Sampling and prior selection

2024

Model selection: coherent inference

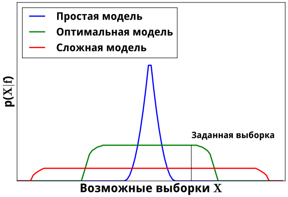
First level: select optimal parameters:

$$\mathbf{w} = \arg\max \frac{p(\mathfrak{D}|\mathbf{w})p(\mathbf{w}|\mathbf{h})}{p(\mathfrak{D}|\mathbf{h})},$$

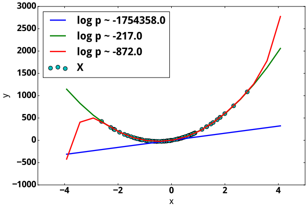
Second level: select optimal model (hyperparameters).

Evidence:

$$p(\mathfrak{D}|\mathbf{h}) = \int_{\mathbf{w}} p(\mathfrak{D}|\mathbf{w})p(\mathbf{w}|\mathbf{h})d\mathbf{w}.$$



Model selection scheme



Example: polynoms

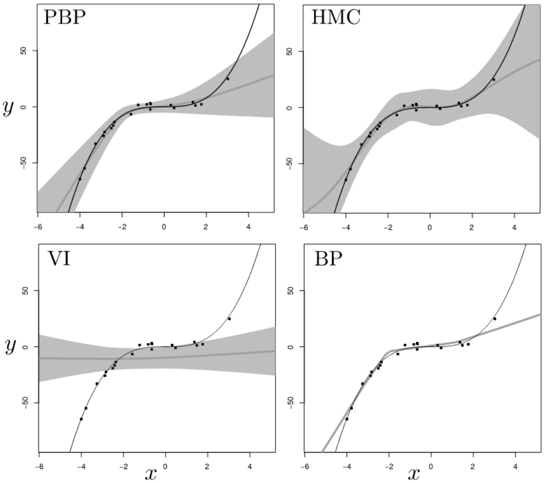
Evidence estimation

$$\mathsf{E}f = \int_{\mathbf{w}} f(\mathbf{w})p(\mathbf{w})d\mathbf{w}.$$

- Laplace approximation
  - Fixed form of approximation distribution
  - Poorly scales
- Variational inference[1]
  - Well scales
  - Can use different forms of approximation distributions
  - Lower bound of evidence => biased
- MC
  - Can use different forms of approximation distributions
  - Approximates well
  - Slow

[1] See the talk from Alexander Kolesov, 2021

VI vs MC



Naive method

$$I = \mathrm{E}f = \int_{w} f(\mathbf{w})p(\mathbf{w})d\mathbf{w}.$$

Approximate:

$$\hat{I} = \frac{1}{N}\sum_{\mathbf{w}\sim p(\mathbf{w})} f(\mathbf{w}).$$

Why this does not work?

Properties

Integral estimation:

- strongly consistent : $\hat{I} \to^{\text{a.s.}} I$
- Unbiased: $\mathrm{E}\hat{I} = I$
- Assymptotically normal;
- $\mathrm{D}\hat{I} = O(\frac{1}{N})$.
- **Challenge:** we need to sample from p.

Inverse transform sampling

Let T be a invertible function from $u \sim \mathcal{U}(0,1)$ to some random variable distribution $p(w)$. Then

$$F_w(t) = p(w \leq t) = p(T(u) \leq t) = p(u \leq T^{-1}(t)) = T^{-1}(u).$$

Therefore $F_u^{-1} = T$.

Example

$$w = \lambda \text{exp}(-\lambda t).$$

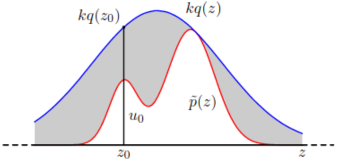
$$F_w(t) = 1 - \text{exp}(-\lambda t).$$

$$F_w^{-1}(u) = -1\frac{1}{\lambda}\text{log}(1-u).$$

Rejection sampling

- Given $p(w)$ (up to normalizing constant)
- Set distribution q
- Set value k so that $kq(w) \geq p(z)$ for all z
- In a loop:
  - Sample $w_0 \sim q$
  - Sample $u \sim \mathcal{U}(0, kq(w_0))$
  - If $u \leq p(w_0)$, use it as a sample from $p(w)$

Core idea: samples u are uniform in a region limited by $p(w)$.



