5th Symposium on Advances in Approximate Bayesian Inference, 2023 1–28

Attacking Bayes: Are Bayesian Neural Networks Inherently Robust?

Anonymous Authors
Anonymous Institution

Abstract

This work aims to deconstruct the belief that Bayesian neural networks (BNNs) are inherently robust to adversarial perturbations. To this end, we demonstrate that it is possible to successfully break Bayesian neural network prediction pipelines using even relatively unsophisticated attacks like PGD-variants for three tasks: (1) label prediction under the posterior predictive mean, (2) adversarial example detection with Bayesian predictive uncertainty, and (3) semantic shift detection. We identify various conceptual and experimental errors in previous works that claim superior adversarial robustness of BNNs, and issue a set of recommendations to avoid pitfalls in the robustness analysis of BNNs. We believe that this work conclusively demonstrates that Bayesian neural networks and uncertainty-aware Bayesian prediction pipelines are *not* a priori robust against adversarial attacks and hope to shift the focus of the community to creating successful defenses for Bayesian prediction pipelines.

1. Introduction

The deployment of machine learning systems in critical applications, such as medical tasks and transportation systems, requires a high level of reliability and robustness. However, modern machine learning systems have been shown to lack robustness in the presence of adversarially chosen inputs that are perceptually indistinguishable from data the model can successfully handle.

This vulnerability was first observed in computer vision by Szegedy et al. (2014) and has since been shown to be consistent across various benchmarks and environments, including standard academic benchmarks (Goodfellow et al., 2015; Papernot et al., 2017; Carlini and Wagner, 2017b), as well as in less controlled environments (Kurakin et al., 2017; Alzantot et al., 2018). While various defense methods have been proposed to improve adversarial robustness, many have been broken by stronger attacks and comprehensive evaluations. Subsequent transformative works (Carlini et al., 2019a; Croce and Hein, 2020; Croce et al., 2021) have established baseline evaluation protocols to help researchers avoid common pitfalls and ensure confidence in proposed defenses, benefiting the entire community.

In this paper, we challenge the belief that Bayesian neural networks (BNNs) are inherently robust to adversarial attacks. In contrast to deterministic methods, Bayesian machine learning (MacKay, 1992; Neal, 1996; Murphy, 2013) offers a principled way to quantify models' predictive uncertainty. In BNNs, this is achieved by viewing the network parameters as random variables, inferring a posterior distribution over the network parameters using Bayesian inference, and using samples from this posterior distribution to obtain a posterior predictive distribution. Unlike deterministic neural networks, a BNN's posterior predictive distribution expresses uncertainty both due to noise in the data and due to the BNN's own

© A. Authors.

limitations, for example, because of data scarcity or semantic shift in the data. Naturally, one could hope that a BNN's posterior predictive uncertainty may make the predictions more robust to adversarial attacks, and a number of published works claim to provide empirical and theoretical evidence that BNNs are able to detect adversarial examples (Rawat et al., 2017; Smith and Gal, 2018) and able to defend against gradient-based attacks on predictive accuracy (Carbone et al., 2020; Bortolussi et al., 2022; Zhang et al., 2021). At the same time, other works show that Gaussian Processes (GPs), a non-parametric alternative to BNNs, are not robust to adversarial attacks on simple prediction tasks (Grosse et al., 2018) and that BNNs with simple architectures are not robust to adversarial attacks in small-data settings (Blaas, 2021), casting doubt on the robustness claims made in the literature.

In this work, we aim to settle this ambiguous state of affairs on the robustness of BNNs and their prediction pipelines. After reviewing the body of prior evidence (Section 2), we conduct a thorough evaluation of past work and then tackle well-established and state-of-the-art BNN inference methods in a principled way, by focusing on three key tasks: 1) classification under the posterior predictive mean; 2) adversarial example (AE) detection; and 3) semantic shift detection. To summarize, our key contributions are as follows:

- We re-examine prior evidence on robustness of BNNs for 1) AE detection (Smith and Gal, 2018) and 2) prediction (Carbone et al., 2020; Bortolussi et al., 2022; Zhang et al., 2021) and conclude that none of them convincingly demonstrates robustness against adversarial attacks. We identify conceptual and practical errors in their evaluations and provide a series of guiding principles to evaluate robustness of BNNs with suitable attack methods. We hope that this effort not only corrects previous misconceptions, but also helps evaluate BNN robustness with more confidence (Section 2).
- We conduct thorough evaluations of well-established state-of-the-art predictive uncertainty estimates (HMC, Neal (2010); PSVI Blundell et al. (2015); MCD, Gal and Ghahramani (2016); FSVI, Rudner et al. (2022)) on benchmarking tasks such as MNIST, FashionMNIST, and CIFAR-10, breaking all of them. We demonstrate that 1) classification under the posterior lacks predictive accuracy under adversarial attacks, 2) AE detection fails even under attacks targeting accuracy only, and 3) in semantic shift detection (on MNIST vs. FashionMNIST and CIFAR-10 vs. SVHN) adversarial attacks fool BNNs to mainly reject in-distribution samples (our work is the first to break semantic-shift detection by BNNs) (Section 3).

