On the Performance of Direct Loss Minimization for Bayesian Neural Networks

Yadi Wei
Indiana University
Bloomington, IN
weiyadi@iu.edu

Roni Khardon
Indiana University
Bloomington, IN
rkhardon@iu.edu

Abstract

Direct Loss Minimization (DLM) has been proposed as a pseudo-Bayesian method motivated as regularized loss minimization. Compared to variational inference, it replaces the loss term in the evidence lower bound (ELBO) with the predictive log loss, which is the same loss function used in evaluation. A number of theoretical and empirical results in prior work suggest that DLM can significantly improve over ELBO optimization for some models. However, as we point out in this paper, this is not the case for Bayesian neural networks (BNNs). The paper explores the practical performance of DLM for BNN, the reasons for its failure and its relationship to optimizing the ELBO, uncovering some interesting facts about both algorithms.

1 Introduction

One of the main goals of probabilistic machine learning is to develop algorithms that can make well calibrated probabilistic predictions. From the frequentist view, we need to find a *single* set of parameters that best fits the data; while from a Bayesian view, we specify a prior distribution on the parameters, calculate the posterior, and then use the posterior to predict on new data. Let $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^N$ be the dataset sampled i.i.d. from distribution $\mathcal{D}$. From now on, we use superscript with parentheses to denote the i-th instance. Let θ denote the parameters. The frequentist method chooses one best set of parameters θ^* and makes predictions on new data x^* such that $p(y^*|x^*) = p(y^*|\theta^*, x^*)$. Bayesian methods specify a prior $p(\theta)$, calculate the posterior

$$p(\theta|D) \propto p(\theta)p(D|\theta),$$

and make predictions on new data x^* as

$$p(y^*|x^*) = \mathbb{E}_{p(\theta|D)}[p(y^*|\theta, x^*)].$$

For simple models, the posterior can be computed analytically. But for complicated models, the posterior becomes intractable. One solution is to get samples from true posterior and calculate the objective, for example using MCMC. Another line of work aims to find a distribution q from an analytical distribution family $\mathcal{Q}$ that is closest to the posterior. When making prediction on new data x^*, we marginalize over q, that is, $p(y^*|x^*) = \mathbb{E}_{q(\theta)}[p(y^*|\theta, x^*)]$. A typical example is variational inference, which tries to minimize the KL divergence between the variational distribution and the true posterior:

$$\begin{aligned} q^*(\theta) &= \arg\min_{q\in\mathcal{Q}} \mathrm{KL}(q(\theta)||p(\theta|D)) \\ &= \arg\min_{q\in\mathcal{Q}} \mathbb{E}_{q(\theta)}[\log q(\theta) - \log p(\theta, D)] + \log p(D) \\ &= \arg\min_{q\in\mathcal{Q}} \sum_i \mathbb{E}_{q(\theta)}[-\log p(y^{(i)}|\theta, x^{(i)})] + \mathrm{KL}(q(\theta)||p(\theta)). \end{aligned} \tag{1}$$

I Can't Believe It's Not Better Workshop at NeurIPS 2022.

The last line is the negation of the Evidence Lower Bound (ELBO). The most common measure to evaluate the quality of predictions is negative log-loss (NLL):

$$l(q, (x^*, y^*)) = -\log \mathbb{E}_{q(\theta)}[p(y^*|\theta, x^*)]. \tag{2}$$

We use $l_{\text{test}}(q)$ to denote the averaged NLL on test set. We optimize eq (1) but hope to get lower NLL, i.e. eq (2). This suggests a discrepancy. If we care about the NLL, why not directly optimize NLL (2)? From this perspective the KL term in eq (1) is seen as a regularizer to prevent overfitting. This motivates the idea of Direct Loss Minimization (DLM) which has been studied by multiple authors:

$$q^{(\eta)}_{\text{DLM}}(\theta) = \arg\min_{q \in \mathcal{Q}} \sum_i -\log \mathbb{E}_{q(\theta)}[p(y^{(i)}|\theta, x^{(i)})] + \eta \text{KL}(q(\theta)||p(\theta)). \tag{3}$$

Notice that by Jensen's inequality,

$$l_{\text{dlm}}(q, (x, y)) = -\log \mathbb{E}_{q(\theta)}[p(y|\theta, x)] \leq l_{\text{elbo}}(q, (x, y)) = -\mathbb{E}_{q(\theta)}[\log p(y|\theta, x)]$$

so that l_{elbo} can be seen as a surrogate loss for the log loss which is used during training. But DLM optimizes the desired objective and the idea of DLM can be applied on various loss functions. We can replace the first term in eq (3) with any loss functions we are interested in. Similarly, the regularizer can be chosen among several options. However, in this paper we focus on the choice given above.

