A STATISTICAL THEORY OF COLD POSTERIORS IN DEEP NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

To get Bayesian neural networks to perform comparably to standard neural networks it is usually necessary to artificially reduce uncertainty using a "tempered" or "cold" posterior. This is extremely concerning: if the generative model is accurate, Bayesian inference/decision theory is optimal, and any artificial changes to the posterior should harm performance. While this suggests that the prior may be at fault, here we argue that in fact, BNNs for image classification use the wrong likelihood. In particular, standard image benchmark datasets such as CIFAR-10 are carefully curated. We develop a generative model describing curation which gives a principled Bayesian account of cold posteriors, because the likelihood under this new generative model closely matches the tempered likelihoods used in past work.

1 INTRODUCTION

Recent work has highlighted that Bayesian neural networks (BNNs) typically have better predictive performance when we "sharpen" the posterior (Wenzel et al., 2020). In stochastic gradient Langevin dynamics (SGLD) (Welling & Teh, 2011), this can be achieved by multiplying the log-posterior by $1/T$, where the "temperature", T is smaller than 1 (Wenzel et al., 2020). Broadly the same effect can be achieved in variational inference by "tempering", i.e. downweighting the KL term. As noted in Wenzel et al. (2020), this approach has been used in many recent papers to obtain good performance, albeit without always emphasising the importance of this factor (Zhang et al., 2017; Bae et al., 2018; Osawa et al., 2019; Ashukha et al., 2020).

These results are puzzling if we take the usual Bayesian viewpoint, which says that the Bayesian posterior, used with the right prior, and in combination with Bayes decision theory should give optimal performance (Jaynes, 2003). Thus, these results may suggest we are using the wrong prior. While new priors have been suggested (e.g. Ober & Aitchison, 2020), they give only minor improvements in performance — certainly nothing like enough to close the gap to carefully trained finite networks. In contrast, tempered posteriors directly give performance comparable to a carefully trained finite network.

The failure to develop an effective prior suggests that we should consider alternative explanations for the effectiveness of tempering. Here, we consider the possibility that it is predominantly (but not entirely) the *likelihood*, and not the prior that is at fault. In particular, we note that standard image benchmark datasets such as ImageNet and CIFAR-10 are carefully curated, and that it is important to consider this curation as part of our generative model. We develop a simplified generative model describing dataset curation which assumes that a datapoint is included in the dataset only if there is unanimous agreement on the class amongst multiple labellers. This model naturally multiplies the effect of each datapoint, and hence gives posteriors that closely match tempered or cold posteriors. We show that toy data drawn from our generative model of curation can give rise to optimal temperatures being smaller than 1. Our model predicts that cold posteriors will not be helpful when the original underlying labels from all labellers are available. While these are not available for standard datasets such as CIFAR-10, we found a good proxy: the CIFAR-10H dataset (Peterson et al., 2019), in which ~ 50 humans annotators labelled the CIFAR-10 test-set (we use these as our training set, and use the standard CIFAR-10 training set for test-data). As expected, we find dramatic cold-posteriors effects when using the original single-label, and cold-posterior effects are almost entirely eliminated when using the 50 labels from CIFAR-10H.

$Y_1 = Y_2 = Y_3 = \text{cat}$ $\quad$ $Y_1 = Y_2 = Y_3 = \text{dog}$ $\quad$ $Y_1 = Y_2 = \text{cat}; \quad Y_3 = \text{dog}$

$Y = \text{cat}$ $\quad$ $Y = \text{dog}$ $\quad$ $Y = \texttt{None}$

Figure 1: A simple example of our generative process describing dataset curation, with $S = 3$ labellers with the task of classifying images as cats or dogs. The first image is unambiguously a cat, so the three labellers agree, consensus is reached, and the image is included in the dataset. The second image is unambiguously a dog, the labellers agree, consensus is reached, and the image is included in the dataset. However, the third image is ambiguous: the labellers disagree, consensus is not reached, and the image is excluded from the dataset.

