Ensembling as Approximate Bayesian Inference for Predictive Uncertainty Estimation in Deep Learning

Fredrik K. Gustafsson
Department of Information Technology
Uppsala University, Sweden
fredrik.gustafsson@it.uu.se

Martin Danelljan
Computer Vision Laboratory
ETH Zurich, Switzerland
martin.danelljan@vision.ee.ethz.ch

Thomas B. Schön
Department of Information Technology
Uppsala University, Sweden
thomas.schon@it.uu.se

Abstract

We view ensembling as an approximate Bayesian inference method, justify why it should be a reasonable approximation for Deep Neural Networks and extensively compare it with other approximate methods in terms of predictive uncertainty estimation quality. We provide experimental results on illustrative toy problems and the real-world computer vision tasks of street-scene semantic segmentation and depth completion. *This extended abstract describes preliminary results from ongoing work intended for NeurIPS 2019.*

1 Introduction

Deep Neural Networks (DNNs) have become the standard paradigm within most computer vision problems due to their astonishing predictive power compared to previous alternatives. Current applications include many safety-critical tasks, such as street-scene semantic segmentation [5] and depth completion [18]. Since erroneous predictions can have disastrous consequences, such applications require an accurate measure of the predictive uncertainty.

Within the Bayesian framework, the learned models should ideally be able to capture two different types of uncertainty, as described by Kendall and Gal [15]. *Epistemic (model) uncertainty* accounts for uncertainty in the model parameters, while *aleatoric (data) uncertainty* captures inherent and irreducible data noise. Large *epistemic uncertainty* is present in cases where a large set of model parameters explains the data about equally well. Input-dependent *aleatoric uncertainty* is present whenever the estimated targets y are expected to be inherently more uncertain for some inputs x.

In many computer vision applications, *aleatoric uncertainty* can be effectively estimated by letting a DNN directly output the parameters of some probability distribution, modeling the conditional distribution $p(y|x)$. For classification tasks, this is often realized by a softmax output layer, while Laplace and Gaussian models have been employed for regression [14, 4, 15, 17]. In the conventional approach of learning just a single point estimate of the DNN parameters, these models do however fail to capture any notion of *epistemic uncertainty*. Estimating *epistemic uncertainty* with DNNs is in fact a highly challenging task, since the vast dimensionality of the parameter space renders standard Bayesian inference approaches intractable. To tackle this problem, various approximate inference techniques have been explored [19, 1, 13, 12, 2, 22, 3, 23], with the most commonly used method being MC-dropout [9, 8, 15, 10, 16]. Previous work has also explored the use of ensembling [6] as a non-Bayesian alternative for epistemic uncertainty estimation [17, 4, 14].

Preprint. Work in progress.

In this work, we study how to learn DNN models for computer vision classification and regression tasks which are capable of capturing *both* aleatoric and epistemic uncertainty. Similar to previous work [17, 4, 14], we directly model the conditional distribution in order to estimate input-dependent *aleatoric uncertainty*, and employ ensembling in order to estimate *epistemic uncertainty*. We do however view ensembling as an approximate Bayesian inference method, justify why it should be a reasonable approximation for DNNs and extensively compare it with other approximate methods in terms of predictive uncertainty estimation quality.

Specifically, our main contributions are: **(1)** We quantitatively measure how well the predictive distribution of various approximate Bayesian inference methods, including ensembling, approximates that of Hamiltonian Monte Carlo [19, 20] on an illustrative toy problem for both regression and classification. **(2)** We propose a framework for evaluating the quality of predictive uncertainty estimates that is specifically designed for real-world computer vision applications. **(3)** We demonstrate via a rigorous experimental evaluation that ensembling seems to consistently produce more reliable and useful predictive uncertainty estimates than the commonly used MC-dropout method.

2 Ensembling as approximate Bayesian inference

Ensembling [6] is a general procedure of learning M point estimates $\{\hat{\theta}^{(m)}\}_{m=1}^{M}$ of the parameters θ of some model. In our setting, we let a DNN f_θ output the parameters of a certain probability distribution to create a parametric model $p(y|x,\theta)$ of the conditional distribution. If we learn multiple point estimates $\{\hat{\theta}^{(m)}\}_{m=1}^{M}$ and average over the corresponding parametric models $\{p(y|x,\hat{\theta}^{(m)})\}_{m=1}^{M}$, we obtain the following predictive distribution,

$$\hat{p}(y^\star|x^\star) \triangleq \frac{1}{M}\sum_{m=1}^{M} p(y^\star|x^\star,\hat{\theta}^{(m)}). \tag{1}$$

