Stochastic Bayesian Neural Networks

Abhinav Sagar*
Vellore Institute of Technology
Vellore, Tamil Nadu, India
abhinavsagar4@gmail.com

Abstract

Bayesian neural networks perform variational inference over weights but calculation of the posterior distribution remains a challenge. Our work builds on variational inference techniques for bayesian neural networks using the original Evidence Lower Bound. In this paper, we present a stochastic bayesian neural network in which we maximize Evidence Lower Bound using a new objective function which we name as Stochastic Evidence Lower Bound. We tested our approach on 5 publicly available UCI datasets using test RMSE and log likelihood as the evaluation metrics. We demonstrate that our work not only beats the previous state of the art algorithms but also allows uncertainty quantification and is scalable to larger datasets.

1 Introduction

Neural Networks have been highly successful in a variety of domains including computer vision, natural language processing, recommender systems, reinforcement learning etc. They have considerably surpassed previous state of the art algorithms in machine learning which require manual feature engineering. However, applying them to sensitive domains like self driving cars, healthcare etc is still a major challenge. This is due to the fact that not only we need predictions made by the model but also with how much certainty it is making those predictions. This is why bayesian neural networks have gained a huge traction recently as they combine the flexibility, scalability and predictive performance with a probabilistic approach to measure uncertainty.

The challenge with bayesian neural networks is that we have to specify a meaningful prior distribution in advance and also the calculation of posterior distribution is often intractable. A good prior distribution is difficult to get as the relationship between the weights of the network (Graves et al, 2011) and the output is non linear in nature while the calculation of the posterior requires doing an integral which is often intractable in nature (Welling et al, 2016).

To avoid the above two difficulties, there has been considerable work done showing that as the width of a BNN was increased, the limiting distribution turns out to be a Gaussian process (Lee et al 2018). However still the relationship of BNN with GP remains unclear as it fails to match the predictions made by GP. This could be due to the fact that there are a lot of kernels with different structured approximations and finding the one which best suits the task at hand is not straight forward.

In this paper, we perform variational inference with a new type of model architecture which we named as stochastic bayesian neural network. The update step is similar to the traditional backpropagation method in our method. In this method, a BNN is trained to produce a custom distribution with small KL-divergence with the true posterior. We do this by maximizing the Evidence Lower Bound (ELBO) by sampling based approximation. We specify stochastic process priors which are by their inherent nature rich in structured dependencies between function values. Using this method, we can model

*Website of author - https://abhinavsagar.github.io/

Preprint. Under review.

various structures including periodicity and smoothness (Sun et al, 2018). Thus stochastic bayesian neural networks combine the advantage of GP with the fact that posterior distribution becomes compuatable. We show that this approach beats the previous state of the art on regression datasets.

2 Related Work

Bayesian Neural Networks using Variational Inference approach has a rich history first being applied by (Hinton et al, 1993). The work was later extended by (Graves et al, 2011) using gaussian priors using covariance estimates. Later (Welling et al, 2013) proposed Variational Autoencoders for generative modelling using reparameterization technique. More work from (Bae et al, 2018) have used gaussian variational posteriors while (Welling et al, 2017) have used normalizing flows for computing the posterior distributions. (Gal et al, 2017) have shown that dropouts can be approximated as an ensemble of neural networks in a bayesian setting. Neural networks with dropout were also interpreted as BNNs (Gal & Ghahramani, 2016; Gal et al., 2017). Local reparameterization trick (Kingma et al., 2015) proposed a new perspective by adding an additional parameter in the latent space after the encoder.

Stochastic variational inference uses a new technique by using update rules which resemble ordinary backpropagation (Graves, 2011; Blundell et al., 2015). The challnege with this approach is computing the posterior distributions is difficult as it is intractable in nature (Louizos & Welling, 2016; Zhang et al., 2018; Shi et al., 2018a). Some of the popular choices for priors are gaussian, gaussian mixture distributions etc. Other priors, including log-uniform priors (Kingma et al., 2015; Louizos et al., 2017) and horseshoe priors (Ghosh et al., 2018; Louizos et al., 2017) have also been used successfully.

One common approach used in all of the previous papers is that they used priors over the model parameters. The posterior distribution resulting is often intractable and also weight space distributions are difficult to characterize. In this paper, we have used an alternative approach to automatically compute the prior using a well known theory known as stochastic process. The resulting neural networks which are still based on variational inference techniques are named as Stochastic Bayesian Neural Networks. Our method makes it possible to specify a range of priors and in particular stochastic process priors as has been done in gaussian process.

