Laplace Approximation with Diagonalized Hessian for Over-parameterized Neural Networks

Ming Gui *
Technical University of Munich
ming.gui@tum.de

Ziqing Zhao *
Technical University of Munich
ziqing.zhao@tum.de

Tianming Qiu
fortiss
qiu@fortiss.org

Hao Shen
fortiss
shen@fortiss.org

Abstract

Bayesian Neural Networks (BNNs) provide valid uncertainty estimation on their feedforward outputs. However, it can become computationally prohibitive to apply them to modern large-scale neural networks. In this work, we combine the Laplace approximation with linearized inference for a real-time and robust uncertainty evaluation. Specifically, we study the effectiveness and computational necessity of a diagonal Hessian approximation in the Laplace approximation on over-parameterized networks. The proposed approach is investigated on object detection tasks in an autonomous driving scenario and demonstrates faster inference speed and convincing results.

1 Introduction

Two major techniques used for building Bayesian Neural Networks (BNNs) are variational inference [1, 2, 3] and Laplace approximation (LA) [4, 5]. In contrast to the former method, LA can provide a post-hoc uncertainty estimation on a well-trained neural network without retraining. However, the LA method encounters a severe computational burden and memory constraint when dealing with deeper networks due to the increasing size of the Hessian matrix. A most simple solution is to take a diagonal approximation of the Hessian. This work demonstrates that a diagonal Hessian approximation not only brings computational convenience on large-scale problems but also overcomes limitations of the ill-posedness and overestimation of the Laplace approximation.

2 Methodology

2.1 Laplace approximation and linearized inference

Our approach utilizes LA for Bayesian uncertainty estimation. Let us denote the dataset for training by D, the weights of a neural network by w, and the Maximum a Posteriori (MAP) estimation of well-trained weights by w_{MAP}. Then, the second-order Taylor expansion of the log posterior $p(w|D)$ at w_{MAP} can be approximated as

$$\ln(p(w|D)) \approx \ln(p(w_{\text{MAP}}|D)) - \frac{1}{2}(w - w_{\text{MAP}})^\top \widehat{H}(w - w_{\text{MAP}}), \tag{1}$$

*Equal contribution. Work done during internship at fortiss (The Research Institute of the Free State of Bavaria, Germany).

Bayesian Deep Learning workshop, NeurIPS 2021.

where $\widehat{H}$ stands for the Hessian matrix of the negative log posterior $-\ln(p(w|D))$ at w_{MAP}. Exponentiating two sides of Eq. (1), $p(w|D)$ admits the form of $\mathcal{N}(w_{\text{MAP}}, \widehat{H}^{-1})$ [5].

Foong et al. [6] demonstrate that the feedforward function in a BNN at w_{MAP} can be approximated by a linear Gaussian model. Hence, for a certain input-output pair x^*, y^*, the probability of feedforward output $p(y^*|x^*, D)$ after training is described by $\mathcal{N}(f(x^*, w_{\text{MAP}}), g(x^*)^\top \widehat{H}^{-1} g(x^*))$, where $g(x^*)$ is the gradient of the feedforward output w.r.t. the network parameters. The variance of the Gaussian distribution can be interpreted as the uncertainty of y^* at w_{MAP}.

2.2 Posterior Hessian inverse calculation

Accurate modeling of prior According to Bayes' theorem, the posterior probability can be formulated as $p(w|D) = \frac{p(D|w)p(w)}{p(D)}$, where $p(w)$ resembles the weight initialization before a training process. Taking the negative log-likelihood and applying the second derivative w.r.t. weights w of both sides of the aforementioned Bayesian equation yields

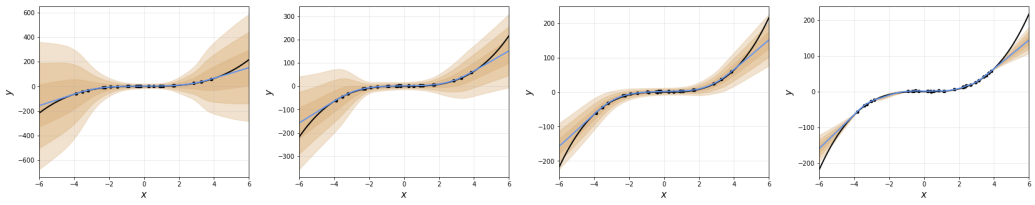
$$\widehat{H} = -\frac{\partial^2 ln(p(D|w))}{\partial w^2} - \frac{\partial^2 ln(p(w))}{\partial w^2} + \frac{\partial^2 ln(p(D))}{\partial w^2} = H + \frac{-\partial^2 ln(p(w))}{\partial w^2}, \tag{2}$$

where H corresponds to the Hessian matrix of the negative log-likelihood $p(D|w)$. H can be approximated as the negative Fisher information matrix under the criterion that the loss function resembles the negative log-likelihood $-\ln(p(D|w))$ [7]. The Fisher matrix is therefore employed as a negative equivalence to the Hessian matrix H due to its lower computational cost [5]. Note that, in the case of uniform weight initialization, $\widehat{H} = H$ remains valid in all circumstances.