In practice, it is common to use a hyperparameter η to tune the KL regularizer, as in eq (3). In addition, when it is hard to compute the expectations analytically we can use Monte Carlo samples to approximate them. With these modifications the objectives in eq (1) and eq (3)) become:

$$\bar{q}^{(\eta,M)}_{\text{ELBO}}(\theta) = \arg\min_{q \in \mathcal{Q}} \sum_i \frac{1}{M} \sum_{m=1}^{M} [-\log p(y^{(i)}|\theta^{(m)}, x^{(i)})] + \eta \text{KL}(q(\theta)||p(\theta)), \quad \theta^{(m)} \sim q(\theta); \tag{4}$$

$$\bar{q}^{(\eta,M)}_{\text{DLM}}(\theta) = \arg\min_{q \in \mathcal{Q}} \sum_i -\log \frac{1}{M} \sum_{m=1}^{M} p(y^{(i)}|\theta^{(m)}, x^{(i)}) + \eta \text{KL}(q(\theta)||p(\theta)), \quad \theta^{(m)} \sim q(\theta). \tag{5}$$

Notice that eq (4) is an unbiased estimate of eq (1) while eq (5) is a *biased* estimate of eq (3).

2 Theoretical Motivation of DLM

Prior work motivates the use of DLM from a theoretical perspective. In this section we review some of these results. Specifically, Sheth and Khardon [2019] provide risk bounds for several variants of DLM. Here we focus on one result (see their appendix B) that uses a bounded optimization view. Suppose we restrict our distribution family $\mathcal{Q}$ to $\mathcal{Q}_A = \{q \in \mathcal{Q} \text{ s.t. } \text{KL}(q, p) \leq A\}$, where p is the prior distribution over θ, and we perform empirical risk minimization (ERM) on $\mathcal{Q}_A$:

$$q^{(A)}_{\text{ERM}}(\theta) = \arg\min_{q \in \mathcal{Q}_A} \sum_i -\log \mathbb{E}_{q(\theta)}[p(y^{(i)}|\theta, x^{(i)})]. \tag{6}$$

Then with probability $1 - \delta$ over the choice of the dataset D, for all $q \in \mathcal{Q}_A$,

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[-\log\left(\mathbb{E}_{q^{(A)}_{\text{ERM}}(\theta)} p(y|\theta, x)\right)\right] \leq \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[-\log(\mathbb{E}_{q(\theta)}(p(y|\theta, x)))\right] + \mathcal{O}\left(\sqrt{\frac{A}{N}} + \sqrt{\frac{\log\frac{1}{\delta}}{N}}\right). \tag{7}$$

This result holds when the log-loss is bounded, which can be achieved by replacing the log loss with

$$\log^{(a)} p = \log((1 - a)p + a) \tag{8}$$

or by further bounding the parameter space of distribution family $\mathcal{Q}$. The following proposition (proof in [Wei and Khardon, 2022]) observes that while the above holds for eq (6) we can obtain similar risk bounds for eq (3). One can further extend eq (7) into a data dependent bound as in Theorem 10 in [Meir and Zhang, 2003]. Therefore DLM as motivated above enjoys some theoretical support.

