Paper Review

Data augmentation in Bayesian neural networks and the cold posterior effect

Seth Nabarro *et al.*

Marat Khusainov

April 2024

Outline

Motivation

The cold posterior effect

$$\mathrm{P}(\mathbf{w} \mid \mathbf{y}, \mathbf{X}) \propto \mathrm{P}(\mathbf{w})\mathrm{P}(\mathbf{y} \mid \mathbf{w}, \mathbf{X}) \tag{1}$$

Better performance when using a “cold” posterior:

$$\mathrm{Q}(\mathbf{w}) \propto (\mathrm{P}(\mathbf{w})\mathrm{P}(\mathbf{y} \mid \mathbf{w}, \mathbf{X}))^{1/T}, \text{ where } T < 1. \tag{2}$$

- One possible explanation is that the CPE is an artifact of data augmentation.
- It is important to investigate integrating DA with Bayesian neural networks, and to examine the interaction with the CPE.

Methods

To incorporate DA into BNN likelihoods, define the probabilities for each class as being averages over augmentations. Authors choose to either average logits (equal to the neural network outputs, $\mathbf{f}(\cdot;\mathbf{w})$) or predictive probabilities ($\operatorname{softmax} \mathbf{f}(\cdot;\mathbf{w})$),

$$\mathbf{p}_{\text{inv}}\left(\mathbf{x}_i;\mathbf{w}\right) = \mathbb{E}\left[\operatorname{softmax} \mathbf{f}\left(\mathbf{x}_i';\mathbf{w}\right)\right] \tag{3}$$

$$\mathbf{f}_{\text{inv}}\left(\mathbf{x}_i;\mathbf{w}\right) = \mathbb{E}\left[\mathbf{f}\left(\mathbf{x}_i';\mathbf{w}\right)\right]. \tag{4}$$

where expectations over $\mathrm{P}\left(\mathbf{x}_i' \mid \mathbf{x}_i\right), \mathbf{x}_i'$ – augmented input.

The resulting (usually intractable) log-likelihoods are

$$\begin{aligned}
\mathcal{L}_{\text{prob}}^i\left(y_i;\mathbf{w}\right) &= \log \mathrm{P}_{\text{prob}}\left(y_i \mid \mathbf{x}_i,\mathbf{w}\right) \\
&= \log \mathbb{E}\left[\operatorname{softmax}_{y_i} \mathbf{f}\left(\mathbf{x}_i';\mathbf{w}\right)\right] && (5) \\
\mathcal{L}_{\text{logits}}^i\left(y_i;\mathbf{w}\right) &= \log \mathrm{P}_{\text{logits}}\left(y_i \mid \mathbf{x}_i,\mathbf{w}\right) \\
&= \log \operatorname{softmax}_{y_i} \mathbb{E}\left[\mathbf{f}\left(\mathbf{x}_i';\mathbf{w}\right)\right] && (6)
\end{aligned}$$

Methods

Authors show that it is possible to get tight, intuitive and easy to evaluate, multi-sample bounds analogous to those in IWAE [1].

$$\hat{\mathcal{L}}^i_{\text{prob },K}\left(y_i;\mathbf{w}\right) = \log\left(\frac{1}{K}\sum_{k=1}^{K}\text{softmax}_{y_i}\mathbf{f}\left(\mathbf{x}'_{i;k};\mathbf{w}\right)\right),$$
$$\hat{\mathcal{L}}^i_{\text{logits },K}\left(y_i;\mathbf{w}\right) = \log\text{softmax}_{y_i}\left(\frac{1}{K}\sum_{k=1}^{K}\mathbf{f}\left(\mathbf{x}'_{i;k};\mathbf{w}\right)\right). \quad (7)$$

Increasing K reduces the variance and tightens the bounds which eventually become exact as $K\rightarrow\infty$.

$$\mathcal{L}^i_{\text{logits }}\left(y_i;\mathbf{w}\right) = \lim_{K\rightarrow\infty}\hat{\mathcal{L}}^i_{\text{logits },K}\left(y_i;\mathbf{w}\right) \quad (8)$$
$$\mathcal{L}^i_{\text{prob }}\left(y_i;\mathbf{w}\right) = \lim_{K\rightarrow\infty}\hat{\mathcal{L}}^i_{\text{prob },K}\left(y_i;\mathbf{w}\right) \quad (9)$$

[1] Burda et al., 2015, Importance weighted autoencoders.

Methods

The authors propose two types of settings.

