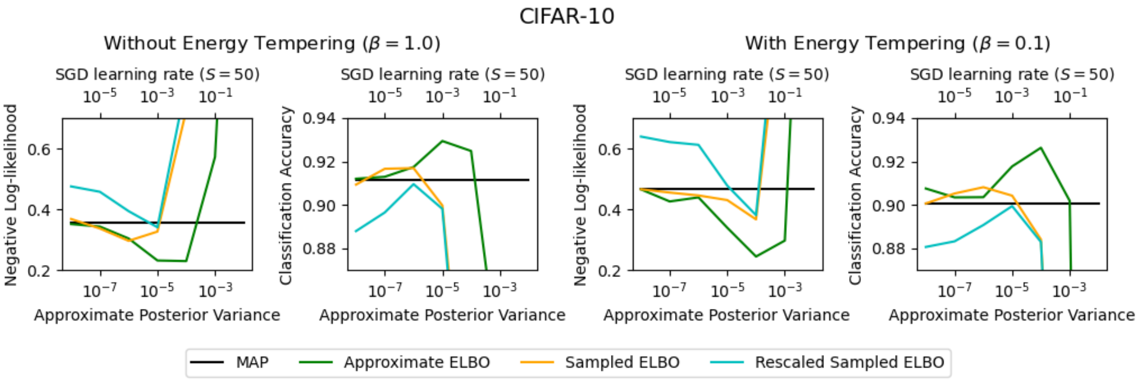

Figure 1: Training a PreactResNet-18 on CIFAR-10 with an fixed-variance approximate posterior, to mirror SGD. The posterior variance is shown on the lower x-axis and the corresponding value for the SGD learning rate is shown on the upper x-axis. "Sampled ELBO" corresponds to 200 training epochs and "Rescaled sampled ELBO" corresponds to 800 training epochs.

Table 1: Performance of a PreactResNet-18 with a single scalar learned variance (σ^2) and a learned diagonal covariance ($\mathbf{\Sigma}$) respectively for each convolutional weight matrix.

β	Covariance		neg. test log-like.	accuracy (%)
1.0	—	MAP	0.36	91.13
	$\sigma^2\mathbf{I}$	Approx. ELBO	**0.44**	**87.21**
		Sampled ELBO	1.67	46.32
	$\mathbf{\Sigma}$	Approx. ELBO	**0.45**	**87.25**
		Sampled ELBO	1.00	72.22
		Ober and Aitchison (2020)	0.66	77.65
0.1	—	MAP	0.47	90.05
	$\sigma^2\mathbf{I}$	Approx. ELBO	**0.21**	**92.74**
		Sampled ELBO	0.65	79.45
	$\mathbf{\Sigma}$	Approx. ELBO	**0.21**	**92.93**
		Sampled ELBO	0.38	91.07

6. Conclusions

We showed that it is possible to use a second-order Taylor expansion to compute a deterministic approximation of the ELBO in Bayesian neural networks. While any drop-in estimate of the diagonal of the Hessian could be used, here we consider the squared gradient (via the Fisher Information), as this allows efficient optimization and allows us to connect to gradient regularisation (Barrett and Dherin, 2020). Further, we connect to SGD by noting that the stationary distribution for SGD is *isotropic* Gaussian (Mandt et al., 2017). This approximate ELBO gives superior performance to unbiased sampling-based estimation of the ELBO, and to MAP inference. Finally, our framework makes a prediction: that the optimal posterior variance for the ELBO corresponds to standard settings of the SGD learning rate. Remarkably, we found that this prediction held, with the optimal posterior variance in ResNets on CIFAR-10 being 10^{-4} or 10^{-5} corresponds to standard values for the final learning rate in SGD used in these models.

References

Laurence Aitchison. Bayesian filtering unifies adaptive and non-adaptive neural network optimization methods. *arXiv preprint arXiv:1807.07540*, 2018.

Filippo Amato, Alberto López, Eladia Maria Peña-Méndez, Petr Vanhara, Ales Hampl, and Josef Havel. Artificial neural networks in medical diagnosis. *J Appl Biomed*, 11:47–58, 2013.

David GT Barrett and Benoit Dherin. Implicit gradient regularization. *arXiv preprint arXiv:2009.11162*, 2020.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424*, 2015.

Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

Felix Dangel, Frederik Kunstner, and Philipp Hennig. Backpack: Packing more into backprop. *arXiv preprint arXiv:1912.10985*, 2019.

Pascal Germain, Francis Bach, Alexandre Lacoste, and Simon Lacoste-Julien. Pac-bayesian theory meets bayesian inference. In *Advances in Neural Information Processing Systems*, pages 1884–1892, 2016.

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016b.