Proposition 1. *Let $A_\eta = KL(q^{(\eta)}_{DLM}||p)$. Then the solution of eq* (3)*, i.e., $q^{(\eta)}_{DLM}$, is also the solution of eq* (6) *with $A = A_\eta$.*

A second theoretical perspective is given by the recent work of Morningstar et al. [2022]. This work presents a PAC-Bayes bound called PAC^m bound. With probability $1-\delta$, for any $q \in \mathcal{Q}$,

$$\begin{aligned}
&\mathbb{E}_{(x,y)\sim\mathcal{D}}[-\log \mathbb{E}_{q(\theta)}[p(y|\theta,x)]] \\
\leq &-\frac{1}{N}\sum_i \mathbb{E}_{q(\theta^M)}\left[\log\left(\frac{1}{M}\sum_m p\left(y^{(i)}|x^{(i)},\theta^{(m)}\right)\right)\right] + \frac{1}{\eta N}\mathrm{KL}(q||p) \\
&+ \psi(\mathcal{D},\eta,M,N,p,\delta) + \frac{1}{\eta MN}\log\frac{1}{\delta},
\end{aligned} \tag{9}$$

where

$$\psi(\mathcal{D},\eta,M,N,p,\delta) = \frac{1}{\eta MN}\log \mathbb{E}_{D\sim\mathcal{D}^N}\mathbb{E}_{p(\theta^{(1:m)})}\left[\exp\left(\eta NM\cdot\Delta\left(D,\theta^{(1:M)}\right)\right)\right],$$

$$\Delta(D,\theta^{(1:M)}) = \frac{1}{N}\sum_i \log\left(\frac{1}{M}\sum_m p(y^{(i)}|x^{(i)},\theta^{(m)})\right) - \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\log\left(\frac{1}{M}\sum_m p(y|x,\theta^{(m)})\right)\right].$$

The PAC^m algorithm minimizes the right-hand-side of (9) to calculate its solution which we denote as $\bar{q}^{(\eta)}_{\mathrm{DLM}}$. For the corresponding algorithm note that the terms on the last line of (9) do not depend on $\bar{q}^{(\eta)}_{\mathrm{DLM}}$ and can be omitted in the optimization. This is different from previous analysis [Sheth and Khardon, 2017, 2019], that uses the predictive loss

$$\begin{aligned}
-\log \mathbb{E}_{q(\theta)}[p(y|x,\theta)] &= -\log \mathbb{E}_{q(\theta^{(1:M)})}\left[\frac{1}{M}\sum_m p(y|x,\theta^{(m)})\right] \\
&\leq -\mathbb{E}_{q(\theta^{(1:M)})}\left[\log\left(\frac{1}{M}\sum_m p\left(y|x,\theta^{(m)}\right)\right)\right]
\end{aligned} \tag{10}$$

for the data-dependent upper bound. When the outside expectation in (9) is implemented with a single multi-sample from $q()$, which is the case in [Morningstar et al., 2022], eq (9) leads to the same implementation as eq (5) and (7). Theoretically, loss term in bound (7) is lower but it requires $\bar{q}^{(\eta)}_{\mathrm{DLM}}$ to converge to $q^{(\eta)}_{\mathrm{DLM}}$ to guarantee the performance (see related analysis by Wei et al. [2021]), while the bound (9) directly guarantees the performance of $\bar{q}^{(\eta)}_{\mathrm{DLM}}$ with a higher loss as shown in eq (10).

Morningstar et al. [2022] also establishes the relationship between ELBO and DLM. Notice that with $M=1$, the right hand sides of eq (4) and eq (5) are the same. They also show that as M becomes larger, the data dependent bound when $M>1$ (i.e., the right hand side of eq (9)) is tighter than that when $M=1$, corresponding to the data dependent bound for ELBO.

3 Applications of DLM

DLM has already been applied in practice and is shown to yield good results. Sheth and Khardon [2017] apply DLM to the correlated topic model and it achieves lower predictive loss than ELBO. Jankowiak et al. [2020] explores the application of DLM in conjugate sparse Gaussian processes and Wei et al. [2021] extends this to non-conjugate sparse Gaussian processes. In those experiments, η is chosen appropriately through cross validation. For conjugate cases, where both the ELBO objective (1) and the DLM objective (3) can be computed exactly without approximation, DLM significantly outperforms ELBO; while for non-conjugate cases, Monte Carlo sampling is needed and DLM is better than or comparable to ELBO.