Ill-posed problem Calculating H^{-1} is known to be computationally expensive due to the huge number of network parameters. In addition, it can be ill-posed and may require regularization, as is further explained in Appendix A.1. Therefore, a diagonal approximation of the Hessian H can be a potentially simple solution.

2.3 Diagonal approximation with over-parameterization

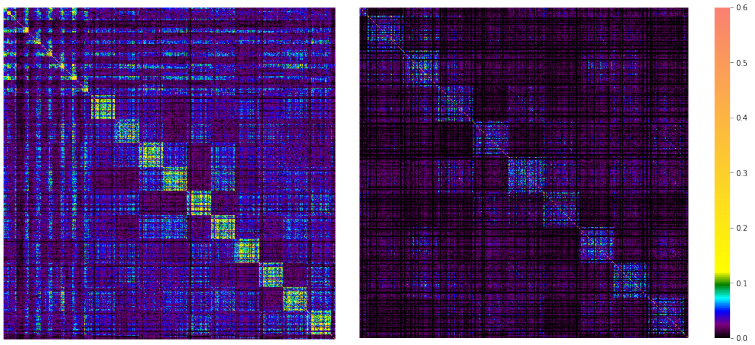
1-D regression problem Our toy dataset is generated with 30 uniformly distributed points $x \sim U(-4, 4)$ and samples $y \sim \mathcal{N}(x^3, 3^2)$. The network contains a single hidden layer with 30 hidden units and uses SiLU activation function [8]. The over-parameterization property of the network is obvious since the number of data points is much smaller than the number of parameters.



(a) FHLA (b) BLA (c) DLA (d) HMC (ground-truth)

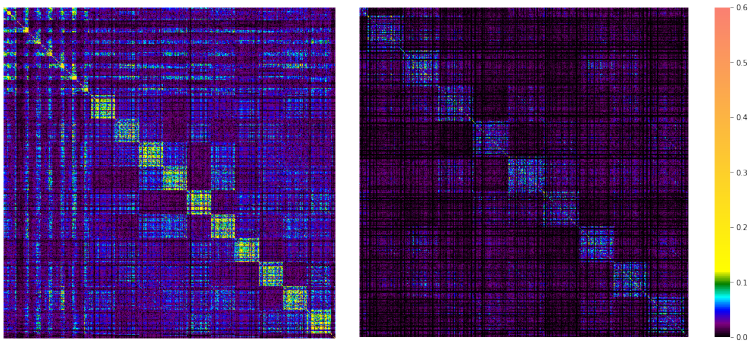
Figure 1: 1-D regression uncertainty results with the different approximated Hessian matrix and HMC ground-truth. The black dots represent the data points, the black line shows the noiseless function x^3, and the blue line shows the mean of our prediction output. Each shade of orange visualizes one additional standard deviation.

Fig. 1 shows the uncertainty of four methods: full Hessian Laplace approximation (FHLA), block-diagonal Hessian Laplace approximation (BLA), pure diagonal Hessian Laplace approximation (DLA), and Hamiltonian Monte Carlo (HMC). The majority of blocks here in BLA are $30 * 30$ in size. Detailed discussion of BLA can be found in the Appendix A.2, where the block size is determined by the network structure. The uncertainty generated by HMC can be considered as the ground truth [9]. The uncertainty changes between the first three methods are consistent: FHLA and BLA tend to overestimate the uncertainty, as the Laplace approximation becomes brittle when the true posterior is multimodal [10]. DLA exhibits smaller variance after discarding part of the Hessian matrix information, resulting in a more similar approximated uncertainty to the true posterior.



(a) $\widehat{H}^{-1}$: Under-param (b) $\widehat{H}^{-1}$: Over-param

Figure 2: The visualized inverse Hessian matrix. The over-parameterization case exhibits significantly fewer inter-dependencies than the under-parameterization case.



(a) Error: Under-param (b) Error: Over-param

Figure 3: The error images of the inverse Hessian matrix.