Noting that $\{\hat{\theta}^{(m)}\}_{m=1}^{M}$ always can be seen as samples from some distribution $q(\theta)$, and comparing (1) to the approximate predictive posterior distribution for Bayesian inference,

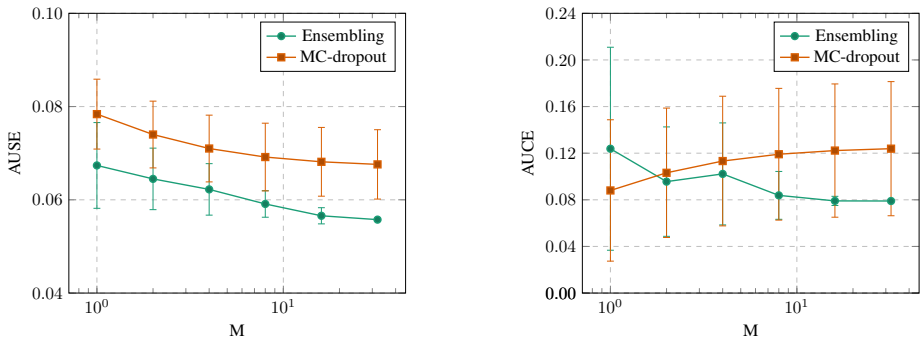
$$p(y^\star|x^\star,\mathcal{D}) = \int p(y^\star|x^\star,\theta)p(\theta|\mathcal{D})d\theta \approx \frac{1}{M}\sum_{i=1}^{M} p(y^\star|x^\star,\theta^{(i)}), \quad \theta^{(i)} \sim p(\theta|\mathcal{D}), \tag{2}$$

we observe that these two expressions are virtually identical. Ensembling can thus be seen as an approximate Bayesian inference method, where the level of approximation is determined by the ensemble size M and how well the implicit sampling distribution $q(\theta)$ approximates the true posterior $p(\theta|\mathcal{D})$. Ideally, we want $\{\hat{\theta}^{(m)}\}_{m=1}^{M}$ to be distributed exactly according to $p(\theta|\mathcal{D}) = p(Y|X,\theta)p(\theta)/p(Y|X)$, where $p(Y|X,\theta)$ is highly *multi-modal* for DNNs. When attempting to represent $p(\theta|\mathcal{D}) \propto p(Y|X,\theta)p(\theta)$ with a relatively small number of samples $\{\hat{\theta}^{(m)}\}_{m=1}^{M}$, the most important aspect to capture in terms of epistemic uncertainty is therefor this multi-modality.

Now consider learning $\{\hat{\theta}^{(m)}\}_{m=1}^{M}$ by repeatedly attempting to find the maximum-a-posteriori (MAP) estimate $\hat{\theta}_{\text{MAP}} = \text{argmax}_\theta\, p(\theta|\mathcal{D})$, by minimizing the corresponding objective $-\log p(Y|X,\theta)p(\theta)$. If there was a unique global optima that we were able to find every time, we would end up with M identical point estimates and $q(\theta)$ would likely be a rather poor approximation of $p(\theta|\mathcal{D})$. However, when we in practice attempt to minimize the objective $-\log p(Y|X,\theta)p(\theta)$ using stochastic gradient algorithms, the best we can hope for is to find a *local* optima. If we attempt to minimize $-\log p(Y|X,\theta)p(\theta)$ multiple times, starting at *randomly chosen* initial points, we are thus likely to end up at different local optima and $\{\hat{\theta}^{(m)}\}_{m=1}^{M}$ will therefor capture some of the multi-modality in $p(\theta|\mathcal{D})$.

3 Experiments

We conduct experiments both on illustrative toy problems and on the real-world computer vision tasks of street-scene semantic segmentation and depth completion. Due to the vastness of the input image space, we argue that it in automotive applications must be expected that models will encounter inputs



(a) (b)

Figure 1: Comparison of ensembling and MC-dropout in terms of AUSE and AUCE on the KITTI depth completion validation dataset.

x which are not well-represented in the training distribution. To better simulate these real-world circumstances in our experiments, we train our semantic segmentation and depth completion models on synthetic data, and evaluate the predictive uncertainty estimates on real-world data.