2 Background: cold and tempered posteriors

Tempered and cold posteriors differ slightly in how they apply the temperature parameter. For cold posteriors, we scale the whole posterior, whereas tempering is a method typically applied in variational inference, and corresponds to scaling the likelihood but not the prior,

$$\log \mathrm{P}_{\text{cold}}\left(\theta|X,Y\right) = \tfrac{1}{T}\log \mathrm{P}\left(X,Y|\theta\right) + \tfrac{1}{T}\log \mathrm{P}\left(\theta\right) + \text{const} \tag{1}$$

$$\log \mathrm{P}_{\text{tempered}}\left(\theta|X,Y\right) = \tfrac{1}{\lambda}\log \mathrm{P}\left(X,Y|\theta\right) + \quad \log \mathrm{P}\left(\theta\right) + \text{const}\,. \tag{2}$$

While cold posteriors are typically used in SGLD, tempered posteriors are usually targeted by variational methods. In particular, variational methods apply temperature scaling to the KL-divergence between the approximate posterior, $\mathrm{Q}\left(\theta\right)$ and prior,

$$\mathcal{L} = \mathbb{E}_{\mathrm{Q}(\theta)}\left[\log \mathrm{P}\left(X,Y|\theta\right)\right] - \lambda\, \mathrm{D}_{\mathrm{KL}}\left(\mathrm{Q}\left(\theta\right) || \,\mathrm{P}\left(\theta\right)\right) \tag{3}$$

Our work justifies the use of tempered posteriors, but in practice we believe that the main effect comes from reducing uncertainty, so we expect similar effects in both settings.

3 Methods: a generative model for curated datasets

Standard image datasets such as CIFAR-10 and ImageNet are carefully curated to include only unambiguous examples of each class. For instance, in CIFAR-10, student labellers were paid per hour (rather than per image), were instructed that "It's worse to include one that shouldn't be included than to exclude one", and then Krizhevsky (2009) "personally verified every label submitted by the labellers". For ImageNet, Deng et al. (2009) required the consensus of a number of Amazon Mechanical Turk labellers before including an image in the dataset.

To understand the statistical patterns that might emerge in these curated datasets, we consider a highly simplified generative model of consensus-formation. In particular, we draw a random image Z from some underlying distribution over images, $\mathrm{P}\left(Z\right)$, and ask S humans to assign a label, $\{Y_s\}_{s=1}^S$ (e.g. using Mechanical Turk). We force every labeller to label every image and if the image is ambiguous they are instructed to give a random label. If all the labellers agree, $Y_1 = Y_2 = \cdots = Y_S$, consensus is reached and we include the datapoint in the dataset. If any of the labellers disagree consensus is not reached, and we exclude the datapoint (Fig. 1), Formally, the observed random variable, Y, is taken to be the usual label pair if consensus was reached and `None` if consensus was not reached,

$$Y|\{Y_s\}_{s=1}^S = \begin{cases} Y_1 & \text{if } Y_1 = Y_2 = \cdots = Y_S \\ \texttt{None} & \text{otherwise} \end{cases} \tag{4}$$

Typically, when consensus is not reached, the input datapoint is also excluded from the dataset,

$$X|Z,\{Y_s\}_{s=1}^S = \begin{cases} Z & \text{if } Y_1 = Y_2 = \cdots = Y_S \\ \texttt{None} & \text{otherwise} \end{cases} \tag{5}$$

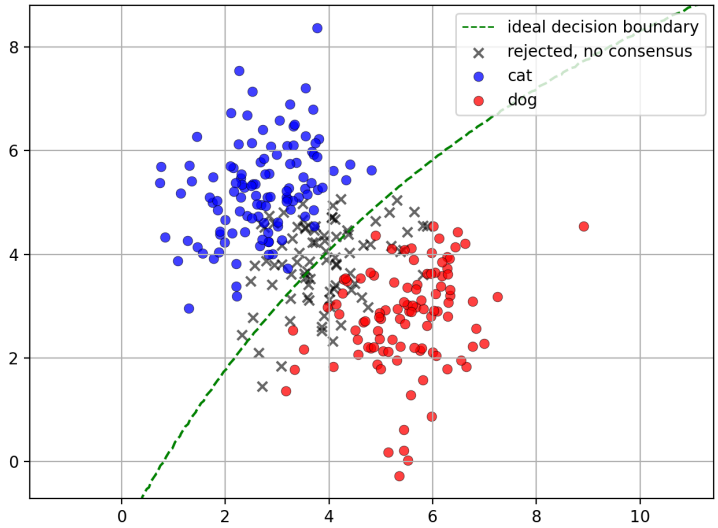

Figure 2: Schematic diagram of artificial clustering induced in the dataset by rejection of ambiguous no-consensus points.

but we can in principle consider the case in which the inputs for no-consensus points are known (e.g. in simulated datasets). Thus, taking the human labels, Y_s, to come from the set $\mathcal{Y}$, so $Y_s \in \mathcal{Y}$, the consensus label, Y, could be any of the underlying labels in $\mathcal{Y}$, or $\texttt{None}$ if no consensus is reached, so $Y \in \mathcal{Y} \cup \{\texttt{None}\}$. Likewise, if Z lives in the space of all images, $Z \in \mathcal{Z}$, then the observation could be any image, or $\texttt{None}$ if no consensus is reached, so $X \in \mathcal{Z} \cup \{\texttt{None}\}$.

