Bootstrap Your Own Variance

Polina Turishcheva[*] [†] [1] **Jason Ramapuram**[*] [2] **Sinead Williamson**[*] [2]

Dan Busbridge[2] **Eeshan Dhekane**[2] **Russ Webb**[2]

[1] University of Göttingen
[2] Apple

turishcheva@cs.uni-goettingen.de
{jramapuram, sa_williamson, dbusbridge, eeshan, rwebb}@apple.com

Abstract

Understanding model uncertainty is important for many applications. We propose Bootstrap Your Own Variance (BYOV), combining Bootstrap Your Own Latent (BYOL), a negative-free Self-Supervised Learning (SSL) algorithm, with Bayes by Backprop (BBB), a Bayesian method for estimating model posteriors. We find that the learned predictive std of BYOV vs. a supervised BBB model is well captured by a Gaussian distribution, providing preliminary evidence that the learned parameter posterior is useful for label free uncertainty estimation. BYOV improves upon the deterministic BYOL baseline (**+2.83%** test ECE, **+1.03%** test Brier) and presents better calibration and reliability when tested with various augmentations (eg: **+2.4%** test ECE, **+1.2%** test Brier for Salt & Pepper noise).

1 Introduction

Quantifying epistemic uncertainty (Hora, 1996) is of crucial importance as we increase the use of machine learning models in daily applications (OpenAI, 2023; Chowdhery et al., 2023; Rombach et al., 2022). This task is well suited for Bayesian machine learning, which replaces point estimates of parameters with a posterior distribution that captures epistemic uncertainty about each parameter's value. While this posterior is typically intractable, we can approximate it using sampling-based methods (Metropolis & Ulam, 1949; Neal et al., 2011; Izmailov et al., 2021) or Stochastic Variational Inference (SVI) (Hoffman et al., 2013) A Bayesian approach facilitates principled model selection (MacKay, 1992; Lotfi et al., 2022) and provides informed decisions that minimize the need for exhaustive hyperparameter searches (Snoek et al., 2012; Lotfi et al., 2022).

Despite its importance, the landscape of uncertainty estimation and calibration within SSL remains relatively unexplored, with limited works addressing this critical aspect (Hendrycks et al., 2019; Bui & Maifeld-Carucci, 2022; Gowal et al., 2021), with none taking a Bayesian approach. We show that SVI approaches—specifically, Bayes by Backprop (BBB)—can be used to learn parameter posteriors in SSL, despite the large scale of models and the absence of a likelihood.

The resulting parameter distributions can be used to provide uncertainty quantification in downstream tasks. It can also give insights about the structure of our model. Modern neural networks are overparameterized (Hu et al., 2021) and most of the common regularisation or pruning methods today are based only on weight magnitudes (Frankle & Carbin, 2018). We show that pruning based on the signal to noise ratio (SNR) ratio of the parameter posterior preserves better performance than magnitude-based pruning in SSL models, extending related findings from the supervised setting (Graves, 2011; Blundell et al., 2015)

[*] Primary contributor. For a detailed breakdown of author contributions see Appendix H.

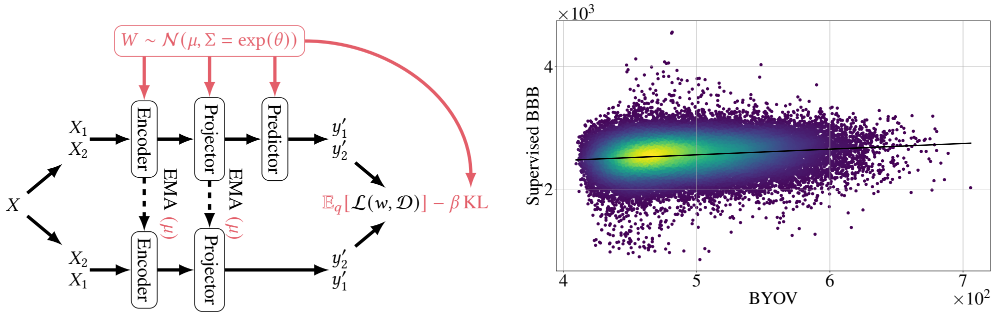
[†] Work done during Apple internship.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

1.1 Contributions

- We propose an algorithm that extends BBB to the SSL setting. Unlike most Bayesian neural networks that work with small models and datasets (e.g., Blundell et al., 2015; Wang et al., 2016; Wen et al., 2018), we scale BBB to Vision Transformers (Dosovitskiy et al., 2020), and train our models on ImageNet-1k (Deng et al., 2009).
- We explore the impact of prior choices on the Bootstrap Your Own Variance (BYOV) posterior, and demonstrate that the resulting uncertainty estimates are distributionally aligned with outputs from a Bayesian supervised model.
- We compare SNR pruning (Graves, 2011) with magnitude based pruning (without retraining) in the SSL setting – SNR pruning is up to 12% better accuracy with a 25% sparser model.

