Variational Inference for Bayesian Neural Networks

Juan Maroñas
jmaronasm@gmail.com

PRHLT Research Center,
Universitat Politècnica de València

April 2019

1 Introduction

In this document I show how can we evaluate the predictive distribution in a classification scenario when using Neural Networks to parameterize the likelihood model in a Bayesian setting. In this case rather than using Markov Chain Monte Carlo algorithms to draw samples from the posterior distribution, I will be using the commonly known Variational Inference approach which aims at recovering a function[1] that approximates the unknown posterior distribution. For further information please see [Jordan et al., 1999, Bishop, 2006].

Rather than a technical paper describing the mathematical foundations I will only describe the model used. In this case let $q_\phi(w) \in \mathcal{Q}$ be the variational function from the family of $\mathcal{Q}$ approximating functions. Variational inference optimization aims at recovering $q^*_\phi(w)$ such that the following training criteria is minimized:

$$q^*_\phi(w) = \underset{q_\phi \in \mathcal{Q}}{\operatorname{argmin}} \ \mathrm{DKL}\{q_\phi(w)||p(w|x,t)\} \tag{1}$$

where DKL is the Kullback-Lieber divergence; w are the Neural Network parameters: bias, weight, kernels...; and $p(w|x,t)$ is the unknown posterior distribution we want to approximate, in terms of the DKL metric. As this is a divergence and not a distance (i.e it is not symetric) there are another family of algorithms that aims at minimizing the reverse divergence [Minka, 2001, Hernandez-Lobato et al., 2016].

[1] I will be covering the finite parametric approach, as the same training criteria holds for Gaussian Processes.

2 The Model

2.1 The basic model

In this approach $\mathcal{Q}$ is chosen to be the factorized Normal Distribution $N(w|\phi)$ where ϕ is the set of variational parameters $\phi = \{(\mu_i, \sigma_i)\}_{i=1}^{|W|}$ and W stands for the set of parameters that parameterize the neural network.

We can minimize the above intractable cost function by maximization of the well known and tractable *Evidence Lower Bound* (ELBO) which is a lower bound on the marginal log-likelihood. Thus the final training criteria is.

$$\begin{aligned} \text{ELBO} &= \log\, p(t|x) - \text{DKL}\{q_\phi(w)||p(w|x,t)\} \\ q_\phi^*(w) &= \underset{q_\phi(w)\in\mathcal{Q}}{\text{argmin}}\, \text{ELBO} \end{aligned} \tag{2}$$

Thus, maximizing the ELBO w.r.t the variational parameters will minimize the proposed criteria (the marginal log likelihood remains the same as it is a constant provided there are no hyperparameters being tunned, something that does happend when this criteria is used in sparse GPs).

As in the rest of the models we choose a Standard Normal prior over the parameters:

$$p(w) = \prod_{i=1}^{|W|} \frac{1}{\sqrt{2\pi}\sqrt{\sigma_w^2}} \exp \frac{1}{2} \cdot (w_i - \mu_w)^2 \tag{3}$$

With this, ELBO we be written as:

$$\text{ELBO} = \underset{w\sim q_\phi(w)}{\mathbb{E}}[\log p(t|x,w)] - \text{DKL}\{q_\phi(w)||p(w)\} \tag{4}$$

and use stochastic optimization using M Monte Carlo samples:

$$\text{ELBO} = \frac{1}{M}\sum_{i=1}^{M}[\log p(t|x,w_i)] - \text{DKL}\{q_\phi(w)||p(w)\}; w_i \sim q_\phi(w) \tag{5}$$

In order to reduce the variance of the estimator I use the reparameterization trick [Kingma and Welling, 2014, Rezende et al., 2014] and the expression of the ELBO in equation 4. When the DKL can be computed analitically this expressions leads to a lower variance gradient estimator. For the chosen family of variational and prior distributions this DKL can be computed in closed form:

$$\text{DKL}\{q_\phi(w)||p(w)\} = -\frac{1}{2}\sum_{i=1}^{|W|}(1 + \log\sigma_i^2 - \mu_i^2 - \sigma_i^2) \tag{6}$$

2.2 Local Reparameterization

The good convergence of the stochastic algorithm always dependes on the variance of the estimator. That is the key-point of using local reparameterization [Kingma et al., 2015]

Basically, it can be shown that by resampling a set of parameters from the variational distribution for each training point in the batch size we can considerably reduce the variance of the gradients estimates. However, this is very costly and cannot be easily parallelized. If we have a batch of size 100 in a 784 dimensional input space, and we want to project to a 1000 dimensional hidden space, then we need to sample and forward 100 with 784x1000 matrices.