In standard supervised learning (Fig. 3A), the likelihood is in truth given by $\mathrm{P}\left(Y, X|\theta\right)$, as the data are both the inputs, X and outputs, Y, rather than the usual conditional distribution, $\mathrm{P}\left(Y|X,\theta\right)$. In the standard supervised setting, this distinction is not relevant because,

$$\mathrm{P}\left(Y, X|\theta\right) = \mathrm{P}\left(Y|X,\theta\right)\mathrm{P}\left(X\right) \propto \mathrm{P}\left(Y|X,\theta\right), \tag{6}$$

where, the proportionality arises because we are only interested in the parameter dependence, and $\mathrm{P}\left(X\right)$ term is independent of the parameters, θ. However, in our generative model of curated data, the observed value of X (in particular whether it is $\texttt{None}$) may depend on $\{Y_s\}_{s=1}^{S}$ and hence on the model parameters, θ (Fig. 3B), so we have to be more careful. In particular, when consensus was reached, i.e. for X, Y not $\texttt{None}$ we have,

$$\mathrm{P}\left(Y{=}y, X{=}z|\theta\right) = \mathrm{P}\left(\{Y_s{=}y\}_{s=1}^{S}|Z{=}z,\theta\right)\mathrm{P}\left(Z{=}z\right) \propto \prod_s \mathrm{P}\left(Y_s{=}y|Z{=}z,\theta\right) \tag{7}$$

where we take the labellers to be IID, and the final proportionality arises because we are only interested in the parameter dependence and $\mathrm{P}\left(Z\right)$ is independent of the parameters. This likelihood is equivalent to labelling each datapoint S times with the same label, and therefore has exactly the effect of setting $\lambda = S$ in a tempered posterior.

However, we also need to consider the likelihood if consensus was not reached. First, we consider the case where all datapoints including those where consensus was not reached are known. Thus, our likelihood for points that did not reach consensus is given by taking $Y = \texttt{None}$ and $X \neq \texttt{None}$,

$$\mathrm{P}\left(Y{=}\texttt{None}, X{=}z|\theta\right) \propto \mathrm{P}\left(Y{=}\texttt{None}|Z{=}z,\theta\right) = 1 - \sum_y \prod_s \mathrm{P}\left(Y_s{=}y|Z{=}z,\theta\right). \tag{8}$$

In the more usual case where the no-consensus inputs are not known, we need to integrate over possible input images,

$$\mathrm{P}\left(Y{=}\texttt{None}, X{=}\texttt{None}|\theta\right) \propto \int dz\, \mathrm{P}\left(Z{=}z\right)\mathrm{P}\left(Y{=}\texttt{None}|Z{=}z,\theta\right) \tag{9}$$

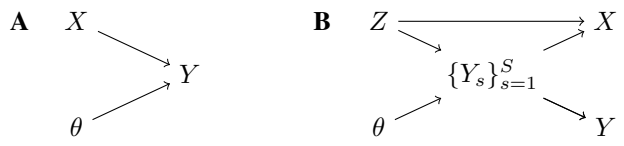

Figure 3: **A** The standard generative model for supervised tasks, assuming no data curation. **B** Our generative model, incorporating the data curation process.

This integral is difficult to compute as in most cases, the distribution over underlying images is not known, and we do not even have samples from the distribution (typically, we only have samples of the consensus images). As such, the typical approach is simply to ignore this term, but we will consider it in our experiments on simulated data, at least for cases where the no-consensus input is known.

4 Results

4.1 Data sampled from a known Gaussian process model

In the introduction, we took it as given that if the prior is correct, the optimal approach is to use the true Bayesian posterior, avoiding either tempering or cold posteriors. To check that this is indeed the case, we tested the performance, measured as test-log-likelihood for tempered posteriors using data generated from a known Gaussian process model. We uniformly generated 50 input points on the 1D interval [-10, 10], and used a Gaussian process with squared exponential kernel with a standard deviation of 4, and kernel bandwidth of 1. For inference, we used reparameterised VI, with an approximate posterior given by multiplying the prior by a single Gaussian factor for each datapoint (Ober & Aitchison, 2020).