2 Background



(a) BYOL and BYOV Architecture.

(b) Predictive std of supervised BBB vs BYOV.

Figure 1: **(a)** The standard BYOL architecture is shown in black and modifications required for BYOV are highlighted in red. The BYOV student is parameterized with an Isotropic Gaussian approximate parameter posterior. The teacher is the Exponential Moving Average (EMA) of the Maximum a Posteriori (MAP) student parameters. **(b)** Predictive test set standard deviation for supervised BBB versus BYOV overlaid with a Gaussian KDE fit. The predictive std relationship is well captured by a Gaussian distribution, highlighting distributional alignment. Models evaluated over the ImageNet1k test set using 1000 MC draws per sample from the approximate parameter posterior, $q(\mathbf{w}|\boldsymbol{\theta})$. Both models are trained with the same $\beta = 0.0 \mapsto 1.0$ schedule.

2.1 Bayes-by-Backprop

Estimation of the parameter posterior, $p(\mathbf{w}|\mathcal{D})$, is central to Bayesian learning. BBB (Blundell et al., 2015) learns the parameters $\boldsymbol{\theta}$ of an approximate posterior, $q(\mathbf{w}|\boldsymbol{\theta})$, by minimizing the KL-divergence against the true posterior. Since the KL-divergence cannot be evaluated directly, we maximize an alternative objective called the Evidence Lower Bound (ELBO) (Dayan et al., 1995)

$$\mathrm{ELBO}(\theta; \mathcal{D}, \mathbf{w}) = \mathbb{E}_{q(\mathbf{w}|\theta)}\left[\log p(\mathcal{D}|\mathbf{w})\right] - \beta\,\mathrm{KL}\left(q(\mathbf{w}|\theta)||p(\mathbf{w})\right), \tag{1}$$

where $\beta > 0$ is a Lagrange multiplier (Higgins et al., 2016). When $\beta = 1$, maximizing the ELBO is equivalent to minimizing KL $(q(\mathbf{w}|\theta)||p(\mathbf{w}|\mathcal{D}))$. In practice, setting $\beta < 1$, approximating a *cold posterior*, improves predictive performance (Osawa et al., 2019; Wenzel et al., 2020).

2.2 Bootstrap Your Own Latent (BYOL)

BYOL is a negative-free student-teacher distillation framework that minimizes the cosine similarity between a teacher model and an online student model (Figure 1a). The student comprises three networks: an encoder, a Multi-Layer Perceptron (MLP) projector, and a MLP predictor. The teacher model is the exponential moving average of the student encoder and projector. The predictor introduces an asymmetry between the branches and is a necessary component to prevent collapse (Grill et al., 2020). BYOL is trained by inferring *two different augmentations of the same image* through both the student and teacher models and minimizing the cosine similarity between the induced representations. After training we can drop the predictor and projector networks and use the student or teacher encoder representations for downstream tasks, such as image classification.

3 BYOV: A Bayesian SSL method

Here we describe BYOV (Figure 1a), which couples BBB with BYOL.[3] BYOV learns a distribution over the parameters of the student model, and uses the student MAP to update the teacher parameters.

[3] We discuss our choice of BBB in Appendix B and include more implementation details in Appendix C.

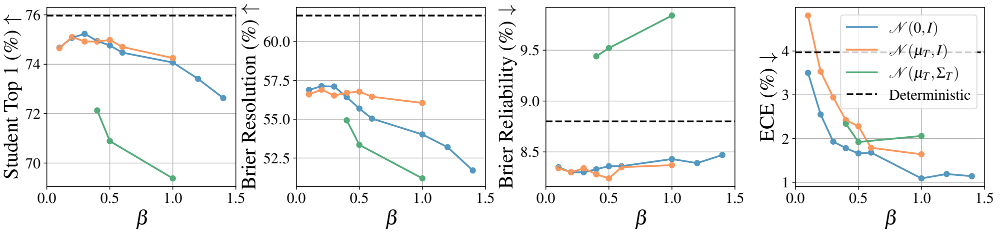


Figure 2: Prior ablation. All metrics here are for in-domain test set using the mean of the student parameter posterior, $\boldsymbol{\mu}$, for inference. The best BYOV models outperform the deterministic BYOL model for ECE (**+2.83%**) and reliability (**+1.03%**), but underperform in top-1 (-0.4%), top-5 (-0.22%) and resolution (-0.57%).