Bishop, 2006

Importance sampling

Consider the case when we cannot sample from $p(w)$, but we can estimate likelihood and want to estimate the integral

$$\mathsf{E}f = \int f(w)p(w)dz.$$

Let q be an auxilary distribution:

$$\mathsf{E}f = \int f(w)p(w)dw = \int f(w)\frac{p(w)}{q(w)}q(w)dw \approx \frac{1}{L}\sum_{l=1}^{L}\frac{p(w^l)}{q(w^l)}f(w^l).$$

MCMC

Basic idea: Sample similar to rejection sampling, but q is a Markov distribution with conditioning on the previous step.
We want the stationary (limiting) distribution to be equal to our $p(w)$.
Sufficient condition

$$p(w')T(w|w') = p(w)T(w'|w).$$

Metropolis-Hastings algorithm

- Sample new $w' \sim q(w|w^t)$.
- Accept with probability $A(w'|w^t) = \min\left(1, \frac{p(w')q(w^t|w')}{p(w^t)q(w'|w^t)}\right)$.
- If accepted: $w^{t+1} = w'$,
- Otherwise: $w^{t+1} = w^t$.

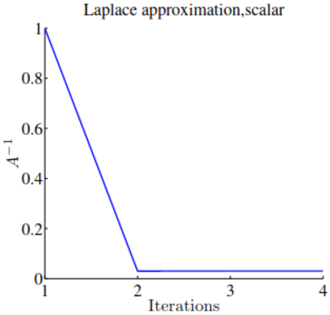
Sufficient condition is satisfied:

$$p(w')T(w|w') = p(w)T(w'|w) = p(w')T(w'|w^t) = p(w')q(w'|w^t)A(w'|w^t) =$$

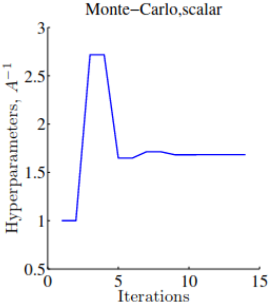
$$= p(w^t)q(w^t|w')A(w^t|w').$$

- Samples are correlated. We can decorrelate sample using each k sample.
- Works better in high-dimmensional settings than rejection sampling.
- Good choice of q is the main challenge for the algorithm.

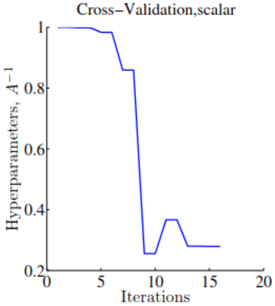
Hyperparameter selection for linear model



