Notes on Bayesian Neural Networks

Dimitrios Tanoglidis

October 2021

Some notes on the theory and practice of Bayesian Neural Networks are presented. Basic Bayesian inference is reviewed, together with the methods to approximately sample from the posterior.

1 Neural Networks with Bayesian Inference

Let us start by defining the training dataset, $\mathcal{D} = (X, Y)$, which consists of features $X = \{\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N}\}$, and labels $Y = \{\mathbf{y_1}, \mathbf{y_2}, \ldots, \mathbf{y_N}\}$.

Standard neural networks map the inputs (features) to outputs through a series of nodes, arranged in layers, where the output of each node is a weighted sum of all the nodes of the previous layers, that subsequently pass through a non-linear activation function (like ReLU or sigmoid).

More complicated architectures exist, like CNNs that connect each node to only a limited number of nodes of the previous layer, but in all cases the weights (which are learned through the training process) are single values (point estimates). In other words, the weights, W, of traditional NNs are deterministic.

In **Bayesian Neural Networks** (BNNs), the point estimates are replaced by appropriate probability distributions over the weights, that can be subsequently used to estimate uncertainties in these weights and in the predictions.

The objective of training a BNN is to find the posterior distribution of the weights, given the training dataset, $p(w|X, Y)$. For that reason, according to the Bayesian recipe, a prior belief on the distribution of the weights, $p(w)$, as well as the likelihood $p(Y|X, w)$ (that connects the predictions with the learned weigths w), are specified. Then the posterior can be calculated using Bayes' theorem:

$$p(w|X, Y) = \frac{p(Y|X, w)p(w)}{p(Y|X)} \tag{1}$$

where $p(Y|X) = \int P(Y|X, w)p(w)dw$, is the **evidence**.

If the posterior has been computed, the probability distribution for the parameter y^* of a new data point with feature vector x^* is:

$$p(y^*|x^*, X, Y) = \int p(y^*|x^*, w)p(w|X, Y)dw \tag{2}$$

2 Variational Inference and the ELBO

Calculating the above posterior (1) is an intractable problem, because of the evidence factor (which has to be integrated over all values of weights in a multi-dimensional space, which requires exponential time).

Approximate solutions should be developed. A common one in many problems in (astro)physics is to sample the posterior through MCMC methods. Another one is to approximate the exact posterior distribution $p(w, \mathcal{D})$ (from now on I change X, Y to $\mathcal{D}$, for ease of writing the expressions; whenever not clear, the explicit form will be introduced again), with a well behaved, computationally tractable one, $q(w|\theta)$. The goal then of **variational inference** is to find those parameters θ such that the approximate posterior, $q(w|\theta)$, matches the true posterior, $p(w|\mathcal{D})$, as accurately as possible.

A measure of (dis)similarity between of the two distributions is the **Kullback-Leibler** (KL) divergence:

$$\boxed{\mathrm{KL}(q(w|\theta)||p(w|\mathcal{D})) \equiv \int q(w|\theta) \log \frac{q(w|\theta)}{p(w|\mathcal{D})} dw.} \tag{3}$$

Thus the inference problem can be rephrased as an optimization problem:

$$q(w|\hat{\theta}) = \mathrm{argmin}_\theta \mathrm{KL}(q(w|\theta)||p(w|\mathcal{D})) \tag{4}$$

Why is that better? To find out, let's use again Bayes' theorem to rewrite the posterior in the expression for the KL divergence:

$$\mathrm{KL}(q(w|\theta)||p(w|\mathcal{D})) = \int q(w|\theta) \log \frac{q(w|\theta)p(\mathcal{D})}{p(\mathcal{D}|w)p(w)} dw \tag{5}$$

$$= \int q(w|\theta) \log \frac{q(w|\theta)}{p(w)} dw - \int q(w|\theta) \log p(\mathcal{D}|w) dw + \log p(\mathcal{D}) \int q(w|\theta) dw \tag{6}$$

$$= \mathrm{KL}(q(w|\theta)||p(w)) - \mathbb{E}_{q(w|\theta)} \log p(w|\mathcal{D}) + \log p(\mathcal{D}) \tag{7}$$

Since the evidence does not depend on the parameters θ, minimizing the KL divergence is equal to minimizing the so-called **variational free energy**:

$$\boxed{\mathcal{F}(\mathcal{D}, \theta) = \mathrm{KL}(q(w|\theta)||p(w)) - \mathbb{E}_{q(w|\theta)} \log p(w|\mathcal{D}),} \tag{8}$$

so we don't have to calculate the evidence.

While it is not really necessary for the definition of the optimization of the problem (I think), usually the above is expressed in terms of the **Evidence Lower Bound (ELBO)**.

Specifically, the ELBO is defined as the negative of the variational free energy, so:

$$\mathrm{ELBO} = \mathbb{E}_{q(w|\theta)} \log p(w|\mathcal{D}) - \mathrm{KL}(q(w|\theta)||p(w)) \tag{9}$$

Thus :

$$\mathrm{KL}(q(w|\theta)||p(w|\mathcal{D})) = -\mathrm{ELBO} + \log p(\mathcal{D}) \Rightarrow \log p(\mathcal{D}) = \mathrm{ELBO} + \mathrm{KL}(q(w|\theta)||p(w|\mathcal{D})). \tag{10}$$

Since $\mathrm{KL}(q(w|\theta)||p(w|\mathcal{D})) \geq 0$ (property of the KL divergence), we see that ELBO is indeed a lower bound of the evidence.

Thus, minimizing the KL divergence between the approximate and true posterior is the same as maximizing the ELBO.

3 Training a BNN

4 Epistemic and Aleatoric uncertainties

5 Tensorflow Probability and friends