BBB typically approximates the posterior distribution over weights, specified in terms of a prior $p(\mathbf{w})$ and a likelihood $p(\mathcal{D}|\mathbf{w})$. However, BYOL does not use a likelihood—our loss is based on the cosine similarity between two representations. Instead, we estimate a *generalized posterior* (Bissiri et al., 2016), $\tilde{p}(\mathbf{w}|\mathcal{D}) \propto p(\mathbf{w}) \exp\{-\mathcal{L}(\mathbf{w},\mathcal{D})\}$, where $\mathcal{L}(\mathbf{w},\mathcal{D})$ is an arbitrary loss term – in our case, cosine similarity. We therefore minimize the generalized ELBO (Knoblauch et al., 2019),[4]

$$\text{Generalized ELBO} = \mathbb{E}_{q(\mathbf{w}|\boldsymbol{\theta})}\left[\mathcal{L}(\mathbf{w},\mathcal{D})\right] - \beta \, \mathrm{KL}[q(\mathbf{w}|\boldsymbol{\theta})||p(\mathbf{w})]. \tag{2}$$

In theory, the prior $p(\mathbf{w})$ captures our beliefs about parameter values. In the context of neural networks, these priors can be hard to define (Vladimirova et al., 2019; Fortuin et al., 2022). This has led to many Empirical Bayes methods that learn a data dependent prior (Tomczak & Welling, 2018; Bornschein et al., 2017; Wu et al., 2018; Ramapuram et al., 2021). In this work, we consider three priors: (i) a $\mathcal{N}(0,I)$ used by Blundell et al. (2015), (ii) $\mathcal{N}(\mu_T,I)$, using teacher weights as a data informed estimates for the prior means, and (iii) $\mathcal{N}(\mu_T,\Sigma_T)$, where $\Sigma_T = \mathrm{diag}(\boldsymbol{\theta}_T^2 - \bar{\boldsymbol{\theta}}_T^2)$, using the variance of the means (for this we keep an EMA of the second order term $\bar{\boldsymbol{\theta}}_T^2 = \gamma\bar{\boldsymbol{\theta}}_T^2 + (1-\gamma)\boldsymbol{\mu}_S^2$.).

4 Results

4.1 Ablations

We explore the impact of different prior choices in Figure 2, which shows accuracy, Expected Calibration Error (ECE) (Guo et al., 2017) and Brier metrics (Gneiting & Raftery, 2007) based on the MAP parameters, across a range of values of β. We find using the posterior mean (obtained by MC sampling) leads to improved results (e.g., 0.34% higher accuracy with $\mathcal{N}(0,I)$ prior and $\beta = 0.3$), but is more expensive to compute.

The $\mathcal{N}(0,I)$ prior achieves comparable accuracy to a deterministic BYOL model, and improved ECE and reliability. We find little difference between a $\mathcal{N}(0,I)$ prior and a $\mathcal{N}(\boldsymbol{\mu}_T,I)$ prior, with the former performing slightly better. We hypothesize that this is because the overall performance of the neural network is relatively invariant to constant shifts in the weights. However, we see notably worse performance using $\mathcal{N}(\boldsymbol{\mu}_T,\Sigma_T)$. This prior actively pulls the student towards the teacher (since Σ_T is typically fairly small), so we hypothesize that this prior does not encourage sufficient difference between teacher and student. In addition, analysis of parameter logs suggest this prior leads to training instabilities, likely because the prior is dynamically varying over training.

4.2 Exploring the posterior distribution

Since BBB explicitly evaluates the posterior over weights, we are able to explore the distribution of posterior variance over network layers. In addition, we are able to analyse the evolution of this uncertainty over training. In Figure 3, we plot the the mean and maximum value of the learned standard deviation σ and SNR $|\mu|/\sigma$, over training for each layer [5]. We observe that the choice of prior makes a large difference on the learned layer-wise standard deviations. However, if we look at the posterior SNR (Figure 3), we see more similarity across priors, particularly between $\mathcal{N}(0,I)$ and $\mathcal{N}(\mu_T,I)$, supporting the idea that performance is relatively invariant to rescaling. In Appendix E, we show that SNR trajectories remain similar under different choices of β (Figure 6).

[4] Note, the term generalized ELBO has been used to describe multiple modifications to the ELBO (e.g., Chen et al., 2018; Domke & Sheldon, 2018). We specifically refer to the form arising from the "Rule of Three" proposed by Knoblauch et al. (2019).

[5] Since BBB updates the natural parameters of the parameter posterior at each step of the optimization process, each minibatch will induce a separate weight, which we then aggregate over each dataset epoch.