In particular, we used a Gaussian process prior for the function values, $\mathbf{u}$, at the training inputs (Williams & Rasmussen, 2006),

$$\mathrm{P}\left(\mathbf{u}|\mathbf{x}\right) = \mathcal{N}\left(\mathbf{u}; \mathbf{0}, \mathbf{K}(\mathbf{x})\right) \tag{10}$$

and we use an approximate posterior (Hensman et al., 2015; Matthews et al., 2016; Ober & Aitchison, 2020) defined by multiplying the prior by a Gaussian with diagonal covariance, where we treat $\mathbf{v}$ and $\mathbf{\Lambda}^{-1}$ as variational parameters.

$$\mathrm{Q}\left(\mathbf{u}\right) \propto \mathrm{P}\left(\mathbf{u}|\mathbf{x}\right)\mathcal{N}\left(\mathbf{v}; \mathbf{u}, \mathbf{\Lambda}^{-1}\right) = \mathcal{N}\left(\mathbf{u}; \mathbf{\Sigma}\mathbf{\Lambda}\mathbf{v}, \mathbf{\Sigma}\right) \quad \text{where} \quad \mathbf{\Sigma} = \left(\mathbf{K}^{-1}(\mathbf{x}) + \mathbf{\Lambda}\right)^{-1} \tag{11}$$

Note that this captures the optimal posterior in the case of regression (where $\mathbf{v}$ is set to $\mathbf{y}$ and $\mathbf{\Lambda}^{-1}$ is the output noise covariance; Ober & Aitchison, 2020). We then optimize the tempered ELBO (e.g. Zhang et al., 2017),

$$\mathcal{L} = \mathbb{E}_{\mathrm{Q}(\mathbf{u})}\left[\log \mathrm{P}\left(\mathbf{y}|\mathbf{u}\right) + \lambda\left(\log \mathrm{P}\left(\mathbf{u}|\mathbf{x}\right) - \log \mathrm{Q}\left(\mathbf{u}\right)\right)\right]. \tag{12}$$

And we used the standard GP approach for prediction at test points (Williams & Rasmussen, 2006).

Initially, we tried using standard regression and classification generative models, without including our model of consensus formation. Unsurprisingly, $\lambda = 1$, corresponding to the Bayesian posterior, is optimal for GP regression (Fig. 4A) and classification (Fig. 4B). For GP regression, we use a Gaussian likelihood with standard deviation 1, and for classification, we use a standard sigmoid probability with a Bernoulli likelihood.

Next, we confirmed that even under our new, more complex generative model of dataset curation $\lambda = 1$ was still optimal (Fig. 4C), if we trained and tested using the correct form for the log-likelihood. In particular, we considered no-consensus inputs as known and used Eq. 8 to incorporate their likelihood. However, in standard benchmarks, only a single consensus label for the consensus datapoints is available, both for training and testing. Next, we generated curated data, but trained and tested on the standard classification log-likelihood. This classification log-likelihood does not match the true generative process which involves curation (Fig. 4D), as it ignores no-consensus points and does not take into account the number of labellers. We used $S = 4$ and found that the optimal setting of λ is no longer 1, but is instead $\lambda \approx 1/4 \approx 1/S$. This is to be expected because the true likelihood for consensus points in Eq. (7) is exactly the classification likelihood, to the power of S.