We conclude: *Bayesian neural networks and the Bayesian selective prediction pipeline are* not *a priori robust against adversarial attacks. Moreover, relatively unsophisticated default attacks like PGD-variants are sufficient to break existing Bayesian pipelines.*

2. Evaluating Claims about BNN Robustness

Here, we examine (and refute) all papers (to the best of our knowledge) that make adversarial robustness claims about BNNs that have publicly accessible code and have not been previously refuted (Smith and Gal, 2018; Bortolussi et al., 2022; Carbone et al., 2020; Zhang et al., 2021).Each of them provides a different failure mode that will help illustrate our recommendations for evaluating robustness of BNNs at the end of this section.

AE detection with epistemic uncertainty. Smith and Gal (2018) examine adversarial detection with MCD using a ResNet-50 on the ASSIRA Cats & Dogs dataset consisting of clean test images, adversarial samples on that test set, and noisy test images with the same perturbation levels as the adversarial ones, where they use PGD 10 in their attacks. They present empirical findings claiming that *epistemic* uncertainty in particular may help detect adversarial examples.

However, after investigating their code, we find several problems: leakage of batch statistics at test time, faulty cropping of noisy images and the "double-softmax" problem (more details in Appendix 3). This leads the BNN to accept the noisy images and reject the clean and adversarial ones, resulting in misleading, overly optimistic ROC curves. After correcting these errors, no evidence of successful AE detection from selective prediction remains. Figure 1 shows the updated selective accuracy using both total and epistemic uncertainty thresholds. A successful AE detector would have a monotonically increasing accuracy curve, whereas we see flat or decreasing curves meaning no better than random detection and no advantage of epistemic over total uncertainty. Lastly, we implement stronger attacks (PGD 40 and TransferPGD+, see Appendix 3) to demonstrate the complete failure of this AE detection method.

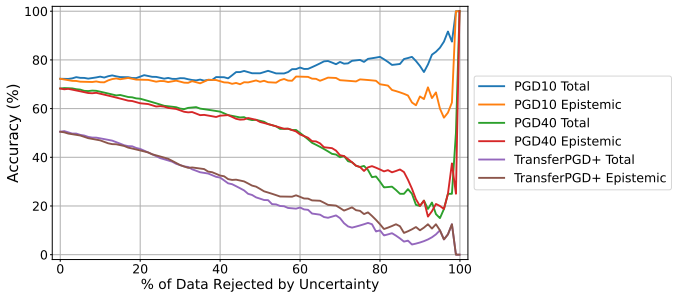

Figure 1: Selective Accuracy for the AE detection in Smith and Gal (2018). Total and Epistemic refer to the thresholding uncertainty. The downward sloping curves as rejection rate increases indicate that the model rejects more clean than adversarial samples. Even for the weakest attack on accuracy alone, the essentially flat curve demonstrates that detection is no better than random. There is no advantage in using epistemic uncertainty rather than total uncertainty.

Robustness of BNN accuracy. Carbone et al. (2020) and Bortolussi et al. (2022) present empirical results claiming robustness of BNN accuracy due to vanishing gradients of the input with respect to the posterior. They implement FGSM and PGD for BNNs with HMC and PSVI on MNIST and FashionMNIST to show robust accuracy. However, when examining the publicly accessible code, we found that instead of using Equation (1.3) to calculate gradients, they compute expectations of gradients. Combined with large logit values before the softmax layer that lead to numerical overflow and result in zero gradients for a large fraction of samples, this leads to inadvertent gradient masking. In addition, we also found the double-softmax problem mentioned above. After correcting and rescaling logits to avoid numerical issues (see Appendix 3), their models are entirely broken by PGD 40, see Table 1.

Regularization for robust accuracy of BNNs. Zhang et al. (2021) propose to defend against adversarial attacks by adding a regularization term and present empirical results for enhanced accuracy robustness for MNIST and CIFAR (most for non-standard settings of attack parameters like smaller radius). We again found the same "double-softmax" problem plaguing other implementations and show that after fixing it the claims do not hold anymore (see Appendix 3).

Table 1: Robust accuracy on MNIST for BNNs trained with HMC and PSVI. We show the results reported in Carbone et al. (2020); Bortolussi et al. (2022) and our reevaluation.

Methods		Clean	FGSM	PGD 40
HMC	*Reported*	-	96.0	97.0
	Re-evaluated	$95.89_{\pm 0.23}$	$11.62_{\pm 2.49}$	$2.19_{\pm 0.12}$
PSVI	*Reported*	-	93.6	93.8
	Re-evaluated	$97.35_{\pm 0.18}$	$51.45_{\pm 3.70}$	$0.07_{\pm 0.05}$

3. Empirical Evaluation of Adversarial Robustness of BNNs

Here we present our findings on the lack of adversarial robustness of BNN pipelines for the three tasks: 1) classification with posterior prediction mean (Section 3.1), 2) AE detection (Section 3.2), and 3) OOD detection (Section 3.3). We evaluate four Bayesian Inference methods, HMC, MCD, PSVI, and FSVI for three datasets: MNIST, FashionMNIST, and CIFAR-10. We implement two architectures, a four-layer CNN and ResNet-18. All hyperparameters can be found in Appendix 7.