DLM has also been applied to BNNs. Dusenberry et al. [2020] apply both ELBO and DLM on a rank-1 parametrization of BNNs that they introduce. Their experiments show that DLM has higher NLL than ELBO, which means that DLM performs worse than ELBO. Morningstar et al. [2022] also compare ELBO and DLM on BNNs and conclude that DLM performs better than ELBO when data is misspecified, i.e. the data generating distribution is not inside the model space. To see this, they apply ELBO and DLM to independently predict pixels of the bottom half of an image given the top half, which encounters data misspecification as the pixels are not independent. When there is no evident data misspecification they show that, under the same KL value, DLM performs slightly better than ELBO. However, the converged ELBO solution may not have the same KL value as the converged DLM solution. So the experiment is not sufficient to show that the converged DLM solution performs better than the converged ELBO solution.

4 DLM in Bayesian Neural Networks

In contrast with the positive evidence, our work found that DLM does not perform as well for Bayesian neural networks (BNNs). In BNN, θ represents the weights of the neural network and we use eq (4) and eq (5) with a mean-field diagonal Gaussian variational distribution $q(\theta_j) = \mathcal{N}(\mu_j, \sigma_j^2)$ and μ and σ^2 are vectors of the same dimension as the parameter space. BNNs normally have millions of parameters, so the KL-divergence term can be very high and using a high value of $\eta = 1$ leads to poor performance, hence we fix $\eta = 0.1$. Following Wilson et al. [2022], we set the prior variance to 0.05. We experimented with two neural network structures, AlexNet [Krizhevsky et al., 2012] and PreResNet [He et al., 2016] with depth 20 on four datasets, CIFAR10, CIFAR100, STL10, SVHN. For all experiments, we set the batch size to 512, use the Adam optimizer with learning rate 0.001 and train for 500 epochs. We set $M = 5$ in eq (4) and eq (5) for training and $M = 10$ for evaluation. Most combinations of datasets and structures have similar results (see Figure 1) and behave similarly during our exploration, so we only show the detailed exploration of using AlexNet on CIFAR10 in the main body of the paper and discuss exceptions in the [Wei and Khardon, 2022]. Our code is available on `https://github.com/weiyadi/dlm_bnn`. To simplify the presentation, we use upper case "ELBO" and "DLM" to indicate the solution trained with ELBO and DLM loss respectively, and use lower case "elbo" and "dlm" to denote the corresponding loss functions.

Observation 1. *ELBO performs better than DLM.*

DLM is worse than ELBO in most experiments. A summary over all experiments is shown in Figure 1 and a concrete learning curve is shown in Figure 2a. In a few cases DLM has similar performance but it does not outperform ELBO in any of the experiments. The range of test losses in the experiments is up to 2, so the differences shown are significant.

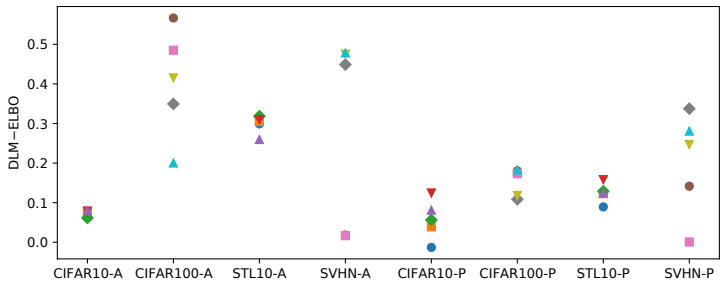


Figure 1: Comparison of ELBO and DLM in all experiments, "-A" means on AlexNet and "-P" means on PreResNet20. y-axis is $l_{\text{test}}(q_{\text{DLM}}) - l_{\text{test}}(q_{\text{ELBO}})$, and a positive value indicates that DLM performs worse than ELBO. Each point represents an independent run with random initialization.