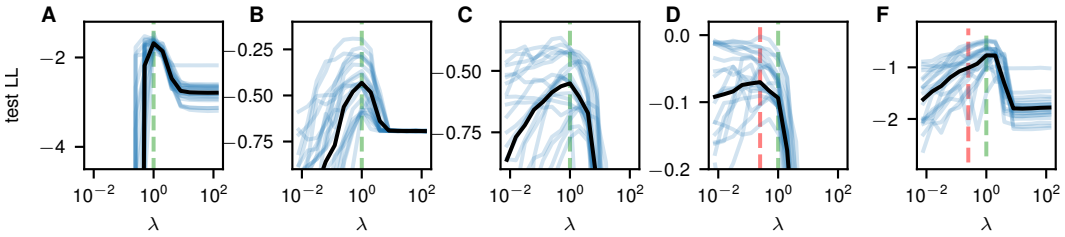

Figure 4: Tempering with data generated from a toy Gaussian process model. On the x-axis, we always display λ, and the green dashed line represents the expected optimal value of $\lambda = 1$. The black line is the mean of 20 runs with different datasets (translucent blue lines). **A** GP regression. **B** GP classification (with no consensus formation). **C** GP classification with our model of consensus formation. We train and test using our exact log-likelihood, and we include knowledge of the no-consensus input datapoints in the datasets. **D** Data is generated as in **C** using our model of curation, but we train and test using the standard cross-entropy (equivalent to the log-likelihood in a classification model). **E** As in D, but we compute the test-log-likelihood using the exact form, including the no-consensus points.

Finally, if we generate curated data, train using classification log-likelihood (which does not match the true generative process) but test on the correct likelihood (Fig. 4E), then the optimal temperature returns to around 1 (though we have no theoretical guarantees that this must always be the case).

4.2 CIFAR-10H

One prediction made by our framework is that if we had access to the original underlying human labels on the full dataset, including images that were rejected because consensus was not reached, then tempering should not be necessary. Obviously, all this additional information is not available for standard datasets such as CIFAR-10. However, we are able to get close by considering the CIFAR-10H dataset (Peterson et al., 2019). The authors of this work asked around 50 human labellers to label the CIFAR-10 test-set. As we might expected given the careful curation that went in to creating the original CIFAR-10 dataset (Krizhevsky, 2009), for almost half of the datapoints, all ~ 50 labellers agreed, and more than three-quarters of images had 2 or fewer disagreements (corresponding to 4%) of labellers (Fig. 5A). While it is not possible to estimate S without having information about the unknown image distribution, the high level of agreement would indicate that the effective value of S is large — potentially even larger than 10.

Next, we trained a neural network on the ~ 50 labels provided by the CIFAR-10H dataset. As these labels are provided for the test-set with 10,000 images, this required us to swap the identities of the test and training sets (so that our training set consisted of 10,000 points, each with around 50 labels, and the test set consists of 50,000 points, with the single label from the original CIFAR-10 dataset). To perform probabilistic inference, we used SGLD with code adapted from (Zhang et al., 2019). In particular, we used their standard settings of a ResNet18, momentum of 0.9, and a cyclic cosine learning rate schedule. Due to the smaller size of our training set, we used longer cycles (150 epochs rather than 50), and more cycles (8 rather than 4). We also decreased the learning rate for CIFAR-10H by a factor of 50 because otherwise the 50 labels would dramatically increase the effective learning rate. Note that under this change in learning rate, the stationary distribution was left unchanged, as changing the learning rate is *not* equivalent to tempering. All other parameters were left at the values specified by Zhang et al. (2019).

As expected, we when training on the single-label from the original CIFAR-10 dataset (blue lines Fig. 5BC), there are very large effects of tempering. In contrast, when training on the ~ 50 labels provided CIFAR-10H (orange), the effects of tempering are far smaller.

Note that while tempering-effects are dramatically reduced, they have not been eliminated entirely. This is expected in our setting, both because of the mismatch between this setup and the exact setup (particularly in the lack of knowledge of the underlying dataset of images, including those that were rejected), but also because of the inference method. SGLD, becomes exact in the limit as the learning rate converges to zero, this limit can never be reached in practical settings (Welling & Teh, 2011).

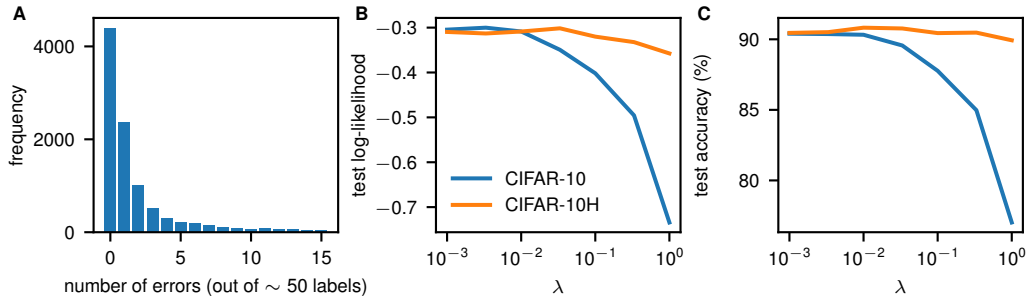

Figure 5: **A** The number of errors (defined as labellers who disagree with the most popular category) out of around 50 labels. **B** The test log-likelihood for different values of λ for training with the standard single-labels provided by CIFAR-10 and the 50 labels provided by CIFAR-10H. Note that here we have swapped the identities of the train and test set. **C** As in **B**, but for accuracy.