We summarize our main contributions as follows:

• An approach to take advantage of flexibility, scalability, predictive performance and a probabilistic approach to measure uncertainty of variational inference techniques on regression problems

• A theoretical analysis of our approach named Stochastic Bayesian Neural Network which uses an alternative lower bound which we call SELBO backed by stochastic process.

• Evaluation on the UCI dataset using test RMSE and log likelihood as the evaluation metrics shows we outperform all state-of-the-art methods.

3 Background

3.1 Variational Inference

Bayesian neural networks are defined in terms of priors on weights and the likelihood of the observation. The goal in variational inference techniques is to maximize the ELBO with the goal of fitting an approximate posterior distribution (Blundell et al, 2015). Bayes by Backprop uses a fully factorized Gaussian approximation while computing the posterior distribution (Blundell et al, 2015). The gradients of ELBO can be computed by backpropagation using the local reparameterization trick by computing the gradients and using it for updates(Welling et al, 2013).

Bayes theorem is used for finding the posterior, given the prior, evidence and likelihood using Equation 1:

$$p(z \mid x) = \frac{p(x \mid z)p(z)}{p(x)} \tag{1}$$

However computation of the posterior distribution is infeasible due to the intractable integral in the likelihood term. This is where variational inference techniques come to rescue by converting the equation to an optimization problem between the prior and posterior distributions. For measuring the difference between two probability distributions p and q, KL divergence is defined in Equation 2:

$$D_{KL}(q(x)\|p(x)) := E_{\sim q}[\Delta I] = \int (\Delta I) q(x) dx = \int q(x) \log \left(\frac{q(x)}{p(x)} \right) dx \tag{2}$$

The priors in variational inference techniques are chosen on the basis of computational convenience.

3.2 Variational Autoencoders

VAEs are a family of generative models which use an encoder-decoder architecture and have recently been used in a range of applications like generating images, generating music, recommender systems etc. The encoder converts the sampling distribution to a latent space in the form of mean and variance vectors, while the decoder reconstructs the original sample using both reconstruction error and the KL divergence between the prior and posterior distributions. Let posterior distribution in encoder be defined as $q(z|x)$, weights by θ and encoder as $q\theta(z|x)$. Let the likelihhod function in decoder be given as $p\phi(x|z)$., weights by ϕ.

The KL divergence between the approximate and the real posterior distributions is defined in Equation 3:

$$D_{KL}\left(q_\theta\left(z \mid x_i\right) \| p\left(z \mid x_i\right)\right) = -\int q_\theta\left(z \mid x_i\right) \log \left(\frac{p\left(z \mid x_i\right)}{q_\theta\left(z \mid x_i\right)} \right) dz \geq 0 \tag{3}$$

The above equation can be converted to an optimization problem as shown in Equation 4:

$$\log p\left(x_i\right) \geq -D_{KL}\left(q_\theta\left(z \mid x_i\right) \| p(z)\right) + E_{\sim q_\theta(z|x_i)}\left[\log p_\phi\left(x_i \mid z\right)\right] \tag{4}$$

The right hand side of the above equation is known as the Evidence Lower Bound (ELBO). The goal is to maximize the ELBO which maximizes the log probability. The first term in the above equation denotes the KL divergence between the true and approximate posterior distributions while the second term denotes the reconstruction error.

4 Proposed Method

Our method can be cast as two player zero sum game analogous to a generative adversarial network (GAN) (Goodfellow et al., 2014). Let the dataset be defined by $\mathcal{D}$, variational posterior by $g()$, prior by p, weight by λ and sampling distribution s for random measurement points.

In this work, we have used a sampling based approach. The network needs to match the prior distribution both near the training data and the test data where predictions are required. This is shown in Equation 5 and Equation 6 where X denotes the M samples independently drawn from c.

$$Sample\ Points\ \mathbf{X}^M \sim s \tag{5}$$

$$\mathbf{f}_i = g\left(\left[\mathbf{X}^M\right]; \theta\right), i = 1 \cdots k \tag{6}$$

Next the network is trained using stochastic gradient descent as shown in Equation 7:

$$\Delta = \frac{1}{k} \frac{1}{|D|} \sum_i \sum_{(x,y)} \nabla_\theta \log p\left(y|\mathbf{f}_i(x)\right) \tag{7}$$

Finally Adam optimizer is used for updating the posterior distribution using the prior distribution and the likelihood in every iteration until the distribution converges. This step is shown in Equation 8:

$$\phi \leftarrow Optimizer\left(\theta,\ \lambda\Delta\right) \tag{8}$$

Here λ is a regularization parameter which is tuned using bayesian optimization techniques. The optimal value of λ was found to be 0.24.

