Robustness Guarantees for Bayesian Neural Networks*

Marta Kwiatkowska[0000-0001-9022-7599]

Department of Computer Science, University of Oxford, Oxford, UK
marta.kwiatkowska@cs.ox.ac.uk
http://www.cs.ox.ac.uk/people/marta.kwiatkowska/

Abstract. Bayesian neural networks (BNNs), a family of neural networks with a probability distribution placed on their weights, have the advantage of being able to reason about uncertainty in their predictions as well as data. Their deployment in safety-critical applications demands rigorous robustness guarantees. This paper summarises recent progress in developing algorithmic methods to ensure certifiable safety and robustness guarantees for BNNs, with the view to support design automation for systems incorporating BNN components.

Keywords: Bayesian neural networks · Probabilistic safety · Adversarial robustness · Certification.

1 Introduction

Neural networks (NNs) are being introduced across many domains, including robotics, autonomous vehicles, security and healthcare, but their deployment in safety-critical scenarios demands rigorous robustness guarantees in the presence of uncertainty, which are lacking for NNs. *Bayesian neural networks* (BNNs) [6] are a family of neural networks that place distributions over their weights, instead of viewing them as fixed values, and can thus account for uncertainty in data and predictions. Starting with a prior distribution and a given likelihood, the application of Bayes' theorem results in posterior probability distribution over the BNN weights conditional on the observed data. This induces posterior predictive distribution on the BNN outputs, with the final BNN prediction selected from this distribution according to Bayesian decision theory. BNNs therefore combine the high capacity of NNs while enabling (Bayesian) probabilistic reasoning, since they can be viewed as stochastic processes.

This invited paper describes recent progress in developing methods to provide robustness guarantees for Bayesian neural networks. These include certifiable adversarial training, statistical evaluation of probabilistic safety, and certified

* This project was funded by the ERC under the European Union's Horizon 2020 research and innovation programme (FUN2MODEL, grant agreement No. 834115, and ELSA: European Lighthouse on Secure and Safe AI project, grant agreement No. 101070617 under UK guarantee).

lower bounding of safety probability. The discussed methods draw on probabilistic reachability analysis, sampling, statistical model checking and convex relaxation, and constitute part of an effort to develop probabilistic verification and synthesis methodologies for systems incorporating BNN components.

2 Background on Bayesian Neural Networks

A feed-forward neural network (NN) is a function $f^w : \mathbb{R}^m \to \mathbb{R}^n$, parametrised by a vector $w \in \mathbb{R}^{n_w}$ that includes all the weights of the network (for simplicity assume no bias). We work in a supervised learning scenario, where we are given a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n_\mathcal{D}}$ of pairs of inputs and ground truth labels, with $x_i \in \mathbb{R}^m$, and where each target output $y \in \mathbb{R}^n$ is either a one-hot class vector for classification or a real-valued vector for regression.

A Bayesian neural network (BNN) [6] is an NN with a distribution placed over the network parameters w, and can thus be viewed as a stochastic process $f^{\mathbf{w}}$ (vector of random variables $\mathbf{w}$ associated to the weights) indexed by the input space. Note that, for a weight vector w sampled from the distribution of $\mathbf{w}$, the BNN induces a (deterministic) NN f^w with weights fixed to w. We employ Bayesian learning to infer the weight parameters, starting with a prior distribution $p_{\mathbf{w}}(w)$ over $\mathbf{w}$ and likelihood $p(\mathcal{D}|w) = \prod_{i=1}^{n_\mathcal{D}} p(y_i|x_i, w)$, to compute the posterior distribution $p_{\mathbf{w}}(w|\mathcal{D})$ of parameters conditioned on data by applying the Bayes formula, i.e., $p_{\mathbf{w}}(w|\mathcal{D}) \propto p(\mathcal{D}|w)p_{\mathbf{w}}(w)$. This induces the distribution over outputs called the posterior predictive distribution defined for an unseen point x^* by $p(y^*|x^*, \mathcal{D}) = \int p(y^*|x^*, w)p_{\mathbf{w}}(w|\mathcal{D})dw$. The final prediction is obtained based on Bayesian decision theory and is the value $\hat{y}$ that minimizes the Bayesian risk of an incorrect prediction according to the posterior predictive distribution and a loss function $\mathcal{L}$, computed as $\hat{y} = \arg\min_y \int_{\mathbb{R}^n} \mathcal{L}(y, y^*)p(y^*|x^*, \mathcal{D})dy^*$. For classification decisions, we typically work with 0-1 loss and the optimal decision is then the class that maximises the predictive distribution, whereas for regression ℓ_2 loss is used and the optimal decision the expected value of the BNN output over the posterior distribution.

Unfortunately, the computation of the posterior distribution $p_{\mathbf{w}}(w|\mathcal{D})$ over weights cannot be computed analytically and is generally intractable [6]. Instead, approximate inference methods have been developed for BNNs, of which Hamiltonian Monte Carlo (HMC) [6] and Variational Inference (VI) [1] are commonly used. HMC considers Hamiltionian dynamics to speed up the exploration, working with a Markov chain whose invariant distribution is $p_{\mathbf{w}}(w|\mathcal{D})$, and is asymptotically correct [6]. The result of HMC is a set of samples that approximates $p_{\mathbf{w}}(w|\mathcal{D})$. VI proceeds by finding a Gaussian approximating distribution over the weight space $q(w) \sim p_{\mathbf{w}}(w|\mathcal{D})$, where $q(w)$ depends on some hyperparameters that are then iteratively optimized by minimizing a divergence measure between $q(w)$ and $p_{\mathbf{w}}(w|\mathcal{D})$, thus trading off approximation accuracy against scalability. Samples can then be efficiently extracted from $q(w)$.