Threat model and principled generation of adversarial perturbations with FGSM, PGD and PGD+. We consider a *full white-box* attack: The adversary has knowledge of the entire model and its parameters, and can obtain samples from its posterior, which is the white-box model considered in most prior work (Carbone et al., 2020; Bortolussi et al., 2022; Zhang et al., 2021). For further details about the attack methods, FGSM, PGD, and PGD+, see Appendix 5.

Metrics: We report accuracy from posterior mean prediction with 100 samples. Our notion of *robustness* for BNNs is the natural one that aligns with deployment of BNNs: Robust accuracy is given by the fraction of correctly classified adversarial samples when predicting the class with the largest posterior mean prediction probability. As summary statistics for the selective prediction curves we use average selective accuracy (ASA), that is, the area under the selective accuracy curve computed with rejection rates from 0% to 99% in integer increments; and average negative log-likelihood (ANLL), for which we average NLL of all non-rejected samples for the same rejection grid as for ASA (see Appendix 6).

3.1. Robust Accuracy of Bayesian Neural Networks

Table 2 shows the predictive accuracies from our evaluation. It demonstrates a significant deterioration in predictive accuracy when evaluated on adversarial examples even for the weakest attacks (FGSM) with a complete breakdown for PGD for all methods and datasets considered. Note that for deterministic neural networks, robust accuracy under adversarial attacks approaches 0% while for our attacks on BNNs it is in the low single digits (still below the 10% accuracy for random guessing). Since the goal of this work is to evaluate claims of *significant* adversarial robustness of BNNs, we have not optimized our attacks to drive accuracy to approach zero but believe this to be possible.

Table 2: Robustness of BNNs to adversarial attacks. The table shows robust accuracy (%).

Methods		MNIST ($\epsilon = 0.3$)			FashionMNIST ($\epsilon = 0.1$)			CIFAR-10 ($\epsilon = 8/255$)		
		Clean	FGSM	PGD	Clean	FGSM	PGD	Clean	FGSM	PGD
HMC	CNN	99.26±0.00	21.44±0.58	0.57±0.05	92.33±0.04	32.79±0.829	6.73±0.30	–	–	–
MCD	CNN	99.39±0.04	10.19±1.51	0.52±0.03	93.04±0.10	18.45±2.78	5.37±0.14	–	–	–
	ResNet-18	99.39±0.05	10.19±1.51	5.20±0.03	94.27±0.05	7.58±0.29	4.40±0.11	94.12±0.07	26.45±0.15	4.23±0.17
PSVI	CNN	99.21±0.02	2.60±0.02	0.64±0.02	92.58±0.01	13.95±0.88	5.50±0.14	–	–	–
	ResNet-18	99.59±0.02	2.07±0.25	0.36±0.02	94.22±0.08	16.86±5.64	4.32±0.10	94.75±0.26	37.17±1.11	5.25±2.27
FSVI	CNN	99.27±0.01	41.94±1.82	0.60±0.06	92.58±0.25	23.93±1.87	5.45±0.28	–	–	–
	ResNet-18	99.58±0.02	6.45±3.33	0.39±0.02	93.62±0.33	28.23±2.63	4.60 ±0.02	93.48±0.18	43.85±0.94	5.18±0.27

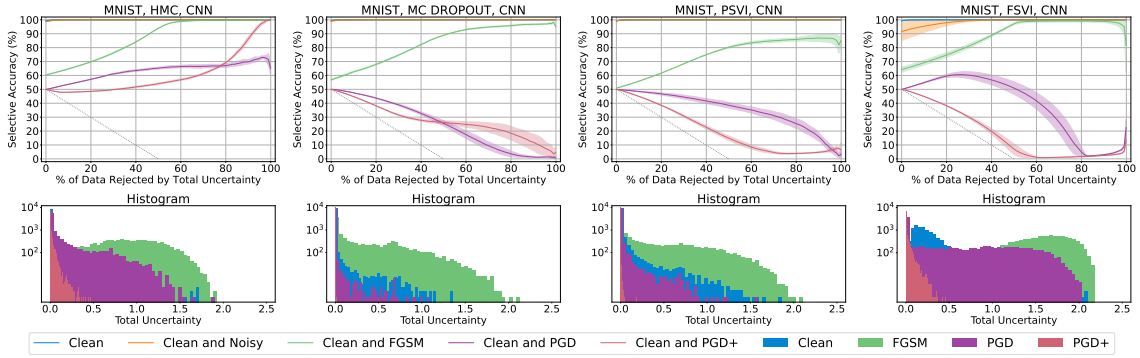

Figure 2: AE detection statistics for all four methods on MNIST with a four-layer CNN.