Chin-Wei Huang, Shawn Tan, Alexandre Lacoste, and Aaron C Courville. Improving explorability in variational inference with annealed variational objectives. In *Advances in Neural Information Processing Systems*, pages 9701–9711, 2018.

Nitish Shirish Keskar and Richard Socher. Improving generalization performance by switching from adam to sgd. *arXiv preprint arXiv:1712.07628*, 2017.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

Mohammad Emtiyaz Khan and Wu Lin. Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models. *arXiv preprint arXiv:1703.04265*, 2017.

Mohammad Emtiyaz Khan, Zuozhu Liu, Voot Tangkaratt, and Yarin Gal. Vprop: Variational inference using rmsprop. *arXiv preprint arXiv:1712.01038*, 2017.

Mohammad Emtiyaz Khan, Didrik Nielsen, Voot Tangkaratt, Wu Lin, Yarin Gal, and Akash Srivastava. Fast and scalable bayesian deep learning by weight-perturbation in adam. *arXiv preprint arXiv:1806.04854*, 2018.

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

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Frederik Kunstner, Philipp Hennig, and Lukas Balles. Limitations of the empirical fisher approximation for natural gradient descent. In *Advances in Neural Information Processing Systems*, pages 4156–4167, 2019.

Deren Lei, Zichen Sun, Yijun Xiao, and William Yang Wang. Implicit regularization of stochastic gradient descent in natural language processing: Observations and implications. *arXiv preprint arXiv:1811.00659*, 2018.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Stephan Mandt, Matthew D Hoffman, and David M Blei. Stochastic gradient descent as approximate bayesian inference. *The Journal of Machine Learning Research*, 18(1): 4873–4907, 2017.

Rowan McAllister, Yarin Gal, Alex Kendall, Mark Van Der Wilk, Amar Shah, Roberto Cipolla, and Adrian Weller. Concrete problems for autonomous vehicle safety: Advantages of bayesian deep learning. In *International Joint Conferences on Artificial Intelligence, Inc.*, 2017.

Sebastian W. Ober and Laurence Aitchison. Global inducing point variational posteriors for bayesian neural networks and deep gaussian processes, 2020.

Tom Rainforth, Adam R Kosiorek, Tuan Anh Le, Chris J Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. *arXiv preprint arXiv:1802.04537*, 2018.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Omar Rivasplata, Vikram M Tankasali, and Csaba Szepesvari. Pac-bayes with backprop. *arXiv preprint arXiv:1908.07380*, 2019.

Daniel A Roberts. Sgd implicitly regularizes generalization error. In *NIPS 2018 Workshop*, 2018.

Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688, 2011.

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. In *Advances in neural information processing systems*, pages 4148–4158, 2017.

Anqi Wu, Sebastian Nowozin, Edward Meeds, Richard E Turner, José Miguel Hernández-Lobato, and Alexander L Gaunt. Deterministic variational inference for robust bayesian neural networks. *arXiv preprint arXiv:1810.03958*, 2018.

Lei Wu, Zhanxing Zhu, et al. Towards understanding generalization of deep learning: Perspective of loss landscapes. *arXiv preprint arXiv:1706.10239*, 2017.

Appendix A. Variational Inference for Bayesian Neural Networks

We optimize the evidence lower bound objective with respect to parameters of the variational posterior,

$$\mathcal{L} = \mathop{\mathbb{E}}_{\mathrm{Q}(\mathbf{w})}\left[\textstyle\sum_i \log \mathrm{P}\left(y_i|x_i, \mathbf{w}\right) + \sum_\lambda \log \frac{\log \mathrm{P}\left(w_\lambda\right)}{\log \mathrm{Q}\left(w_\lambda\right)}\right] \tag{18}$$

where x_i is the training inputs and y_i is the training outputs. Unfortunately, we need to optimize this expectation with respect to parameters of $\mathrm{Q}(\mathbf{w})$, the distribution over which the expectation is taken. We therefore use the reparameterisation trick (Kingma and Welling, 2013; Rezende et al., 2014) — we write $\mathbf{w}$ in terms of $\boldsymbol{\epsilon}$,

$$w_\lambda(\epsilon_\lambda) = \mu_\lambda + \sigma_\lambda \epsilon_\lambda \tag{19}$$

where $\epsilon_\lambda \sim \mathcal{N}(0, 1)$. Thus, the ELBO can be written as an expectation over $\boldsymbol{\epsilon}$,

$$\mathcal{L} = \mathop{\mathbb{E}}_{\boldsymbol{\epsilon}}\left[\textstyle\sum_i \log \mathrm{P}\left(y_i|x_i, \mathbf{w}(\boldsymbol{\epsilon})\right) + \sum_\lambda \log \frac{\log \mathrm{P}\left(w_\lambda(\epsilon_\lambda)\right)}{\log \mathrm{Q}\left(w_\lambda(\epsilon_\lambda)\right)}\right]. \tag{20}$$

However, exactly evaluating this objective remains intractable, as it still involves a high-dimensional expectation. Instead, we typically form an unbiased estimator, by drawing a few (often one) sample of $\mathbf{w}$ or $\boldsymbol{\epsilon}$. While this approach can be effective, it gives high-variance estimates of the gradient, which can cause slow-learning (Wu et al., 2018; Rainforth et al., 2018).