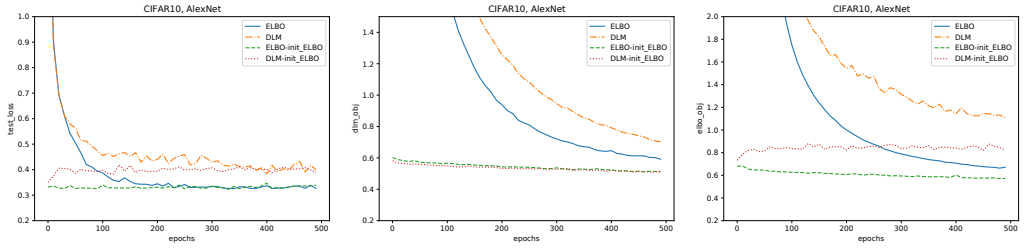
To better understand the behavior of ELBO and DLM, we also compute some quantities during the training, including elbo objective, dlm objective and KL divergence, and see how they change. We repeat with different seeds and the quantities behave similarly regardless of the random seed we choose. To our surprise, we observe:

Observation 2. *ELBO appears to optimize the dlm objective better than the DLM algorithm.*

As shown in Figure 2b, which depicts how the dlm loss changes during training, ELBO (blue solid line) is below DLM (orange dashdot line). At the same time, ELBO optimizes its own objective, elbo objective, better than DLM, as shown in Figure 2c. One might suspect that the reason for this is that the dlm objective is likely to get stuck in local optima. Thus if we initialize DLM with a good starting point, then DLM may be improved. To test this, we initialize DLM with ELBO solution and continue to train with dlm objective and we denote this solution as DLM-init_ELBO. For comparison we also have ELBO-init_ELBO which is initialized with ELBO and then continue to train with elbo objective. However, DLM is not improved with ELBO initialization and it even makes ELBO worse, as shown in Figure 2a. The increase of test loss for DLM shows that DLM is not stuck in local optima by accident, but is inherently worse than ELBO.

Observation 3. *The failure of DLM is not due to local optima.*

The good news is that with ELBO initialization, DLM optimizes the dlm objective slightly better than ELBO, as shown in Figure 2b. We note that this does not happen for every experiment. In Figure C.1c in the [Wei and Khardon, 2022], DLM still optimizes dlm loss worse than ELBO even with ELBO initialization.



(a) test loss(↓) (b) Trajectory of dlm loss (c) Trajectory of elbo loss

Figure 2: Comparison of ELBO and DLM with/without initialization

All these abnormalities lead us to further explore the structure of elbo and dlm losses. Motivated by Garipov et al. [2018], we create a path from ELBO to DLM, i.e. $\mu = (1-\alpha)\mu_{\text{ELBO}} + \alpha\mu_{\text{DLM}}, \sigma^2 = (1-\alpha)\sigma^2_{\text{ELBO}} + \alpha\sigma^2_{\text{DLM}}$, and then evaluate the elbo objective (with/without KL), dlm objective (with/without KL) and test loss on (μ, σ^2). It is clear that ELBO corresponds to $\alpha = 0$ and DLM corresponds to $\alpha = 1$. From the loss surface plotted in Figure 3a, we confirm Observation 1 and 2 (note that the dlm objective is lower at $\alpha = 0$). Figure 3b plots the path from ELBO to DLM-init_ELBO, and we can see how optimizing with dlm loss will change these loss functions.

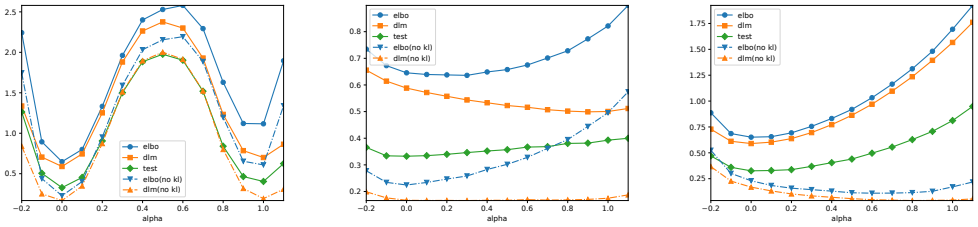
Observation 4. *The elbo objective with $\eta = 0.1$ is better aligned with test loss than the dlm objective, which indicates that the elbo objective generalizes better.*

Figure 3b shows that as α increases, the dlm objective decreases, but both the elbo objective and the test loss increase. Figure C.2b in the [Wei and Khardon, 2022] shows a different situation (on CIFAR100 with AlexNet) where the dlm objective also increases, i.e., it is also aligned. But in our experiments the elbo objective and the test loss are always aligned in our experiments. Observation 4 also explains the abnormality in Figure C.1c, in which DLM-init_ELBO significantly increases the dlm objective in first few epochs. This is because we optimize the dlm objective within a batch but plot the average dlm objective value among all batches. The poor generalization of the dlm objective may cause the value evaluated on other batches to increase and the overall value increases.