MNIST classification task Two neural networks are designed and compared on the MNIST dataset [11], one under-parameterized with about 750 parameters and one significantly over-parameterized with about 15000 parameters. Fig. 2 and 3 demonstrates that covariances between different weights in the over-parameterized case diminishes significantly, proving the feasibility of the diagonal approximation. The error images are produced by taking the absolute value of difference of the dense Hessian inverse and the diagonalized Hessian inverse. It can be observed that the diagonal approximation is more precise as the number of network parameters increases. As modern neural networks are typically intensely over-parameterized, we conclude that this method is applicable in common deep learning scenarios. Supplementary discussions regarding Hessian structures, e.g. kernel diagonal can be found in Appendix A.2.

3 Experiment: object detection on Kitti

The proposed approach is applied on Single Shot Multibox Detector (SSD) [12] with Kitti dataset [13]. Both Laplace approximation and Variational Inference, specifically HMC Sampling techniques are employed and compared in our experiment. We use all of the weights for backpropagation as well as uncertainty deduction in Laplace approximation, while 20 full forward propagations are adopted in the MC Sampling case to produce the optimal results [14]. We used a NVIDIA GeForce RTX 2080 Graphics Card for our experiment. It is demonstrated that uncertainty estimation using our proposal can process a Kitti image in 0.42s in average, significantly faster than MC Sampling's 2.62s.

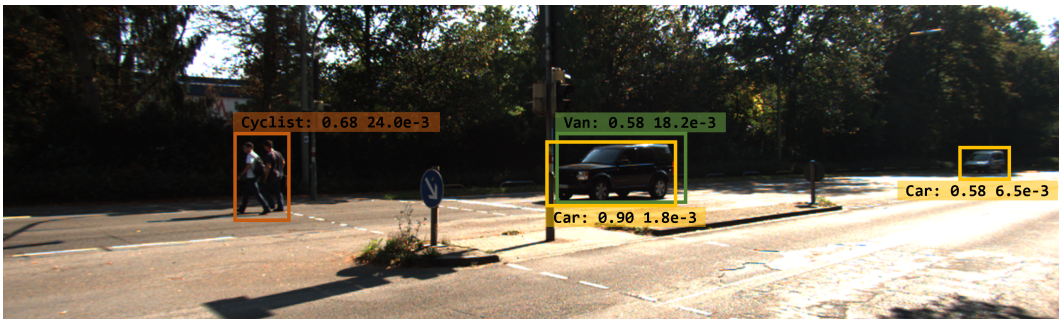


Figure 4: Bounding box classification uncertainty description on Kitti dataset using SSD. The first value in the bounding box is the softmax score, whereas the second value portrays the deducted uncertainty. On the far right of the picture, the correct bounding box of the car has a low score (0.58) and a low classification uncertainty (6.5×10^{-3}), whereas the wrongly classified bounding box of a van in the middle shares the same score, but suffers from a much higher uncertainty (1.82×10^{-2}).

4 Conclusions

We present the superiority of the diagonal Laplace approximation in over-parameterized deep neural networks through experiment results, which requires less computation and produces better real-time performance. We also investigate that this approximation method is more accurate than the full Hessian Laplace approximation. More empirical analysis regarding the Hessian structure of different network structures and the corresponding calculation simplification are expected in the future. Potential engineering acceleration of the diagonal Laplace approximation is also beneficial.

Acknowledgements

The research leading to these results has been carried out within the project "Methoden und Maßnahmen zur Absicherung von KI basierten Wahrnehmungsfunktionen für das automatisierte Fahren (KI-Absicherung)", which is funded by the German Federal Ministry for Economic Affairs and Energy.

References

[1] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, 20–22 Jun 2016.

[2] Yarin Gal. *Uncertainty in deep learning*. PhD thesis, Department of Engineering, University of Cambridge, 2016.

[3] Alex Kendall and Yarin Gal. What uncertainties do we need in Bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[4] John S Denker and Yann LeCun. Transforming neural-net output levels to probability distributions. In *Proceedings of the 3rd International Conference on Neural Information Processing Systems*, pages 853–859, 1990.

[5] Hippolyt Ritter, Aleksandar Botev, and David Barber. A scalable Laplace approximation for neural networks. In *6th International Conference on Learning Representations, ICLR 2018-Conference Track Proceedings*, volume 6. International Conference on Representation Learning, 2018.

[6] Andrew YK Foong, Yingzhen Li, José Miguel Hernández-Lobato, and Richard E Turner. 'In-between' uncertainty in Bayesian neural networks. *ICML 2019 Workshop on Uncertainty and Robustness in Deep Learning*, 2019.