On the other side, note that a linear combination of N independent Gaussian distributions induce another Gaussian distribution with parameters:

$$\mu = \sum_{i=1}^{N} \lambda_i \mu_i \qquad \sigma^2 = \sum_{i=1}^{N} \lambda_i^2 \sigma_i^2 \tag{7}$$

where λ_i are the coefficients of the linear combination. In our case of study, is it straightforward to apply the same concept to induce the distribution over the pre-activation of the next layer, in a neural network (remember the variational distributions are parameterized by Gaussian). Following the standard notation for the elements of a matrix and assigning λ_{kj} to the *j-th* dimension of the *k-th* sample in a batch (x_{kj}), the distribution induced over the *k-th* pre activation is Gaussian with parameters:

$$\begin{aligned} \mu_{kj} &= \sum_{i=1}^{N} x_{ki} \mu_{ij} + \mu_k \\ \sigma_{kj}^2 &= \sum_{i=1}^{N} x_{ki}^2 \sigma_{ij}^2 + \sigma_k^2 \end{aligned} \tag{8}$$

where μ_k and σ_k^2 stands for the parameters of the variational distribution for the bias term. The bias is added with coefficient $\lambda = 1$.

Once we parameterize the distribution over the pre-activation we can (in just one call to the random generator) generate different parameters per data point and thus achieve our goal. The only thing that differes from the baseline model is when and from which to sample, the rest is the same.

2.3 Training criteria

Putting all this in common and by specifying a k-categorical distribution for the likelihood, the final training criteria is given by:

$$q^*_\phi(w) = \underset{q_\phi(w) \in \mathcal{Q}}{\text{argmin}} \frac{1}{M} \sum_{i=1}^{M} \text{CE}(t, f_{\theta_i = g(\epsilon_i, \phi)}(x)) + \text{DKL}\{q_\phi(w) || p(w)\}; \epsilon_i \sim \mathcal{N}(0, I) \tag{9}$$

where $f_{\theta_i}(x)$ denotes the neural network parameterized by the set of sampled parameters θ_i(parameters of the neural network in the standard model and preactivations when using local reparameterization) that are obtained by transforming a sample ϵ from standard Normal distribution using a function $g()$ that takes as input both the sample and the variational parameters. CE denotes the cross entropy loss. The sampling process using this setting is thus given by:

$$\begin{aligned} &\epsilon \sim \mathcal{N}(0, I) \\ &\theta = g(\epsilon, \phi = (\mu, \sigma)) = \mu + \epsilon \cdot \sigma \end{aligned} \tag{10}$$

which is the magic reparameterization trick.

References

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

Jose Hernandez-Lobato, Yingzhen Li, Mark Rowland, Thang Bui, Daniel Hernandez-Lobato, and Richard Turner. Black-box alpha divergence minimization. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1511–1520, New York, New York, USA, 20–22 Jun 2016. PMLR. URL `http://proceedings.mlr.press/v48/hernandez-lobatob16.html`.

Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Mach. Learn.*, 37(2):183–233, November 1999. ISSN 0885-6125. doi: 10.1023/A:1007665907178. URL `https://doi.org/10.1023/A:1007665907178`.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL `http://arxiv.org/abs/1312.6114`.

Durk P Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2575–2583. Curran Associates, Inc., 2015. URL `http://papers.nips.cc/paper/5666-variational-dropout-and-the-local-reparameterization-trick.pdf`.

Thomas P. Minka. Expectation propagation for approximate bayesian inference. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, UAI '01, pages 362–369, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1-55860-800-1. URL `http://dl.acm.org/citation.cfm?id=647235.720257`.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China, 22–24 Jun 2014. PMLR. URL `http://proceedings.mlr.press/v32/rezende14.html`.