3 Certifiable Adversarial Robustness

Though the ability of Bayesian neural networks to capture uncertainty is appealing for safety-critical applications, they are susceptible to adversarial attacks. In [7], a principled Bayesian approach was proposed for incorporating adversarial robustness in the posterior inference procedure of BNNs. To this end, the robustness requirement is formulated as the worst-case prediction over an adversarial input ball of radius $\epsilon \geq 0$ induced by a user-defined probability density function p_ϵ, and the standard cross-entropy likelihood model was extended by marginalising the network output over p_ϵ called *robust likelihood*. Further, for any $\epsilon > 0$, certified lower bounds to the robust likelihood can be computed by employing interval bound propagation techniques. This novel adversarial training procedure adapts naturally to the main approximate inference techniques employed for training of BNNs, including HMC and VI. An experimental evaluation in [7] demonstrated that the robust likelihood can double the maximal safe radius for the standard model and results in better calibrated uncertainty when predicting out-of-distribution samples.

4 Probabilistic Safety Evaluation

Safe decision making is important in autonomous scenarios, where it can benefit from uncertainty estimates being propagated through the decision pipeline. In [5], a setting involving an end-to-end BNN autonomous driving controller based on NVIDIA's PilotNet was considered, which can be viewed as a discrete-time stochastic process, and a framework was proposed for evaluating safety of the controller's decisions. Two properties were considered, probabilistic safety, i.e., the probability that the controller will maintain the safety of the car for a given time horizon, and real-time decision confidence, i.e., the probability that the BNN is certain of a given decision. We remark that probabilistic safety represents a probabilistic variant of the notion of safety [3] commonly used to certify deterministic NNs. A statistical model checking framework based on [2] is employed to evaluate robustness of these properties to changes in weather, location and observation noise with a priori confidence interval guarantees (using Chernoff bounds) in a simulated scenario. Here, we exploit the fact that sampling BNN weights results in a deterministic NN, which can be checked using conventional methods for NNs, and the proportion of sampled NNs that are safe yields a probability estimate of BNN safety. [5] also shows how to quantify the uncertainty of the controller's decisions and utilise uncertainty thresholds in order to guarantee the safety of the self-driving car with high probability. Separately, [4] study infinite-time horizon robustness properties for BNNs.

5 Certified Bounds on Safety Probability

Probabilistic safety evaluation based on [2] can only provide guarantees in the form of confidence intervals, which may not be sufficient for highly safety-critical systems. [8] considered certification of (lower bounds on) the safety probability. The method is based on observing that probabilistic safety translates into computing the probability that adversarial perturbations of an input cause small variations in the BNN output. For BNNs, this involves working with posterior probability and showing that the computation of probabilistic safety for BNNs is equivalent to computing the measure, w.r.t. BNN posterior, of the set of weights for which the resulting deterministic NN is safe, i.e., robust to adversarial perturbations. Once the set of such weights is computed, relaxation techniques from non-linear optimisation (interval bound propagation and linear bound propagation) are employed to check whether all the networks instantiated by these weights are safe. This yields lower bounds on the probability for the case of BNNs trained with VI, but the method extends to other approximate Bayesian inference techniques. Experimental evaluation on the VCAS collision-avoidance case study demonstrates the practicality of the method. In follow-on work, [9] consider also synthesis of certified policies for BNNs.

6 Conclusion and Further Work

We have provided an overview of algorithmic techniques developed to ensure certified guarantees of safety and adversarial robustness for BNNs. Certification of BNNs is more involved than for NNs, because of the need to consider weight intervals instead of single values, and presents significant computational challenges that have so far been tackled using a combination of numerical, statistical and symbolic techniques. Despite encouraging progress, much remains to be done, including upper bounding of safety, certified bounds on decision probability, temporal logic specifications, strategy synthesis and explanations for BNNs.

References

1. Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: Weight uncertainty in neural networks. In: ICML (2015)
2. Cardelli, L., Kwiatkowska, M., Laurenti, L., Paoletti, N., Patane, A., Wicker, M.: Statistical guarantees for the robustness of Bayesian neural networks. In: IJCAI (2019)
3. Huang, X., Kwiatkowska, M., Wang, S., Wu, M.: Safety verification of deep neural networks. In: CAV (2017)
4. Lechner, M., Zikelic, D., Chatterjee, K., Henzinger, T.A.: Infinite time horizon safety of Bayesian neural networks. In: NeurIPS (2021)
5. Michelmore, R., Wicker, M., Laurenti, L., Cardelli, L., Gal, Y., Kwiatkowska, M.: Uncertainty quantification with statistical guarantees in end-to-end autonomous driving control. In: ICRA (2020)
6. Neal, R.M.: Bayesian learning for neural networks. Springer Science & Business Media (2012)
7. Wicker, M., Laurenti, L., Patane, A., Chen, Z., Zhang, Z., Kwiatkowska, M.: Bayesian inference with certifiable adversarial robustness. In: AISTATS (2021)

8. Wicker, M., Laurenti, L., Patane, A., Kwiatkowska, M.: Probabilistic safety for Bayesian neural networks. In: UAI (2020)
9. Wicker, M., Laurenti, L., Patane, A., Paoletti, N., Abate, A., Kwiatkowska, M.: Certification of iterative predictions in Bayesian neural networks. In: UAI (2021)