[7] James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *International Conference on Machine Learning*, pages 2408–2417, 2015.

[8] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018.

[9] Radford M Neal. Bayesian learning via stochastic dynamics. In *Advances in Neural Information Processing Systems*, pages 475–482, 1993.

[10] Adriano Azevedo-Filho and Ross D Shachter. Laplace's method approximations for probabilistic inference in belief networks with continuous variables. In *Uncertainty Proceedings 1994*, pages 28–36. Elsevier, 1994.

[11] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[12] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. *Lecture Notes in Computer Science*, page 21–37, 2016.

[13] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[14] Florian Kraus and Klaus Dietmayer. Uncertainty estimation in one-stage object detection. *2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019*, pages 53–60, 2019.

[15] Per Christian Hansen. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*. Society for Industrial and Applied Mathematics, 1999.

[16] Jongseok Lee, Matthias Humt, Jianxiang Feng, and Rudolph Triebel. Estimating model uncertainty of neural networks in sparse information form. In *37th International Conference on Machine Learning (ICML)*. Proceedings of Machine Learning Research, July 2020.

A Appendix

A.1 Ill-posed matrix inverses and Tikhonov regularization

For an ill-posed question, Tikhonov regularization can be particularly useful. In principle, applying a ridge regression on the matrix could effectively mitigate the inverse problem.

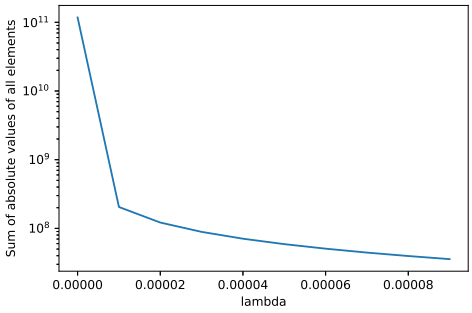
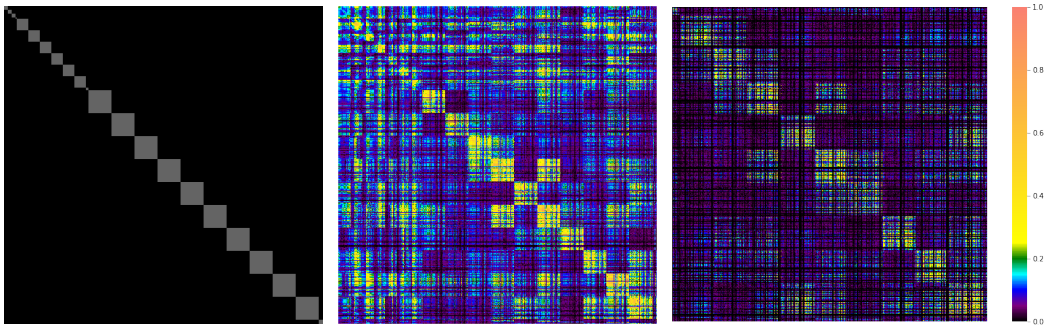


Figure 5: The sum of absolute values of all elements inside the inverse of the regularized Hessian matrix. The result forms a L-curve, which is typical for an ill-posed matrix inverse problem.

The sum of absolute values of all elements inside $(H + \lambda \mathbb{1})^{-1}$ is plotted in Fig. 5. It is pointed out in [15] that the 'vertex' of the L-curve should be the optimal λ to solve the ill-posed question. While this lambda is highly dependent on the network structure, the dataset, and the loss function, introducing such regularization to the matrix can also severely impact the structure and the diagonal property of the Hessian as well as its inverse. Hence, it is essential to find a universal approximation of the Hessian matrix to solve this inverse issue.

A.2 The structures of the Hessian matrix



(a) Under-param kernel structure (b) Under-param (c) Over-param

Figure 6: The visualized Hessian matrix of an under- and an over-parameterized network. While some dependencies still exist between blocks in the under-parameterized network (b), less covariance is found in the over-parameterized scenario (c). Fig. (a) shows the blocks whose information can be extracted from the Hessian matrix in Fig. (b), which saves substantially more memory and calculation than the layer-block approximation proposed by [16].

The Hessian matrix has a specific structure that corresponds to the structure of the network especially when dealing with under-parameterized networks. It can be observed in Fig. 6 that the information of both Hessian likelihood matrices are concentrated in certain blocks, giving rise to the block diagonal approximation. Inferring from this figure, we may assume that for convolutional layers (CNN), the elements within a single kernel are interrelated and kernels are independent of each other. The fully connected layers, or multilayer perceptrons (MLP), exhibit independence between each output layer.