3.2. Detecting Adversarial Examples

AE detection setting: We evaluate all AE detectors on test data consisting of 50% clean samples and 50% adversarially perturbed samples, using total uncertainty for the rejection as described in Sections 1.2 and 1.4. In the idealized case of perfect accuracy on clean data and 0% accuracy on adversarial samples, a perfect AE detector would start with 50% accuracy at 0% rejection rate, increasing linearly to 100% accuracy at 50% rejection rate, for a maximum ASA of 87.5%. A completely defunct AE detector, on the other hand, would start at 50% and reject clean samples first, reaching 0% accuracy at 50% rejection rate for a minimum ASA of 12.5%. A random detector would yield a horizontal curve with ASA 50%.

Table 3: The table shows the average accuracy for selective prediction (ASA).

		CNN					ResNet-18				
		Clean	Noisy	FGSM	PGD	PGD+	Clean	Noisy	FGSM	PGD	PGD+
MNIST	HMC	99.98±0.00	99.95±0.00	86.67±0.13	63.16±0.73	60.16±0.33	-	-	-	-	-
	MCD	99.99±0.00	99.97±0.00	83.33±1.48	24.58±1.21	27.33±1.78	99.99±0.00	80.67±2.35	83.10±0.38	23.20±0.63	16.25±0.12
	PSVI	99.98±0.00	99.98±0.00	74.98±0.83	35.28±1.88	20.28±0.70	99.99±0.00	87.49±0.79	83.00±0.83	19.30±0.46	15.95±0.07
	FSVI	99.98±0.00	98.84±1.09	88.53±1.31	38.79±2.72	17.52±0.68	99.37±0.88	88.82±9.03	81.52±5.85	26.86±3.81	20.14±5.48
FMNIST	HMC	98.99±0.00	98.76±0.01	76.22±0.41	47.73±0.67	41.30±0.60	-	-	-	-	-
	MCD	99.18±0.01	99.07±0.01	75.31±0.35	31.92±0.77	28.89±0.45	99.23±0.02	98.68±0.03	69.73±1.99	20.18±0.46	19.89±0.19
	PSVI	98.98±0.01	98.86±0.03	62.03±2.02	30.68±0.58	24.92±0.49	98.72±0.16	97.91±0.22	79.97±3.92	19.73±0.51	19.56±0.44
	FSVI	98.58±0.04	97.93±0.28	69.59±1.48	30.94±1.91	24.17±0.82	98.87±0.12	98.06±0.17	78.49±1.79	21.45 ±0.08	20.49±0.43
CIFAR -10	MCD	-	-	-	-	-	99.37±0.01	99.35±0.01	76.97±0.21	22.85±0.49	19.73±0.24
	PSVI	-	-	-	-	-	99.40±0.04	99.36±0.04	82.38±0.50	19.78±0.97	19.89±1.51
	FSVI	-	-	-	-	-	99.06±0.08	99.04±0.07	85.16±0.35	20.82±0.38	20.09±0.33

Authors

To benchmark, we also show ASA for 100% clean test data ("Clean") and for a 50-50 mix of clean and noisy data where we add a pixel-wise Gaussian perturbation with the same standard deviation as the radius in our adversarial perturbations ("Noisy").

Results: Table 3 lists our results for ASA, with all methods failing under attack, coming quite close to the idealized minimum ASA of 12.5%. Figure 2 (MNIST with CNN) (and Figures 7, 4, 5, 6 in Appendix 6 for the other datasets and architectures) illustrate the selective accuracy curve for the benchmarks and our three attacks, FGSM, PGD and PGD+, and show a histogram of uncertainties for adversarial samples. Table 6 in Appendix 6 further lists ANLL. Our results show that our iterative attacks, PGD and PGD+, essentially completely fool AE detection. Particularly interesting is the fact that PGD samples, optimized against predictive accuracy only (but not against uncertainty), already manage to fool the uncertainty thresholding, resulting in decreasing accuracy curves! The principled two-stage PGD+ attack that also targets uncertainty directly further lowers detection performance and decreases uncertainty further for all adversarial samples. We note that the weaker FGSM attack is not as successful: As the histograms show, uncertainty does not fall below that of clean samples and the selective accuracy curve increases with an increased rejection rate.

3.3. Robustness of Semantic Shift Detection

We also evaluated the robustness of BNNs' ability to perform semantic shift detection. For details, see Appendix 5.1

4. Discussion & Conclusion

Our empirical analysis has refuted prior accessible evidence that BNNs enjoy some natural inherent robustness to adversarial attacks, or that they can be successfully deployed a priori for AE detection. We benchmarked a set of contemporary BNN inference methods to substantiate this claim. In addition, we are the first to demonstrate that OOD detection of semantic shifts with BNNs is equally vulnerable to relatively simple attacks: Slightly perturbing even just the semantically shifted samples leads the model to reject in-distribution samples instead.