We also observe from Figure 3b that the dlm objective goes down but its loss term goes up, implying that the reduction in objective is due to the KL term. The same sensitivity regarding the *tradeoff between the loss term and regularizer* appears in other cases as well. To explore this we reduce η to 0 after initializing with ELBO. Figure 4a shows that reducing η to 0 makes both ELBO and DLM perform worse than their original version but the relationship of ELBO-init-no_kl and DLM-init-no_kl still follows Observation 1. Figure 4b again shows Observation 2, i.e., that the dlm objective ($\eta = 0$) achieves lower value at ELBO-init-no_kl than at DLM-init-no_kl.

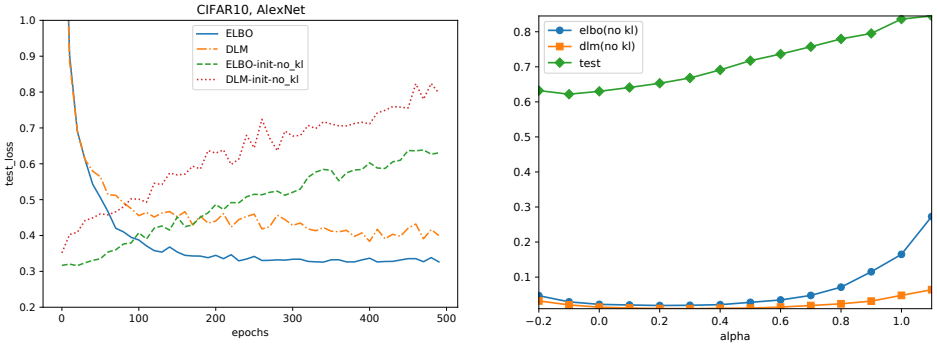
In contrast with Figure 4b, Figure 3c depicts the path between DLM-init-no_kl and the original ELBO solution (which is the best), instead of ELBO-init-no_kl. Then we can see that neither the elbo loss nor the dlm loss without KL is aligned with the test loss, indicating overfitting. From another view, the three plots in Figure 3 depict the change of loss functions along three directions from ELBO. The elbo objective with $\eta = 0.1$ is aligned with the test loss in all three cases. But we cannot find such proper η for dlm. In (a) and (c), the dlm objective with $\eta = 0.1$ is aligned with the test loss, but in (b) the dlm objective with $\eta = 0$ is aligned with the test loss. All these results support Observation 4.

In addition to the work mentioned above, we have explored bounded optimization, smoothed loss, collapsed variational inference [Tomczak et al., 2021] and empirical Bayes [Wu et al., 2019]. The first two measures aim to close the gap between theoretical analysis and real applications so that we can utilize the upper bounds to guarantee the performance of DLM. The latter two define a hierarchical model and perform inference on the prior parameters, which results in a different regularizer to replace the original KL divergence. Although these measures can sometimes improve the performance of DLM, they do not help DLM outperform ELBO. Details are in the [Wei and Khardon, 2022].



(a) ELBO and DLM (b) ELBO and DLM-init_ELBO (c) ELBO and DLM-init-no_kl

Figure 3: Loss Surface



(a) test loss (b) ELBO-init-no_kl and DLM-init-no_kl

Figure 4: Test Performance and Loss Surface for $\eta = 0$

Overall, we found that at least one of Observation 2 and 4 appears in all experiments. In cases where ELBO does not optimize the dlm objective better than DLM, Observation 4 kicks in and shows that optimizing the dlm objective cannot make the performance better; In cases where the dlm objective is aligned with the test loss, we find that ELBO optimizes the dlm objective better. Thus, none of the variants of DLM mentioned in this paper outperforms ELBO.