4.1 Stochastic Evidence Lower Bound (SELBO)

In our technique, we use a stochastic prior which can be any distribution including the well known Gaussian Process. Here we consider the neural network with stochastic weights and stochastic bias. We sample a function by sampling a random noise vector for some function. This sampling in turns helps in uncertainty quantification by maximizing the Stochastic Evidence Lower Bound (SELBO). The difference with the original ELBO is that in our case the distribution over the weights have been replaced by that over functions.

The KL term here represents the KL divergence between two stochastic processes instead of the earlier approach in which they were over two distributions. The computation of the KL-divergence between stochastic processes requires doing an integral which can be intractable in nature depending on the problem.

4.2 The Algorithm

Next, we present our algorithm used in this paper. In every iteration, we sample a mini batch of training data D and random points X from a distribution c. We forward the sample through a network $g(\phi)$ which defines the posterior distribution. The goal is to maximize the objective function defined by which we name as Stochastic Evidence Lower Bound as shown in Equation 5:

$$\frac{1}{|\mathcal{D}|}\sum_{(\mathbf{x},y)\in\mathcal{D}_\theta}\mathrm{E}_{q_\theta}[\log p(y|f(\mathbf{x}))] - \lambda\mathrm{KL}\left[q\left(\mathbf{f}^{\mathcal{D}}\right)\|p\left(\mathbf{f}^{\mathcal{D}}\right)\right] \tag{9}$$

Here λ is a regularization hyperparameter which needs to be tuned carefully to avoid overfitting.

Algorithm 1: Stochastic Bayesian Neural Networks (SBNN)

Dataset $\mathcal{D}$, variational posterior $g()$, prior p, weight λ
Sampling distribution s for random measurement points
while θ *not converged* **do**
  Sample Points $\mathbf{X}^M \sim s$
  $\mathbf{f}_i = g\left(\left[\mathbf{X}^M\right];\theta\right), i = 1\cdots k$
  $\Delta = \frac{1}{k}\frac{1}{|D|}\sum_i\sum_{(x,y)}\nabla_\theta \log p\left(y|\mathbf{f}_i(x)\right)$
  $\phi \leftarrow Optimizer\left(\theta,\ \lambda\Delta\right)$
end

4.3 Hyperparameters

The hyperparameters used in our model are specified in Table 1.

Table 1: Hyperparameters details

Parameter	Value
Batch Size	16
Optimizer	Adam
Learning Rate	0.0002

5 Results

Next we show our results in Table 1. We have used 5 publicly available UCI datasets for regression and have used two evaluation metrics - test RMSE and log likelihood for testing.

Table 2: Averaged test RMSE and log-likelihood for the regression benchmarks

Dataset	BBB	Noisy K-FAC	SBNN	BBB	Noisy K-FAC	SBNN
Boston	3.171±0.149	2.742±0.125	2.424±0.112	-2.602±0.031	-2.446±0.029	-2.296±0.042
Concrete	5.678±0.087	5.019±0.127	5.003±0.107	-3.149±0.018	-3.039±0.025	-3.016±0.015
Energy	0.565±0.018	0.485±0.023	0.408±0.019	-1.500±0.006	-1.421±0.005	-0.824±0.017
Wine	0.643±0.012	0.637±0.011	0.653±0.005	-0.977±0.017	-0.969±0.014	-1.025±0.014

6 Conclusions

In this paper, we investigated a new technique for training bayesian neural networks using stochastic processes. We proposed a new lower bound using variational inference techniques which we named as Stochastic Evidence Lower Bound. We trained the neural network using gradient descent algorithms by sampling a mini batch of data in every iteration. We show that our work using evaluation metrics test RMSE and log likelihood beats the previous state of the art on 5 public UCI datasets on regression problems. This approach allows estimating uncertainties and is also scalable to large datasets.

Acknowledgments

We would like to thank Nvidia for providing the GPUs.

References

1. Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. arXiv preprint arXiv:1701.04862, 2017.

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

3. Arthur Asuncion and David Newman. UCI machine learning repository, 2007.

4. Stefan Depeweg, José Miguel Hernández-Lobato, Finale Doshi-Velez, and Steffen Udluft. Uncertainty decomposition in Bayesian neural networks with latent variables. arXiv preprint arXiv:1706.08495, 2017.