In our empirical evaluation, we found that even BNNs trained with HMC, the gold standard for inference in BNNs, do not withstand adversarial attacks and exhibit a significant deterioration in robust accuracy, average selective accuracy, and semantic shift detection when either their predictions or their predictive uncertainty are attacked. Unfortunately, our analysis of HMC is limited to moderately-sized CNNs since training larger BNNs with HMC is computationally infeasible without super computer-grade hardware—leaving the adversarial robustness of larger BNNs trained with HMC an open question.

Additionally, with recent progress on creating priors from pretrained publicly-available models (Shwartz-Ziv et al., 2022; Tran et al., 2022) and using transfer learning to apply BNNs to challenging downstream tasks, an interesting area for future research would be to explore the vulnerability of BNNs against transfer attacks from models that do not have access to the posterior.

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani B. Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2890–2896. Association for Computational Linguistics, 2018.

Anish Athalye, Nicholas Carlini, and David A. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 274–283. PMLR, 2018a.

Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 284–293. PMLR, 2018b.

Neil Band, Tim G. J. Rudner, Qixuan Feng, Angelos Filos, Zachary Nado, Michael W. Dusenberry, Ghassen Jerfel, Dustin Tran, and Yarin Gal. Benchmarking Bayesian Deep Learning on Diabetic Retinopathy Detection Tasks. 2021.

Artur Bekasov and Iain Murray. Bayesian adversarial spheres: Bayesian inference and adversarial examples in a noiseless setting. *arXiv preprint arXiv:1811.12335*, 2018.

Arno Blaas and Stephen J Roberts. The effect of prior lipschitz continuity on the adversarial robustness of bayesian neural networks. *arXiv preprint arXiv:2101.02689*, 2021.

Arno C Blaas. *On the adversarial robustness of Bayesian machine learning models*. PhD thesis, University of Oxford, 2021.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. volume 37 of *Proceedings of Machine Learning Research*, pages 1613–1622, Lille, France, 07–09 Jul 2015. PMLR.

Luca Bortolussi, Ginevra Carbone, Luca Laurenti, Andrea Patane, Guido Sanguinetti, and Matthew Wicker. On the robustness of bayesian neural networks to adversarial attacks. *arXiv preprint arXiv:2207.06154*, 2022.

Ginevra Carbone, Matthew Wicker, Luca Laurenti, Andrea Patane', Luca Bortolussi, and Guido Sanguinetti. Robustness of bayesian neural networks to gradient-based attacks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 15602–15613. Curran Associates, Inc., 2020.

Luca Cardelli, Marta Kwiatkowska, Luca Laurenti, and Andrea Patane. Robustness guarantees for bayesian inference with gaussian processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7759–7768, 2019.

Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, AISec '17, page 3–14, New York, NY, USA, 2017a. Association for Computing Machinery.

Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 39–57. IEEE Computer Society, 2017b.

Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019a.

Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian J. Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *CoRR*, abs/1902.06705, 2019b.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In Bhavani M. Thuraisingham, Battista Biggio, David Mandell Freeman, Brad Miller, and Arunesh Sinha, editors, *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, AISec@CCS 2017, Dallas, TX, USA, November 3, 2017*, pages 15–26. ACM, 2017.

Adam D Cobb and Brian Jalaian. Scaling hamiltonian monte carlo inference for bayesian neural networks with symmetric splitting. *Uncertainty in Artificial Intelligence*, 2021.

Thomas M. Cover and Joy A. Thomas. *Elements of information theory.* Wiley, New York, 1991.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Giacomo De Palma, Bobak Kiani, and Seth Lloyd. Adversarial robustness guarantees for random deep neural networks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 2522–2534. PMLR, 18–24 Jul 2021.

Zhijie Deng, Xiao Yang, Shizhen Xu, Hang Su, and Jun Zhu. Libre: A practical bayesian approach to adversarial detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 972–982, 2021.

Stefan Depeweg, Jose-Miguel Hernandez-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive

learning. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1184–1193, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/depeweg18a.html.

Bao Gia Doan, Ehsan M. Abbasnejad, Javen Qinfeng Shi, and Damith Ranasinghe C. Bayesian learning with information gain provably bounds risk for a robust adversarial defense. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5309–5323. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/doan22a.html.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(53):1605–1641, 2010. URL http://jmlr.org/papers/v11/el-yaniv10a.html.

Reuben Feinman, Ryan R. Curtin, Saurabh Shintre, and Andrew B. Gardner. Detecting adversarial samples from artifacts, 2017. URL https://arxiv.org/abs/1703.00410.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML 2016, pages 1050–1059, 2016.

Ido Galil and Ran El-Yaniv. Disrupting deep uncertainty estimation without harming accuracy. *Advances in Neural Information Processing Systems*, 34:21285–21296, 2021.

Yue Gao, Ilia Shumailov, Kassem Fawaz, and Nicolas Papernot. On the limitations of stochastic pre-processing defenses. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Alex Graves. Practical variational inference for neural networks. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, page 2348–2356, Red Hook, NY, USA, 2011. Curran Associates Inc. ISBN 9781618395993.