5 Conclusion and Future Work

Direct loss minimization has a strong motivation that we should use the same loss function in both training and testing. During training, we add a regularizer to prevent overfitting. Many theoretical results guarantee the performance of DLM optimizers. Despite its empirical success in sparse Gaussian processes, we observe that such success does not appear for Bayesian neural networks. In empirical exploration, we found that the goal of DLM is also severely challenged as ELBO optimizes dlm objective better than DLM itself. The most likely reason for this is that the dlm objective is hard to optimize for Bayeisan neural networks. Besides, DLM generalizes worse than ELBO, because elbo loss is more consistent with test loss, pointing out overfitting of the dlm objective. This relates to data misspecification as suggested in [Morningstar et al., 2022] but how to test the notion of misspecification in image classification remains unclear as the neural networks are expressive. It would be interesting to explore what distinguishes cases where DLM succeeds, such as sparse Gaussian processes, from the behavior shown in this paper. As mentioned above, we can view the elbo loss as a (potentially better behaved) surrogate loss of the true loss given by dlm. It would be interesting to explore theoretical analysis that explains differences in behavior from this perspective.

Acknowledgments and Disclosure of Funding

This work was partly supported by NSF under grant IIS-1906694. Some of the experiments in this paper were run on the Big Red computing system at Indiana University, supported in part by Lilly Endowment, Inc., through its support for the Indiana University Pervasive Technology Institute.

References

Michael W. Dusenberry, Ghassen Jerfel, Yeming Wen, Yi-An Ma, Jasper Snoek, Katherine Heller, Balaji Lakshminarayanan, and Dustin Tran. Efficient and scalable bayesian neural nets with rank-1 factors. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry Vetrov, and Andrew Gordon Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 8803–8812, Red Hook, NY, USA, 2018. Curran Associates Inc.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 630–645, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46493-0.

Martin Jankowiak, Geoff Pleiss, and Jacob R. Gardner. Parametric gaussian process regressors. In *ICML*, 2020.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Ron Meir and Tong Zhang. Generalization error bounds for bayesian mixture algorithms. *J. Mach. Learn. Res.*, 4(null):839–860, dec 2003. ISSN 1532-4435.

Warren R. Morningstar, Alex Alemi, and Joshua V. Dillon. Pacm-bayes: Narrowing the empirical risk gap in the misspecified bayesian regime. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 8270–8298. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/morningstar22a.html.

Rishit Sheth and Roni Khardon. Excess risk bounds for the Bayes risk using variational inference in latent Gaussian models. In *NIPS*, pages 5151–5161, 2017.

Rishit Sheth and Roni Khardon. Pseudo-bayesian learning via direct loss minimization with applications to sparse gaussian process models. In *Symposium on Advances in Approximate Bayesian Inference (AABI)*, 2019.

Marcin B. Tomczak, Siddharth Swaroop, Andrew Y. K. Foong, and Richard E Turner. Collapsed variational bounds for bayesian neural networks. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=ykN3tbJ0qmX.

Yadi Wei and Roni Khardon. On the performance of direct loss minimization for bayesian neural networks, 2022. URL https://arxiv.org/abs/2211.08393.

Yadi Wei, Rishit Sheth, and Roni Khardon. Direct loss minimization for sparse gaussian processes. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 2566–2574. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/wei21b.html.

Andrew Gordon Wilson, Pavel Izmailov, Matthew D Hoffman, Yarin Gal, Yingzhen Li, Melanie F Pradier, Sharad Vikram, Andrew Foong, Sanae Lotfi, and Sebastian Farquhar. Evaluating approximate inference in bayesian deep learning. In Douwe Kiela, Marco Ciccone, and Barbara Caputo, editors, *Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track*, volume 176 of *Proceedings of Machine Learning Research*, pages 113–124. PMLR, 06–14 Dec 2022. URL https://proceedings.mlr.press/v176/wilson22a.html.

Anqi Wu, Sebastian Nowozin, Edward Meeds, Richard E. Turner, Jose Miguel Hernandez-Lobato, and Alexander L. Gaunt. Deterministic variational inference for robust bayesian neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=B1l08oAct7.