Appendix B. The Stationary Distribution of SGD

We then sought to relate these gradient regularisers back to the excellent empirical performance of SGD. In particular, we looked to work on the stationary distribution of SGD, which noted that under quadratic losses functions, SGD samples an *isotropic* Gaussian (i.e. with covariance proportional to the identity matrix). In particular, consider a loss function which is locally closely approximated by a quadratic. Without loss of generality, we consider a mode at $\mathbf{w} = \mathbf{0}$,

$$\log \mathrm{P}\left(\{y_i\}_{i=1}^P | \{x_i\}_{i=1}^P, \mathbf{w}\right) = -\tfrac{P}{2} \mathbf{w}^T \mathbf{H} \mathbf{w} + \text{const}\,. \tag{21}$$

where P is the total number of datapoints. Typically, the objective used in SGD is the loss for a minibatch of size S with indices $\mathcal{S}$. Following Mandt et al. (2017), we use the Fisher Information to identify the noise in the minibatch gradient,

$$\frac{\partial}{\partial \mathbf{w}} \left[\tfrac{1}{S} \log \mathrm{P}\left(\{y_i\}_{i\in\mathcal{S}_j} | \{x_i\}_{i\in\mathcal{S}_j}, \mathbf{w}\right)\right] = -\mathbf{H}\mathbf{w} + \tfrac{1}{\sqrt{S}} \mathbf{H}^{1/2} \boldsymbol{\xi}(t), \tag{22}$$

where $\boldsymbol{\xi}(t)$ is sampled from a standard IID Gaussian. For SGD, this gradient is multiplied by a learning rate, η,

$$\mathbf{w}(t+1) = \mathbf{w}(t) - \eta \mathbf{H}\mathbf{w}(t) + \tfrac{\eta}{\sqrt{S}} \mathbf{H}^{1/2} \boldsymbol{\xi}(t), \tag{23}$$

This is an multivariate Gaussian autoregressive process, so we can solve for the stationary distribution of the weights. In particular, we note that the covariance at time $t+1$ is

$$\begin{aligned}
\mathbb{C}\left[\mathbf{w}\left(t+1\right)\right] &= \mathbb{E}\left[\left(\mathbf{w}(t) - \eta\mathbf{H}\mathbf{w}(t) - \tfrac{\eta}{\sqrt{S}}\mathbf{H}^{1/2}\boldsymbol{\xi}(t)\right)^T \left(\mathbf{w}(t) - \eta\mathbf{H}\mathbf{w}(t) - \tfrac{\eta}{\sqrt{S}}\mathbf{H}^{1/2}\boldsymbol{\xi}(t)\right)\right] \\
\mathbb{C}\left[\mathbf{w}\left(t+1\right)\right] &= \left(\mathbf{I} - \eta\mathbf{H}\right)^T \mathbb{C}\left[\mathbf{w}(t)\right]\left(\mathbf{I} - \eta\mathbf{H}\right) + \tfrac{\eta^2}{S}\mathbf{H} \qquad (24)
\end{aligned}$$

Following Mandt et al. (2017), when the learning rate is small, the quadratic term can be neglected.

$$\mathbb{C}\left[\mathbf{w}\left(t+1\right)\right] \approx \mathbb{C}\left[\mathbf{w}(t)\right] - \eta\mathbf{H}\,\mathbb{C}\left[\mathbf{w}(t)\right] - \eta\,\mathbb{C}\left[\mathbf{w}(t)\right]\mathbf{H} + \tfrac{\eta^2}{S}\mathbf{H} \tag{25}$$

We then solve for steady-state in which $\boldsymbol{\Sigma} = \mathbb{C}\left[\mathbf{w}(t+1)\right] = \mathbb{C}\left[\mathbf{w}(t)\right]$,

$$0 \approx -\eta\left(\mathbf{H}^T\boldsymbol{\Sigma} + \boldsymbol{\Sigma}\mathbf{H}\right) + \tfrac{\eta^2}{S}\mathbf{H} \tag{26}$$

so,

$$\boldsymbol{\Sigma} \approx \tfrac{\eta}{2S}\mathbf{I}. \tag{27}$$

We therefore considered a family of approximate posteriors matching these stationary distributions, by setting all variances, σ_λ^2, to a single, fixed value, $\frac{\eta}{2S}$.