(a) Laplace approximation

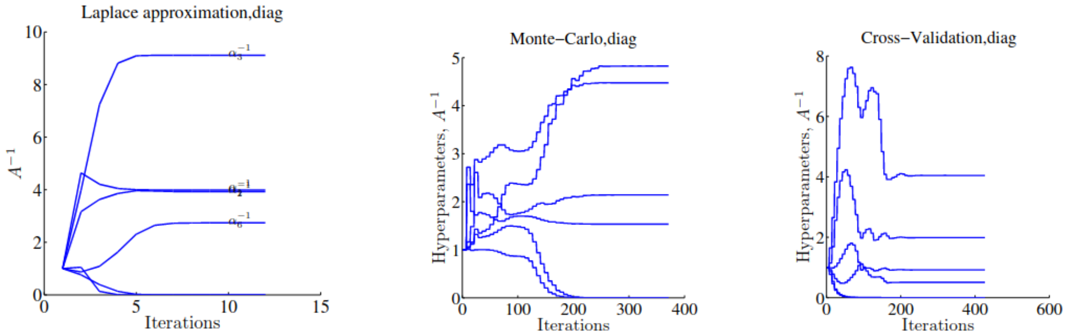


(b) Monte-Carlo



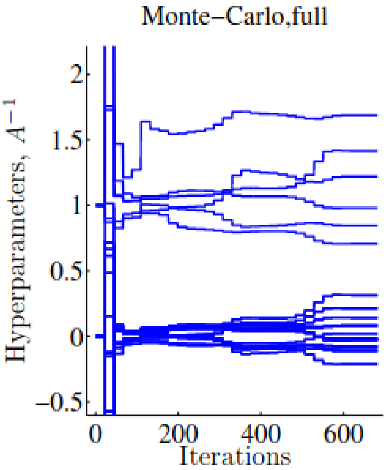
(c) Cross validation

Hyperparameter selection for linear model

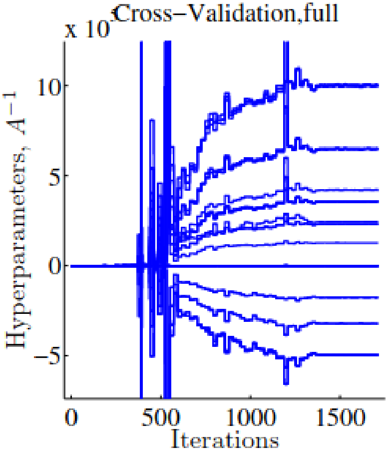


(a) Laplace approximation (b) Monte-Carlo (c) Cross validation

Hyperparameter selection for linear model



(a) Monte-Carlo



(b) Cross validation

Autoencoder: generative model?

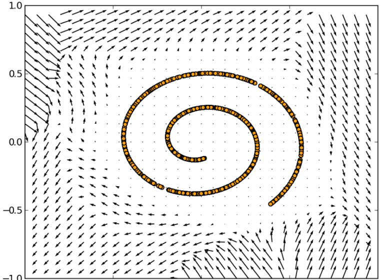
(Alain, Bengio 2012): consider regularized autoencoder:

$$||\mathbf{f}(\mathbf{x}, \sigma) - \mathbf{x}||^2,$$

where σ is a noise level.
Then

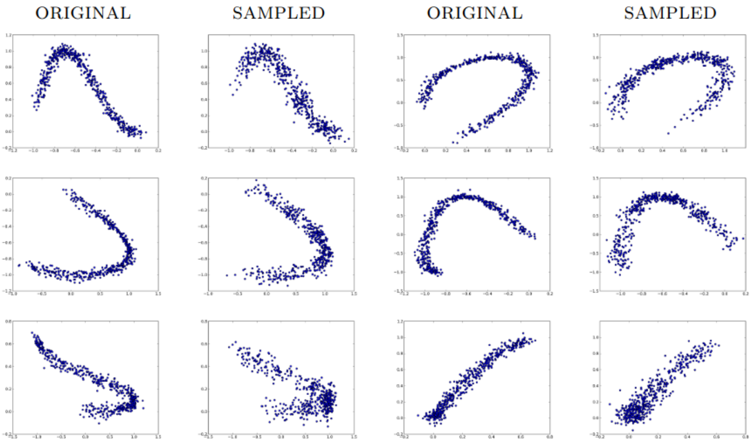
$$\frac{\partial \log p(x)}{\partial x} = \frac{||\mathbf{f}(\mathbf{x}, \sigma) - \mathbf{x}||^2}{\sigma^2} + o(1) \text{ when } \sigma \to 0.$$

Vector field induced by reconstruction error



Autoencoder for sampling

$$A = \frac{p(x^*)}{p(x)} = \exp\left(E(x) - E(x^*)\right) \approx \frac{\partial E(x)}{\partial x}^{\mathsf{T}} (x^* - x) + o(||x - x^*||).$$



Optimization of q

Distribution q can be set using neural networks.