Kathrin Grosse, David Pfaff, Michael Thomas Smith, and Michael Backes. The limitations of model uncertainty in adversarial settings. *arXiv preprint arXiv:1812.02606*, 2018.

Kathrin Grosse, Michael T Smith, and Michael Backes. Killing four birds with one gaussian process: the relation between different test-time attacks. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 4696–4703. IEEE, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition,*

CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, pages 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90. URL https://doi.org/10.1109/CVPR.2016.90.

J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, UAI'13, pages 282–290, Arlington, Virginia, United States, 2013. AUAI Press. URL http://dl.acm.org/citation.cfm?id=3023638.3023667.

James Hensman, Alexander G. de G. Matthews, and Zoubin Ghahramani. Scalable variational Gaussian process classification. In *AISTATS*, 2014.

Matthew D. Hoffman, David M. Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14(1):1303–1347, May 2013. ISSN 1532-4435.

Pavel Izmailov, Sharad Vikram, Matthew D Hoffman, and Andrew Gordon Gordon Wilson. What are bayesian neural network posteriors really like? In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4629–4640. PMLR, 18–24 Jul 2021.

Anna-Kathrin Kopetzki, Bertrand Charpentier, Daniel Zügner, Sandhya Giri, and Stephan Günnemann. Evaluating robustness of predictive uncertainty estimation: Are dirichlet-based models reliable? In *International Conference on Machine Learning*, pages 5707–5718. PMLR, 2021.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*, 2017.

Yao Li, Tongyi Tang, Cho-Jui Hsieh, and Thomas Lee. Detecting adversarial examples with bayesian neural network. *arXiv preprint arXiv:2105.08620*, 2021.

Xuanqing Liu, Yao Li, Chongruo Wu, and Cho-Jui Hsieh. Adv-BNN: Improved adversarial defense through robust bayesian neural network. In *International Conference on Learning Representations*, 2018.

David J. C. MacKay. A practical Bayesian framework for backpropagation networks. *Neural Comput.*, 4(3):448–472, May 1992. ISSN 0899-7667. doi: 10.1162/neco.1992.4.3.448.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*, 2018.

Kevin P. Murphy. *Machine learning : a probabilistic perspective*. MIT Press, Cambridge, Mass. [u.a.], 2013. ISBN 9780262018029 0262018020.

Radford M Neal. Bayesian Learning for Neural Networks. 1996.

Radford M. Neal. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 54:113–162, 2010.

Yutian Pang, Sheng Cheng, Jueming Hu, and Yongming Liu. Evaluating the robustness of Bayesian neural networks against different types of attacks. *arXiv preprint arXiv:2106.09223*, 2021.

Nicolas Papernot, Patrick D. McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*, pages 582–597. IEEE Computer Society, 2016.

Nicolas Papernot, Patrick D. McDaniel, Ian J. Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In Ramesh Karri, Ozgur Sinanoglu, Ahmad-Reza Sadeghi, and Xun Yi, editors, *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, AsiaCCS 2017, Abu Dhabi, United Arab Emirates, April 2-6, 2017*, pages 506–519. ACM, 2017.

Andrea Patane, Arno Blaas, Luca Laurenti, Luca Cardelli, Stephen Roberts, and Marta Kwiatkowska. Adversarial robustness guarantees for gaussian processes. *Journal of Machine Learning Research*, 23:1–55, 2022.

Stephan Rabanser, Anvith Thudi, Kimia Hamidieh, Adam Dziedzic, and Nicolas Papernot. Selective classification via neural network training dynamics, 2022.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Massachusetts Institute of Technology, Cambridge, MA, USA, 2006. URL www.{G}aussianProcess.org/gpml.

Ambrish Rawat, Martin Wistuba, and Maria-Irina Nicolae. Adversarial phenomenon in the eyes of bayesian deep learning, 2017.

Tim G.J. Rudner, Zonghao Chen, Yee Whye Teh, and Yarin Gal. Tractable Function-space Variational Inference in Bayesian Neural Networks. In *Advances in Neural Information Processing Systems*, 2022.

Vikash Sehwag, Arjun Nitin Bhagoji, Liwei Song, Chawin Sitawarin, Daniel Cullina, Mung Chiang, and Prateek Mittal. Better the devil you know: An analysis of evasion attacks using out-of-distribution adversarial examples. *arXiv preprint arXiv:1905.01726*, 2019.

Claude E. Shannon and Warren Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, Urbana and Chicago, 1949.

Ravid Shwartz-Ziv, Micah Goldblum, Hossein Souri, Sanyam Kapoor, Chen Zhu, Yann LeCun, and Andrew Gordon Wilson. Pre-train your loss: Easy bayesian transfer learning with informative priors. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=YCniF6_3Jb.

Lewis Smith and Yarin Gal. Understanding measures of uncertainty for adversarial example detection. In Amir Globerson and Ricardo Silva, editors, *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*, pages 560–569. AUAI Press, 2018. URL http://auai.org/uai2018/proceedings/papers/207.pdf.