Depth completion In depth completion, we are given an RGB image $x_{\text{img}} \in \mathbb{R}^{h \times w}$ from a front-facing camera and a corresponding sparse depth map $x_{\text{sparse}} \in \mathbb{R}^{h \times w}$. Non-zero pixels of x_{sparse} are LiDAR depth measurements, projected onto the image plane. The goal is to predict a dense depth map $y \in \mathbb{R}^{h \times w}$, in which each pixel corresponds to a predicted depth measurement. We utilize the Virtual KITTI dataset [7] for training and the KITTI depth completion dataset [11, 21] for evaluation. We use the DNN model presented by Ma *et al.* [18]. The inputs $x_{\text{img}}, x_{\text{sparse}}$ are separately processed by initial convolutional layers, concatenated and fed to an encoder-decoder architecture. We duplicate the final convolutional layer, outputting $\mu \in \mathbb{R}^{h \times w}$ and $\log \sigma^2 \in \mathbb{R}^{h \times w}$ instead of just $\hat{y} \in \mathbb{R}^{h \times w}$. That is, we use a Gaussian model for each pixel.

We evaluate the models in terms of the *Area Under the Sparsification Error curve (AUSE)* metric [14]. AUSE is a *relative* measure of the uncertainty estimation quality, comparing the ordering of predictions induced by the estimated predictive uncertainty with the "oracle" ordering in terms of true prediction error. As an *absolute* measure of uncertainty estimation quality, we also evaluate the models in terms of calibration. Since our models output the mean and variance of a Gaussian distribution for each pixel, we can construct prediction intervals of varying confidence level $p \in]0, 1[$ using the corresponding quantiles. When computing the proportion of pixels for which the prediction interval covers the target, we expect this value to equal $p \in]0, 1[$ for a perfectly calibrated model. We compute the absolute error with respect to perfect calibration and use the area under this curve as our metric, which we call *Area Under the Calibration Error curve (AUCE)*.

A comparison of ensembling and MC-dropout in terms of AUSE and AUCE on the KITTI depth completion validation dataset is found in Figure 1. We observe in Figure 1a that ensembling consistently outperforms MC-dropout in terms of AUSE. The curves do however decrease as a function of M in a similar manner, complicating a definitive ranking in terms of epistemic uncertainty estimation quality. A ranking of the methods can be more readily conducted based on Figure 1b, where we observe a clear improving trend as M increases for ensembling, whereas MC-dropout gets progressively worse.

References

[1] D. Barber and C. M. Bishop. Ensemble learning in Bayesian neural networks. *Nato ASI Series F Computer and Systems Sciences*, 168:215–238, 1998.

[2] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning (ICML)*, pages 1613–1622, 2015.

[3] T. Chen, E. Fox, and C. Guestrin. Stochastic gradient Hamiltonian Monte Carlo. In *International Conference on Machine Learning (ICML)*, pages 1683–1691, 2014.

[4] K. Chua, R. Calandra, R. McAllister, and S. Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4759–4770, 2018.

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016.

[6] T. G. Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[7] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[8] Y. Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, 2016.

[9] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning (ICML)*, pages 1050–1059, 2016.

[10] Y. Gal, J. Hron, and A. Kendall. Concrete dropout. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3581–3590, 2017.

[11] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[12] A. Graves. Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2348–2356, 2011.

[13] G. Hinton and D. Van Camp. Keeping neural networks simple by minimizing the description length of the weights. In *Proceedings of the 6th Annual ACM Conference on Computational Learning Theory (COLT)*, 1993.

[14] E. Ilg, O. Cicek, S. Galesso, A. Klein, O. Makansi, F. Hutter, and T. Bro. Uncertainty estimates and multi-hypotheses networks for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 652–667, 2018.

[15] A. Kendall and Y. Gal. What uncertainties do we need in Bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5574–5584, 2017.

[16] A. Kendall, V. Badrinarayanan, and R. Cipolla. Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2017.

[17] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6402–6413, 2017.

[18] F. Ma, G. V. Cavalheiro, and S. Karaman. Self-supervised sparse-to-dense: Self-supervised depth completion from LiDAR and monocular camera. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[19] R. M. Neal. *Bayesian learning for neural networks*. PhD thesis, University of Toronto, 1995.

[20] R. M. Neal. MCMC using Hamiltonian dynamics. *Handbook of Markov chain Monte Carlo*, 2: 113–162, 2011.

[21] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger. Sparsity invariant CNNs. In *International Conference on 3D Vision (3DV)*, 2017.

[22] M. Welling and Y. W. Teh. Bayesian learning via stochastic gradient Langevin dynamics. In *International Conference on Machine Learning (ICML)*, pages 681–688, 2011.

[23] R. Zhang, C. Li, J. Zhang, C. Chen, and A. G. Wilson. Cyclical stochastic gradient MCMC for Bayesian deep learning. *arXiv preprint arXiv:1902.03932*, 2019.