- **Main requirements:** existance of $p(x|x'), p(x'|x) \rightarrow$ the distribution must be invertible.
- Neural network in a form of $\mathbf{f}(\mathbf{x}, \mathbf{w}) = \mathbf{x} + \mathbf{g}(\mathbf{x}, \mathbf{w})$ is a flow and invertible.

Optimization variants:

- Entropy * Acceptance rate (Li et al., 2020)
- GAN between empirical distribution and q (Song et al., 2017).

How Good is the Bayes Posterior in Deep Neural Networks Really?

- Wenzel et al., 2020: model performance increases if instead of simple posterior we use "cold" posterior:

$$p(\mathbf{w}|\mathfrak{D}) = \exp(-U(\mathbf{w})/T), T < 1.$$

$$U = -\log p(\mathfrak{D}|\mathbf{w}) - \log p(\mathbf{w}).$$

- Multiple hypothesis were considered:
  - Inaccurate MC simulation (no)
  - Minibatch from noise causes bad sampling (no)
  - $T \to 0$ reduces variance and gives better performace (no)
  - Dirty likelihood performance (operations like batch-norm breaks likelihood) (no)
  - Bad prior (no)
  - Inductive bias in SGD (no)
  -

What Are Bayesian Neural Network Posteriors Really Like?

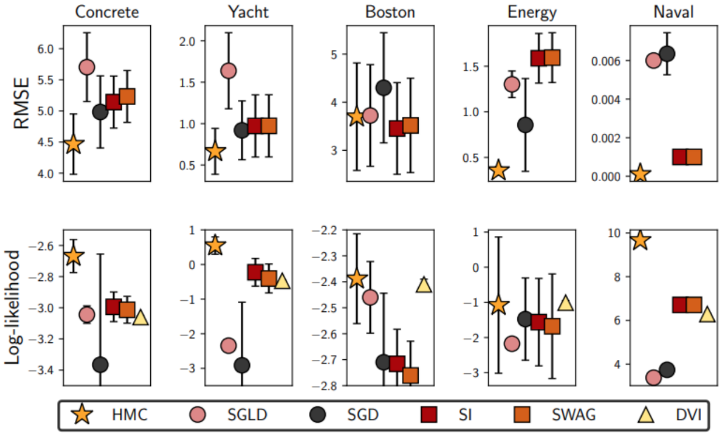
Izmailov et al., 2021:[2]

- HMC for posterior distribuition estimation for deep models on some standard datasets.
- Resources: 512 TPU

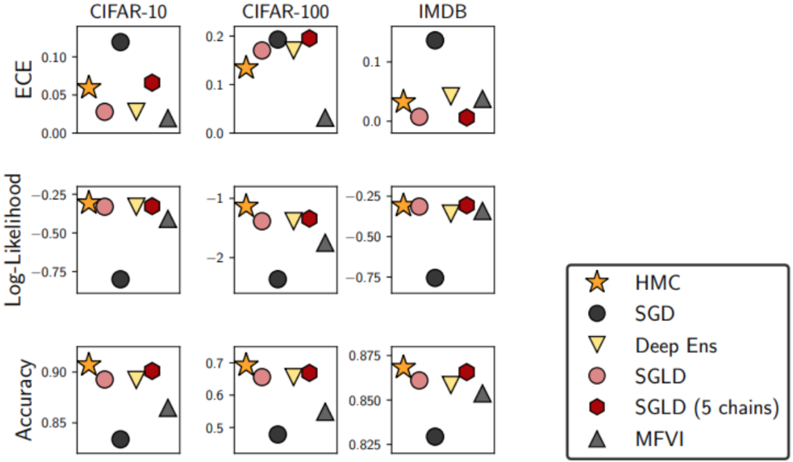
[2] https://docs.google.com/presentation/d/1WPjqKw3b-TpPSaHcwhqFsuAE575_nDigt5SVoP3CoNI/edit?usp=sharing

What Are Bayesian Neural Network Posteriors Really Like?