Edward Snelson and Zoubin Ghahramani. Sparse Gaussian processes using pseudo-inputs. In Y. Weiss, B. Schölkopf, and J. C. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 1257–1264. MIT Press, 2006.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks, 2014.

Florian Tramer. Detecting adversarial examples is (Nearly) as hard as classifying them. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 21692–21702. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/tramer22a.html.

Florian Tramèr, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Dustin Tran, Jeremiah Liu, Michael W. Dusenberry, Du Phan, Mark Collier, Jie Ren, Kehang Han, Zi Wang, Zelda Mariet, Huiyi Hu, Neil Band, Tim G. J. Rudner, Karan Singhal, Zachary Nado, Joost van Amersfoort, Andreas Kirsch, Rodolphe Jenatton, Nithum Thain, Honglin Yuan, Kelly Buchanan, Kevin Murphy, D. Sculley, Yarin Gal, Zoubin Ghahramani, Jasper Snoek, and Balaji Lakshminarayanan. Plex: Towards reliability using pretrained large model extensions, 2022. URL https://arxiv.org/abs/2207.07411.

Adaku Uchendu, Daniel Campoy, Christopher Menart, and Alexandra Hildenbrandt. Robustness of bayesian neural networks to white-box adversarial attacks. In *2021 IEEE Fourth International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, pages 72–80. IEEE, 2021.

Joost van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In *International Conference on Machine Learning*, 2020.

Martin J Wainwright and Michael I Jordan. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers Inc., Hanover, MA, USA, 2008. ISBN 1601981848.

Ke Wang, Geoff Pleiss, Jacob Gardner, Stephen Tyree, Kilian Q Weinberger, and Andrew Gordon Wilson. Exact Gaussian processes on a million data points. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors,

Advances in Neural Information Processing Systems, volume 32, pages 14648–14659. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/01ce84968c6969bdd5d51c5eeaa3946a-Paper.pdf.

Nanyang Ye and Zhanxing Zhu. Bayesian adversarial learning. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/586f9b4035e5997f77635b13cc04984c-Paper.pdf.

Huimin Zeng, Zhenrui Yue, Yang Zhang, Ziyi Kou, Lanyu Shang, and Dong Wang. On attacking out-domain uncertainty estimation in deep neural networks. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 4893–4899. International Joint Conferences on Artificial Intelligence Organization, 7 2022. doi: 10.24963/ijcai.2022/678. URL https://doi.org/10.24963/ijcai.2022/678. Main Track.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7472–7482. PMLR, 2019.

Jiaru Zhang, Yang Hua, Zhengui Xue, Tao Song, Chengyu Zheng, Ruhui Ma, and Haibing Guan. Robust bayesian neural networks by spectral expectation bound regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3815–3824, June 2021.

Roland S Zimmermann. Comment on "Adv-BNN: Improved adversarial defense through robust bayesian neural network". *arXiv preprint arXiv:1907.00895*, 2019.

Supplementary Material

1. Background & Preliminaries

We consider supervised learning tasks with N i.i.d. (independent and identically distributed) data realizations $\mathcal{D} = \{\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\}_{i=1}^{N} = (\mathbf{x}_{\mathcal{D}}, \mathbf{y}_{\mathcal{D}})$ of inputs $\mathbf{x} \in \mathcal{X}$ and labels $\mathbf{Y} \in \mathcal{Y}$ with input space $\mathcal{X} \subseteq \mathbb{R}^D$ and label space $\mathcal{Y} \subseteq \mathbb{R}^K$ for regression and $\mathcal{Y} \subseteq \{0,1\}^K$ for classification tasks with K classes.

1.1. Adversarial Examples

An adversarial example for an accurate machine learning model/classifier f is an input that is indistinguishable from a "natural" one (measured in some metric—typically an ℓ_p ball), yet it is being misclassified by f. Adversarial examples in deep learning systems were first observed in Szegedy et al. (2014). Since then many approaches in generating such examples have been proposed - called *adversarial attacks* (Goodfellow et al., 2015; Papernot et al., 2017; Carlini and Wagner, 2017b; Chen et al., 2017; Kurakin et al., 2017), and subsequently methods for shielding models against them—called *defenses* (Goodfellow et al., 2015; Papernot et al., 2016; Madry et al., 2018). Many such defenses have been later found to be broken—either by carefully implementing already known attacks or by *adapting* to the defense (Carlini and Wagner, 2017b; Tramèr et al., 2020; Carlini et al., 2019b).