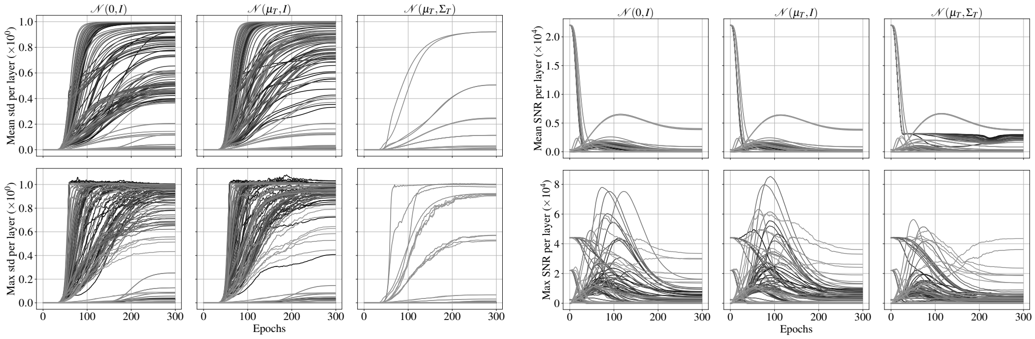


Figure 3: Prior layerwise σ (left) and SNR (right) evolution. All models are trained with $\beta : 0.0 \mapsto 1.0$. Top: layerwise std mean. Bottom: layerwise std max. The layerwise σ trajectories appear different for three different priors. By contrast, the SNR evolution dynamics (including their maximum values) follow a similar trend.

4.3 Does BYOV capture meaningful model uncertainty?

A natural question that arises is whether the learned posterior distribution captures uncertainties that are relevant to downstream tasks. To assess this, in Figure 1b we look at the relationship between uncertainty of the BYOV predictive distribution, and uncertainty under a supervised BBB model. We observe that the relationship between the predictive standard deviation of both models can be suitably captured using a Gaussian, which gives credence to using BYOV as a proxy for the uncertainty of a supervised BBB model. We also look at how incorporating BBB impacts prediction quality, by looking at ECE and Brier reliability score. Previous work on supervised models suggests that BBB improves calibration and reliability (Ovadia et al., 2019). In Figure 4 (Center/Right) we also observe improved calibration on the in-distribution data (ImageNet Test). In an out-of-distribution task, we see improved calibration and reliability on many types of augmentation, but notably worse reliability and calibration under shearing and Gaussian augmentations.

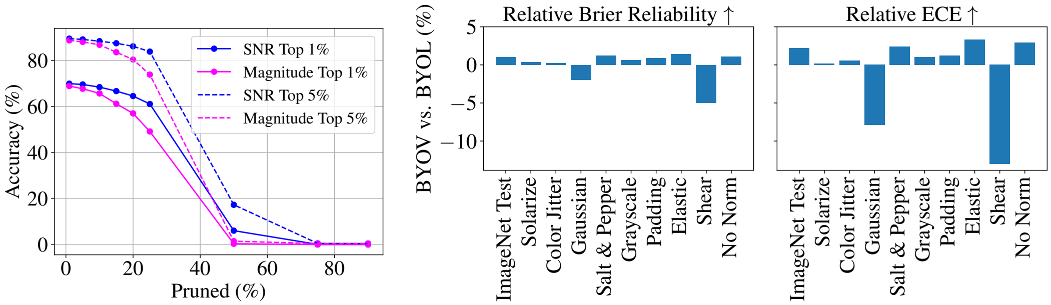


Figure 4: **Left**: SNR vs. magnitude based pruning. **Center** : Relative Brier reliability between BYOV and deterministic BYOL. **Right**: Relative ECE between BYOV and deterministic BYOL.

4.4 Pruning

A high posterior variance indicates that the model lacks confidence in a weight's value. We can use this to prune the network, removing weights where the network lacks confidence. Since network performance can be invariant to weight scale, we follow Blundell et al. (2015) and use SNR for pruning, keeping the xth percentile per layer. In Figure 4 (Left), we show that this achieves better performance than magnitude-based pruning (Frankle & Carbin, 2018), with a sparser model. To simplify our analysis, we do not retrain either model as in Frankle & Carbin (2018).

5 Conclusion

In this work, we introduce BYOV, a method to learn model uncertainty in a label free manner. We explore posterior performance and show that the resulting layerwise SNR is a good metric for model pruning. We show that posterior variance is correlated with that of supervised models, suggesting the distributions can be used for approximate inference in downstream tasks.

References

Pier Giovanni Bissiri, Chris C Holmes, and Stephen G Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78 (5):1103–1130, 2016.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. In *International Conference on Machine Learning*, pp. 1613–1622, 2015.

Jörg Bornschein, Andriy Mnih, Daniel Zoran, and Danilo Jimenez Rezende. Variational memory addressing in generative models. In *Advances in Neural Information Processing Systems*, pp. 3920–3929, 2017.

Jochen Brocker. Reliability, sufficiency, and the decomposition of proper scores. *Quarterly Journal of the Royal Meteorological Society*, 135(643):1512–1519, 2009.

Ha Manh Bui and Iliana Maifeld-Carucci. Benchmark for uncertainty & robustness in self-supervised learning. *arXiv preprint arXiv:2212.12411*, 2022.

Dan Busbridge, Jason Ramapuram, Pierre Ablin, Tatiana Likhomanenko, Eeshan Gunesh Dhekane, Xavier Suau, and Russ Webb. How to scale your ema. *arXiv preprint arXiv:2307.13813*, 2023.

Liqun Chen, Chenyang Tao, Ruiyi Zhang, Ricardo Henao, and Lawrence Carin Duke. Variational inference and model selection with generalized evidence bounds. In *International Conference on Machine Learning*, pp. 893–902, 2018.

Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. Training deep nets with sublinear memory cost. *CoRR*, abs/1604.06174, 2016. URL `http://arxiv.org/abs/1604.06174`.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux-effortless Bayesian deep learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 20089–20103, 2021.

Peter Dayan, Geoffrey E. Hinton, Radford M. Neal, and Richard S. Zemel. The Helmholtz machine. *Neural Comput.*, 7(5):889–904, 1995.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Justin Domke and Daniel R Sheldon. Importance weighting and variational inference. volume 31, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Vincent Fortuin, Adrià Garriga-Alonso, Sebastian W Ober, Florian Wenzel, Gunnar Rätsch, Richard E Turner, Mark van der Wilk, and Laurence Aitchison. Bayesian neural network priors revisited. In *International Conference on Learning Representations*, 2022.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, volume 48, pp. 1050–1059, 2016.

Jochen Gast and Stefan Roth. Lightweight probabilistic deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3369–3378, 2018.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

Sven Gowal, Po-Sen Huang, Aäron van den Oord, Timothy A. Mann, and Pushmeet Kohli. Self-supervised adversarial robustness for the low-label, high-data regime. In *International Conference on Learning Representations*, 2021.

Alex Graves. Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33: 21271–21284, 2020.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330, 2017.

Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2016.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347, 2013.

Stephen C Hora. Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering & System Safety*, 54(2-3):217–223, 1996.

Tianyang Hu, Wenjia Wang, Cong Lin, and Guang Cheng. Regularization matters: A nonparametric perspective on overparametrized neural network. In *International Conference on Artificial Intelligence and Statistics*, pp. 829–837, 2021.

Pavel Izmailov, Sharad Vikram, Matthew D Hoffman, and Andrew Gordon Gordon Wilson. What are Bayesian neural network posteriors really like? In *International Conference on Machine Learning*, pp. 4629–4640, 2021.

Jeremias Knoblauch, Jack Jewson, and Theodoros Damoulas. Generalized variational inference: Three arguments for deriving new posteriors. *arXiv preprint arXiv:1904.02063*, 2019.

Sanae Lotfi, Pavel Izmailov, Gregory Benton, Micah Goldblum, and Andrew Gordon Wilson. Bayesian model selection, the marginal likelihood, and generalization. In *International Conference on Machine Learning*, pp. 14223–14247, 2022.

David J. C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, 1992.

Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. A simple baseline for Bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Nicholas Metropolis and Stanislaw Ulam. The Monte Carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.

Thomas P. Minka. Expectation propagation for approximate Bayesian inference. In *Uncertainty in Artificial Intelligence*, pp. 362–369, 2001.

Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte Carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21:132:1–132:62, 2020.

Radford M Neal et al. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11):2, 2011.

OpenAI. GPT-4 technical report. *CoRR*, abs/2303.08774, 2023. doi: 10.48550/arXiv.2303.08774. URL `https://doi.org/10.48550/arXiv.2303.08774`.

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*, volume 32, 2019.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*, 32, 2019.

Jason Ramapuram, Yan Wu, and Alexandros Kalousis. Kanerva++: Extending the Kanerva machine with differentiable, locally block allocated latent memory. In *International Conference on Learning Representations*, 2021.

Pierre H Richemond, Jean-Bastien Grill, Florent Altché, Corentin Tallec, Florian Strub, Andrew Brock, Samuel Smith, Soham De, Razvan Pascanu, Bilal Piot, et al. Byol works even without batch statistics. *arXiv preprint arXiv:2010.10241*, 2020.

Carlos Riquelme, George Tucker, and Jasper Snoek. Deep Bayesian bandits showdown: An empirical comparison of bayesian deep networks for thompson sampling. In *International Conference on Learning Representations*, 2018.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10674–10685, 2022.

Mrinank Sharma, Sebastian Farquhar, Eric Nalisnick, and Tom Rainforth. Do Bayesian neural networks need to be fully stochastic? In *International Conference on Artificial Intelligence and Statistics*, pp. 7694–7722, 2023.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, 2012.

Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems*, pp. 3738–3746, 2016.

Jakub M. Tomczak and Max Welling. VAE with a vampprior. In *International Conference on Artificial Intelligence and Statistics*, volume 84, pp. 1214–1223, 2018.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pp. 10347–10357, 2021.

Mariia Vladimirova, Jakob Verbeek, Pablo Mesejo, and Julyan Arbel. Understanding priors in Bayesian neural networks at the unit level. In *International Conference on Machine Learning*, pp. 6458–6467, 2019.

Hao Wang, Xingjian Shi, and Dit-Yan Yeung. Natural-parameter networks: A class of probabilistic neural networks. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Yeming Wen, Paul Vicol, Jimmy Ba, Dustin Tran, and Roger Grosse. Flipout: Efficient pseudo-independent weight perturbations on mini-batches. In *International Conference on Learning Representations*, 2018.