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

6. Daniel Flam-Shepherd, James Requeima, and David Duvenaud. Mapping Gaussian processes prior to Bayesian neural networks. In NIPS Bayesian deep learning workshop, 2017.

7. Soumya Ghosh, Jiayu Yao, and Finale Doshi-Velez. Structured variational learning of Bayesian neural networks with horseshoe priors. In International Conference on Machine Learning, pp. 1744–1753, 2018.

8. Alex Graves. Practical variational inference for neural networks. In Advances in Neural Information Processing Systems, pp. 2348–2356, 2011.

9. José Miguel Hernández-Lobato and Ryan Adams. Probabilistic backpropagation for scalable learning of Bayesian neural networks. In International Conference on Machine Learning, pp. 1861–1869, 2015.

10. Danijar Hafner, Dustin Tran, Alex Irpan, Timothy Lillicrap, and James Davidson. Reliable uncertainty estimates in deep neural networks using noise contrastive priors. arXiv preprint arXiv:1807.09289, 2018.

11. Ferenc Huszár. Variational inference using implicit distributions. arXiv preprint arXiv:1702.08235, 2017.

12. Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. arXiv preprint arXiv:1312.6114, 2013.

13. Diederik P Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. In Advances in Neural Information Processing Systems, pp. 2575–2583, 2015.

14. Jaehoon Lee, Jascha Sohl-dickstein, Jeffrey Pennington, Roman Novak, Sam Schoenholz, and Yasaman Bahri. Deep neural networks as Gaussian processes. In International Conference on Learning Representations, 2018.

15. Christos Louizos and Max Welling. Structured and efficient variational deep learning with matrix Gaussian posteriors. In International Conference on Machine Learning, pp. 1708–1716, 2016.

16. Chao Ma, Yingzhen Li, and José Miguel Hernández-Lobato. Variational implicit processes. arXiv preprint arXiv:1806.02390, 2018.

17. Alexander G de G Matthews, James Hensman, Richard Turner, and Zoubin Ghahramani. On sparse variational methods and the Kullback-Leibler divergence between stochastic processes. In Artificial Intelligence and Statistics, pp. 231–239, 2016.

18. Radford M Neal. Bayesian Learning for Neural Networks. PhD thesis, University of Toronto, 1995.

19. Geoffrey Roeder, Yuhuai Wu, and David K Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. In Advances in Neural Information Processing Systems, pp. 6925–6934, 2017.

20. Hugh Salimbeni and Marc Deisenroth. Doubly stochastic variational inference for deep Gaussian processes. In Advances in Neural Information Processing Systems, pp. 4588–4599, 2017.

21. Jiaxin Shi, Shengyang Sun, and Jun Zhu. Kernel implicit variational inference. In International Conference on Learning Representations, 2018.

22. Shengyang Sun, Changyou Chen, and Lawrence Carin. Learning structured weight uncertainty in Bayesian neural networks. In Artificial Intelligence and Statistics, pp. 1283–1292, 2017.

23. Guodong Zhang, Shengyang Sun, David Duvenaud, and Roger Grosse. Noisy natural gradient as variational inference. In International Conference on Machine Learning, pp. 5852–5861, 2018.

24. Hernández-Lobato, José Miguel, and Ryan Adams. "Probabilistic backpropagation for scalable learning of bayesian neural networks." International Conference on Machine Learning. 2015.

25. Li, Yingzhen, and Yarin Gal. "Dropout inference in Bayesian neural networks with alpha-divergences." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017.

26. Wang, Hao, and Dit-Yan Yeung. "Towards Bayesian deep learning: A framework and some existing methods." IEEE Transactions on Knowledge and Data Engineering 28.12 (2016): 3395-3408.

27. Gal, Yarin, Riashat Islam, and Zoubin Ghahramani. "Deep bayesian active learning with image data." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017.

28. Wang, Hao, and Dit-Yan Yeung. "Towards bayesian deep learning: A survey." arXiv preprint arXiv:1604.01662 (2016).

29. Khan, Mohammad Emtiyaz, et al. "Fast and scalable bayesian deep learning by weight-perturbation in adam." arXiv preprint arXiv:1806.04854 (2018).

30. Maddox, Wesley J., et al. "A simple baseline for bayesian uncertainty in deep learning." Advances in Neural Information Processing Systems. 2019.

31. Mukhoti, Jishnu, Pontus Stenetorp, and Yarin Gal. "On the importance of strong baselines in bayesian deep learning." arXiv preprint arXiv:1811.09385 (2018).