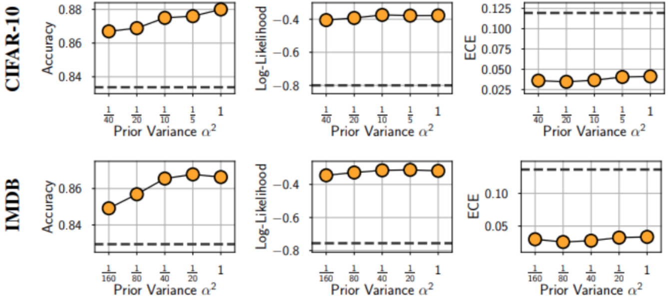
BNN evaluation: UCI



BNN evaluation: CIFAR and IMDB



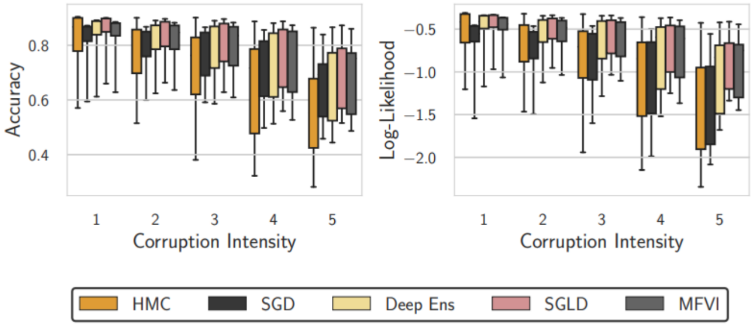
Effect of priors



HMC BNNs are fairly robust to Gaussian prior variance.

What Are Bayesian Neural Network Posteriors Really Like?

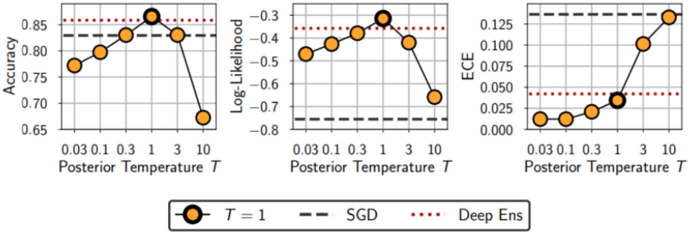
Train on CIFAR-10, test on CIFAR-10-C



What Are Bayesian Neural Network Posteriors Really Like?

Posterior temperature effect

- We have already seen that BNNs can do well at T=1
- What is the effect of T then?



Cold posteriors are not required for good results and in fact can hurt performance!

References

- Bishop C. M., Nasrabadi N. M. Pattern recognition and machine learning. – New York : springer, 2006. – T. 4. – №. 4. – C. 738.
- Kuznetsov M., Tokmakova A., Strijov V. Analytic and stochastic methods of structure parameter estimation //Informatica. – 2016. – T. 27. – №. 3. – C. 607-624.
- Mandt, Stephan, Matthew Hoffman, and David Blei. "A variational analysis of stochastic gradient algorithms."International conference on machine learning. PMLR, 2016.
- Alain, Guillaume, and Yoshua Bengio. "What regularized auto-encoders learn from the data-generating distribution."The Journal of Machine Learning Research 15.1 (2014): 3563-3593.
- Li Z., Chen Y., Sommer F. T. A neural network mcmc sampler that maximizes proposal entropy //arXiv preprint arXiv:2010.03587. – 2020.
- Song J., Zhao S., Ermon S. A-nice-mc: Adversarial training for mcmc //Advances in Neural Information Processing Systems. – 2017. – T. 30.
- Atanov, Andrei, et al. "The deep weight prior."arXiv preprint arXiv:1810.06943 (2018).
- Kolesov A. An adversarial method for neural network fine-tuning for transfer learning problem, Master thesis, 2022.
- Shwartz-Ziv, Ravid, et al. "Pre-train your loss: Easy bayesian transfer learning with informative priors."Advances in Neural Information Processing Systems 35 (2022): 27706-27715.
- Izmailov, Pavel, et al. "What are Bayesian neural network posteriors really like?."International conference on machine learning. PMLR, 2021.