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? In *International Conference on Machine Learning*, 2020.

Yan Wu, Greg Wayne, Alex Graves, and Timothy P. Lillicrap. The kanerva machine: A generative distributed memory. In *International Conference on Learning Representations*, 2018.

Appendices

A Acknowledgements

We thank Arno Blaas, Jonathan Crabbé, Megan Maher, and Amitis Shidani for their helpful feedback and critical discussions throughout the process of writing this paper; Okan Akalin, Hassan Babaie, Denise Hui, Mubarak Seyed Ibrahim, Li Li, Cindy Liu, Rajat Phull, Evan Samanas, Guillaume Seguin, and the wider Apple infrastructure team for assistance with developing and running scalable, fault tolerant code. Names are in alphabetical order by last name within group.

B Discussion – Why BBB?

We choose Bayes by Backprop (BBB) as our Bayesian estimator because of its theoretical scalability and the findings from Ovadia et al. (2019), which highlight the competitive performance of Stochastic Variational Inference (SVI) methods for uncertainty estimation. The authors demonstrate that BBB outperforms Expectation Propagation (Minka, 2001), Monte Carlo Dropout (Gal & Ghahramani, 2016), and last layer (LL) Bayesian variants (Riquelme et al., 2018) in terms of Expected Calibration Error (ECE) (Guo et al., 2017) and Brier score (Gneiting & Raftery, 2007; Brocker, 2009).

Although BBB presents a concrete algorithm to learn posterior variances, it draws only one posterior weight variate per minibatch. While being an unbiased estimate, this can potentially be high variance. Towards that end, Flipout (Wen et al., 2018) aims to mitigate the high variance estimate through weight perturbation. However, since Self-Supervised Learning (SSL) methods typically rely on large batch training, we find that this negates the need for such strategies. The authors also confirm this in Appendix E2 (Wen et al., 2018) where they train with a batch size of 8192 which is equivalent in our setting.

C Training details

Modifications to Bootstrap Your Own Latent (BYOL) To keep consistency with the baseline SSL method we closely follow the model architecture and hyper-parameters defined for BYOL (Richemond et al., 2020) with alterations made to support Vision Transformers (Dosovitskiy et al., 2020; Busbridge et al., 2023). However, naively applying BBB to BYOL does not work out-of-the-box and required following changes:

- **Removal of weight decay**: BYOL default recipe includes weight decay. We remove weight decay as it interferes with the learning dynamics when coupling BBB with BYOL. The KL loss term already introduces a regularisation effect and explicitly pulls towards a prior.
- **KL annealing**: a commonly applied practice in latent variable stochastic inference is to use a schedule for the KL divergence (Sønderby et al., 2016). We find that this also helps to improve the downstream tasks performance in Bootstrap Your Own Variance (BYOV) paradigm.

In addition, we made a number of changes to the BBB algorithm, in order to encourage stability:

- **Initialization of $\boldsymbol{\sigma}^2$**: When learning $\boldsymbol{\sigma}^2$ as a free parameter a non-negativity constraint needs to be enforced, we use exponential function for this whereas Blundell et al. (2015) uses Softplus. Log-variances are initialized with -10. Means are initialised using trunc_normal(std=0.02) (Touvron et al., 2021). **Other details**: To keep consistency with the baseline SSL method we closely follow the model architecture and hyper-parameters defined for BYOL (Richemond et al., 2020) with alterations made to support Vision Transformers (Dosovitskiy et al., 2020; Busbridge et al., 2023).
- **Scheduling** β. Previous work has indicated that annealing the β weight applied to the KL term in the ELBO from zero to the desired value yields improved performance over using a fixed value of β. We found this to be the case in practice.