In SGLD minibatch gradients are used as a proxy for the gradient for the full dataset, but these minibatch gradients contain additional noise, and there is a potential that reducing the temperature may partially compensate for this noise. This is particularly evident if we consider (Wenzel et al., 2020, Fig. 6) , which showed that the cold-posterior effects can be amplified by using very small minibatch sizes, which makes sense because using smaller minibatch sizes amplifies the noise in the minibatch gradient evaluations. That said, these effects do not appear to be significant for CIFAR-10 (Wenzel et al., 2020, Fig. 5), so we are in agreement with Wenzel et al. (2020) that minibatch noise cannot be the primary source of tempering effects.

5 Related work

Concurrent work (Adlam et al., 2020) has raised the possibility that BNNs overestimate aleatoric uncertainty, in part because of high-quality labels available in benchmark datasets. However, they concluded that while cold posteriors might help us to capture our priors, they do not correspond to an exact inference procedure. In contrast, here we give a generative model of dataset curation in which tempered likelihoods emerge naturally.

6 Conclusions

We showed that modelling the process of data-curation can explain the improved performance of tempering or cold posteriors in Bayesian neural networks. That said, we expect that both the misspecification of priors over neural networks weights and inexact inference makes additional contributions to the cold-posterior effect. We hope that our work will prompt more careful dataset and study of data curation. Finally, the same likelihood has much broader applicability, as it also provides an explanation for the effectiveness of common semi-supervised learning methods (Aitchison, 2020).

References

Ben Adlam, Jasper Snoek, and Samuel L Smith. Cold posteriors and aleatoric uncertainty. *arXiv preprint arXiv:2008.00029*, 2020.

Laurence Aitchison. A statistical theory of semi-supervised learning. *arXiv preprint*, 2020.

Arsenii Ashukha, Alexander Lyzhov, Dmitry Molchanov, and Dmitry Vetrov. Pitfalls of in-domain uncertainty estimation and ensembling in deep learning. *arXiv preprint arXiv:2002.06470*, 2020.

Juhan Bae, Guodong Zhang, and Roger Grosse. Eigenvalue corrected noisy natural gradient. *arXiv preprint arXiv:1811.12565*, 2018.

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

James Hensman, Alexander Matthews, and Zoubin Ghahramani. Scalable variational gaussian process classification. *JMLR*, 2015.

Edwin T Jaynes. *Probability theory: The logic of science*. Cambridge university press, 2003.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Toronto, 2009.

Alexander G de G Matthews, James Hensman, Richard Turner, and Zoubin Ghahramani. On sparse variational methods and the kullback-leibler divergence between stochastic processes. *Journal of Machine Learning Research*, 51:231–239, 2016.

Sebastian W Ober and Laurence Aitchison. Global inducing point variational posteriors for bayesian neural networks and deep gaussian processes. *arXiv preprint arXiv:2005.08140*, 2020.

Kazuki Osawa, Siddharth Swaroop, Mohammad Emtiyaz E Khan, Anirudh Jain, Runa Eschenhagen, Richard E Turner, and Rio Yokota. Practical deep learning with bayesian principles. In *Advances in Neural Information Processing Systems*, pp. 4289–4301, 2019.

Joshua C Peterson, Ruairidh M Battleday, Thomas L Griffiths, and Olga Russakovsky. Human uncertainty makes classification more robust. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9617–9626, 2019.

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)*, pp. 681–688, 2011.

Florian Wenzel, Kevin Roth, Bastiaan S Veeling, Jakub Świątkowski, Linh Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, and Sebastian Nowozin. How good is the bayes posterior in deep neural networks really? *arXiv preprint arXiv:2002.02405*, 2020.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

Guodong Zhang, Shengyang Sun, David Duvenaud, and Roger Grosse. Noisy natural gradient as variational inference. *arXiv preprint arXiv:1712.02390*, 2017.

Ruqi Zhang, Chunyuan Li, Jianyi Zhang, Changyou Chen, and Andrew Gordon Wilson. Cyclical stochastic gradient mcmc for bayesian deep learning. *arXiv preprint arXiv:1902.03932*, 2019.