- The usual **"full orbit"** setting, where there is a distribution over a very large, or even infinite number of possible augmentations. The full orbit setting necessitates the use of the bound (Eq.7).
- Alternative **"finite orbit"** by restricting the augmentations to a small subset, we can exactly evaluate the log-likelihood. In the finite orbit setting, the distribution over augmented images, $\mathbf{x}_i'$, conditioned on the underlying unaugmented image, $\mathbf{x}_i$, can be written as

$$\mathrm{P}\left(\mathbf{x}_i' \mid \mathbf{x}_i\right) = \frac{1}{K} \sum_{k=1}^{K} \delta\left(\mathbf{x}_i' - a_k\left(\mathbf{x}_i\right)\right), \tag{10}$$

and a_k is a function that applies the k th fixed augmentation.

Results

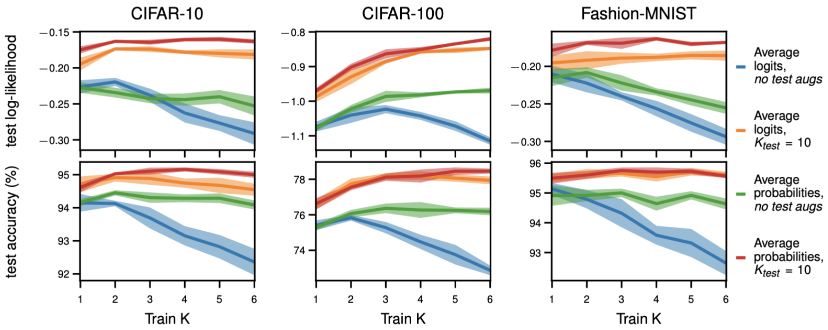

Figure: Comparison of averaging logits and probabilities for different values of K_{train} , and using $K_{\text{test}} = 10$ vs. using no test-time augmentations. Here, we use ResNet18 with SGD (i.e. no Bayesian inference). We use only full orbit to decouple K_{train} from K_{test} .

Results

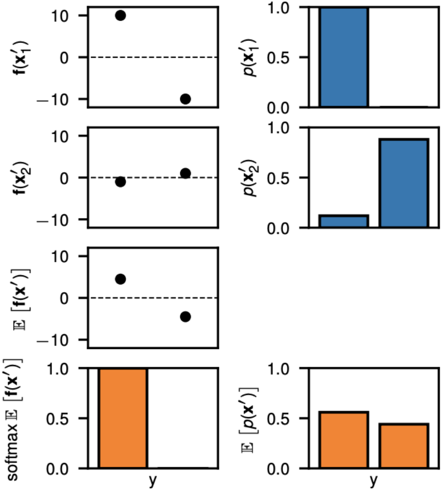

Figure: Example effect of averaging logits against averaging probabilities.

Results

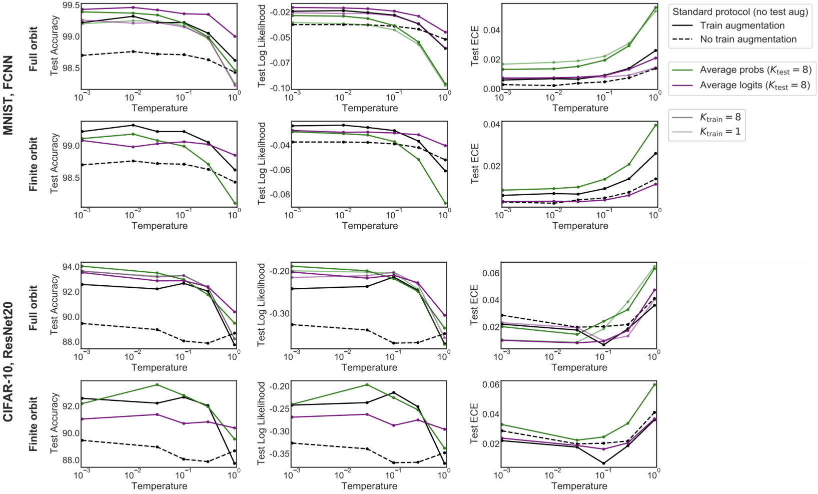

Figure: The cold posterior effect for different DA setups.

Conclusion

- Shown how DA can be properly incorporated into a model suitable for BNN inference, by deriving a lower-bound on the log-likelihood of the augmentation averaged network output.

- Empirically, seen that the CPE persists even when using our principled DA formulation, shown that the CPE disappears without DA.