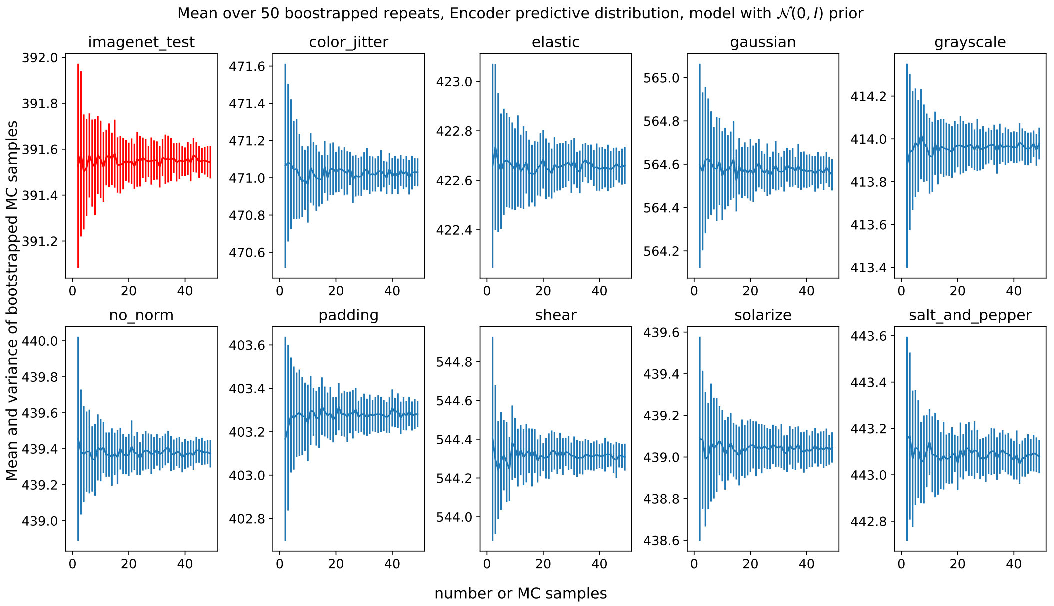


Figure 5: Mean and standard deviation of predictive distribution under different dataset augmentations. Model was trained with a scheduled $\beta : 0.0 \mapsto 1.0$. The standard deviation of the expectation converges to 0 and the expectation appears stable. This is consistent across models.

D MC number of samples estimate

To evaluate the posterior predictive distribution we infer each input sample, $x \sim p(\mathbf{x})$, using K draws from the parameter posterior, $\{\mathbf{w}_i\}_{i=1}^{K} \sim q(\mathbf{w}|\boldsymbol{\theta})$. Previous work uses thirty MC draws (Maddox et al., 2019; Lotfi et al., 2022; Daxberger et al., 2021), but does not justify the validity of this decision. We ablate this in Figure 5 using the entire test set of ImageNet1k (50,000 samples). We sample the posterior from one to fifty times per sample (×50 bootstrap) and evaluate the predictive mean and standard deviation. To provide a tighter estimate, we use 1000 MC draws in Figure 1a(b).

E Exploring the posterior distribution: Additional results

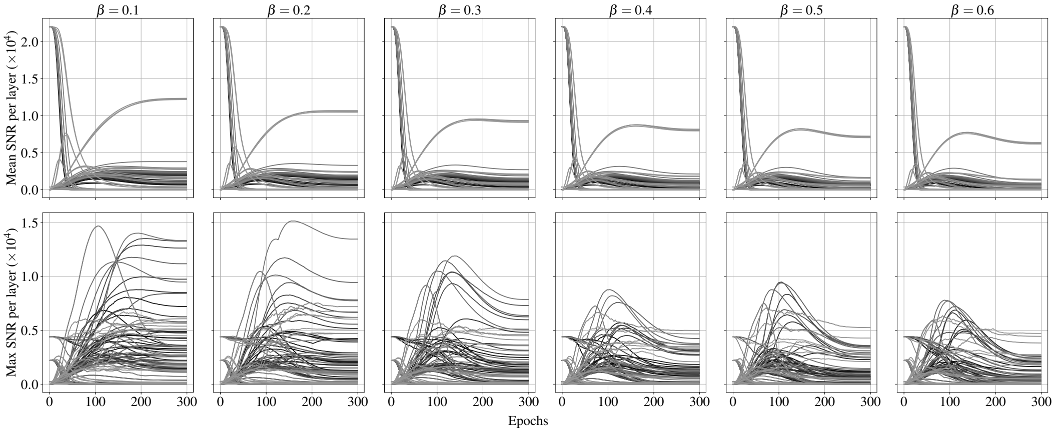


Figure 6: SNR ($|\mu|/\sigma$) with different KL-β schedules. All schedules start from 0.0 and follow a single cycle cosine to end at the β described in the title. All models present similar trajectories over training, but vary slightly for their maximum (bottom row). The values and patterns are mostly identical between all three models. We can see a trend for the growing bundle of lines that achieve smaller values along with increasing β. These lines correspond to the projector and predictor.

In Section 4.2, we looked at how SNR varied across different priors, showing the layerwise SNR is relatively invariant to the choice of prior. In Figure 6, we repeat this analysis using varying values of the KL weight β (with a $\mathcal{N}(0, 1)$ prior). We find that β has a fairly small impact on the SNR values.

F Increasing Gaussian noise

Figure 7 shows posterior predictive variances obtained at three points in the model: after the encoder layer; after the projector, and after the predictor heads (see Figure 1a). Each point represents an image augmented with Gaussian noise; the color of each point represents the strength of the augmentation. We see positive correlation between uncertainty at each location. Moreover, as the amount of noise increases, the average variance increases, as we would expect. Meanwhile, the variation in the variances decreases, as images become increasingly close to pure Gaussian noise, hence, the variance become more concentrated.

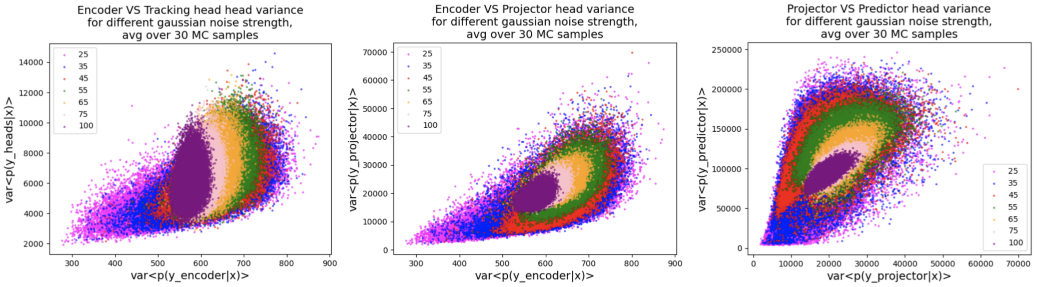


Figure 7: Variance of latent variables from different model parts. The stonger gaussian noise is, the less meaningful features can be extracted from the encoder, hence, the variances of predictive distribution become more concentrated. Each point is based on $M = 50$ weights MC sample (App.D).

G Should the whole network be Bayesian?

Previous work has also questioned whether an entire network needs to be Bayesian (Sharma et al., 2023; Gast & Roth, 2018; Ovadia et al., 2019). A common reason for a partial Bayesian network is due to the incurred memory overhead used to learn the natural parameters of the distribution. While an Isotropic Gaussian BBB doubles parameter counts, the effective memory footprint of the model does not grow proportionally. Typically, the majority of accelerator memory is dominated by activation gradients and not model parameters (Chen et al., 2016). After reparameterization (Mohamed et al., 2020), the effective activation size is equivalent to the Maximum Likelihood Estimation (MLE) case for every layer, thus incurring only a minimal practical overhead for being Bayesian.

In Table 1, we retrain a subset of BYOV models by keeping part of the network Bayesian. In particular, we explore using a point estimate for all LayerNorm layers and the convolutional patcher used in vision transformers. In contrast to previous findings we observe that using a fully Bayesian model (Fig. 2) presents the best performance amongst the class of BBB models.

BBB conversion	Prior Type	β_{start}	β_{end}	top-1 ↑ (student)	top-5 ↑ (student)
No Conv BBB	$\mathcal{N}(0, I)$	0.0	1.0	73.68	91.43
	$\mathcal{N}(\mu_T, I)$			**73.97**	91.52
	$\mathcal{N}(\boldsymbol{\mu}_T, \Sigma_T)$			69.73	88.74
Linear Only BBB	$\mathcal{N}(0, I)$	0.0	1.0	73.62	91.12
	$\mathcal{N}(\boldsymbol{\mu}_T, I)$			74.23	91.48
	$\mathcal{N}(0, I)$	1.0	1.0	72.60	90.92
	$\mathcal{N}(\boldsymbol{\mu}_T, I)$			72.69	91.03
No BBB (baseline)				75.97	92.40

Table 1: All metrics are computed on the in-domain test set and uses the posterior mean, $\boldsymbol{\mu}$, for inference.

H Contributions

All authors contributed to writing this paper, designing the experiments, discussing results at each stage of the project.

Preliminary work Formulation of BBB coupled with SSL developed by Polina Turishcheva, Jason Ramapuram and Sinead Williamson. Idea refined in discussions with Dan Busbridge, Eeshan Dhekane and Russ Webb.

Generalized ELBO formulation Relationship of ELBO to the generalized posterior and related formulations developed by Sinead Williamson (Section 2.1, Section 3).

Pruning Experiments written by Polina Turishcheva in discussions with Russ Webb (Figure 4)-Left.

Briar reliability and ECE analysis Conducted by Polina Turishcheva (Figure 4 - Center, Right and Figure 7).

Layerwise variance and SNR exploration Conducted by Polina Turishcheva in discussions with Sinead Williamson and Jason Ramapuram (Figure 3, Figure 6).

Monte Carlo Variance Estimates Preliminary explorations into Monte Carlo repeats (Figure 5) and estimating predictive distribution done by Polina Turishcheva in discussions with Sinead Williamson and suggestions from Dan Busbridge and Russ Webb (Figure 7). Large sample Monte-Carlo estimate experiment (Figure 1b) and improvements noted in Section 4.1 for top-1 done by Jason Ramapuram.

Data dependent priors Discussions between Jason Ramapuram, Polina Turishcheva, Sinead Williamson and Dan Busbridge led to exploring various priors (Figure 2). Code written by Jason Ramapuram and validated by Eeshan Dhekane.

Should the whole network be Bayesian? Explored by Jason Ramapuram in discussions with Polina Turishcheva and Sinead Williamson (Appendix G).

Implementation details Code for baseline BYOL ViT written by Jason Ramapuram. BYOV implementation written by Polina Turishcheva and Jason Ramapuram. Reviewed by Eeshan Dhekane. Tikz wizardry done by Dan Busbridge.