Formally, in the context of image recognition for the ℓ_∞ distance, the process of generating an adversarial example $\tilde{\mathbf{x}} = \mathbf{x} + \boldsymbol{\eta}$ for a classifier f and a natural input $\mathbf{x}$ involves the solution of the following optimization problem:

$$\boldsymbol{\eta} = \arg\max_{\|\boldsymbol{\eta}\|_\infty \leq \epsilon} \mathcal{L}(f(\mathbf{x} + \boldsymbol{\eta}), y), \tag{1.1}$$

for some $\epsilon > 0$ that quantifies the dissimilarity between the two examples. There is some flexibility in the choice of the loss function $\mathcal{L}$, but typically it is chosen to be the cross-entropy loss. In general, this is a non-convex problem and one can resort to first-order methods (Goodfellow et al., 2015):

$$\tilde{\mathbf{x}} = \mathbf{x} + \epsilon \cdot \operatorname{sign}\left(\nabla_{\mathbf{x}} \mathcal{L}(f(\mathbf{x}), y)\right), \tag{1.2}$$

or iterative versions for solving it (Kurakin et al., 2017; Madry et al., 2018). The former method is usually called *Fast Gradient Sign Method (*FGSM*)* and the latter *Projected Gradient Descent (*PGD*)*.

When a classifier is stochastic, the adversarial defense community argues that the attack should target the loss of the expected prediction (Expectation over Transformation; Athalye et al. (2018a,b)):

$$\tilde{\mathbf{x}} = \mathbf{x} + \epsilon \cdot \operatorname{sign}\left(\nabla_{\mathbf{x}} \mathcal{L}(\mathbb{E} f(\mathbf{x}), y)\right), \tag{1.3}$$

where the expectation is over the randomness at prediction time. Note that some works use an expectation of gradients instead (e.g., Gao et al. (2022)). In the common case of convex loss functions, this is a weaker attack, however.

One common pitfall when evaluating robustness with gradient-based attacks is what has been called *obfuscated gradients* (Athalye et al., 2018a), when the gradients become too uninformative (or zero) to solve the optimization in Equation (1.1). A suite of alternative adaptive attack benchmarks (*AutoAttack*) has been developed to allow for standardized robustness evaluation (Carlini et al., 2019a; Croce and Hein, 2020; Croce et al., 2021)

1.2. Bayesian Neural Networks and Gaussian Processes

Consider a neural network $f(\cdot\,;\boldsymbol{\Theta})$, defined in terms of stochastic parameters $\boldsymbol{\Theta} \in \mathbb{R}^P$. For an observation model $p_{\mathbf{Y}|\mathbf{X},\boldsymbol{\Theta}}$ and a prior distribution over parameters $p_{\boldsymbol{\Theta}}$, Bayesian inference provides a mathematical formalism for finding the posterior distribution over parameters given the observed data, $p_{\boldsymbol{\Theta}|\mathcal{D}}$ (MacKay, 1992; Neal, 1996). However, since neural networks are non-linear in their parameters, exact inference over the stochastic network parameters is analytically intractable.

Full-batch *Hamiltonian Monte Carlo* (HMC) is a Markov Chain Monte Carlo method that produces asymptotically exact samples from the posterior distribution (Neal, 2010) and is commonly referred to as the "gold standard" for inference in Bayesian neural networks (BNN). However, full-batch HMC does not scale to large neural networks and is in practice limited to models with only a few hundred thousand parameters (Izmailov et al., 2021).

Variational inference is an approximate method that seeks to avoid the intractability of exact inference and the limitations of HMC by framing posterior inference as a variational optimization problem. Under this approach, we can obtain a BNN defined in terms of a variational distribution over parameters $q_{\boldsymbol{\Theta}}$ by solving the maximization problem

$$\min_{q_{\boldsymbol{\Theta}} \in \mathcal{Q}_{\boldsymbol{\Theta}}} \mathbb{D}_{\mathrm{KL}}(q_{\boldsymbol{\Theta}} \parallel p_{\boldsymbol{\theta}|\mathcal{D}}) \Longleftrightarrow \max_{q_{\boldsymbol{\Theta}} \in \mathcal{Q}_{\boldsymbol{\Theta}}} \mathcal{F}(q_{\boldsymbol{\Theta}}),$$

where $\mathcal{F}(q_{\boldsymbol{\Theta}})$ is the variational objective

$$\mathcal{F}(q_{\boldsymbol{\Theta}}) \doteq \mathbb{E}_{q_{\boldsymbol{\Theta}}}[\log p_{\mathbf{Y}|\mathbf{X},\boldsymbol{\Theta}}(\mathbf{y}_{\mathcal{D}} \,|\, \mathbf{x}_{\mathcal{D}}, \boldsymbol{\theta}; f)] - \mathbb{D}_{\mathrm{KL}}(q_{\boldsymbol{\Theta}} \parallel p_{\boldsymbol{\Theta}}),$$

and $\mathcal{Q}_{\boldsymbol{\Theta}}$ is a variational family of distributions (Wainwright and Jordan, 2008).

Unlike HMC, variational inference is not guaranteed to converge to the exact posterior, unless the variational objective is convex in the variational parameters and the exact posterior is a member of the variational family. Various approximate inference methods have been developed based on the variational problem above. These methods make different assumptions about the variational family $\mathcal{Q}_{\boldsymbol{\Theta}}$ and therefore result in different posterior approximations. Two particularly simple methods are *Monte Carlo Dropout* (MCD; Gal and Ghahramani (2016)) and *Parameter-Space Variational Inference* under a mean-field assumption (PSVI; also referred to as Bayes-by-Backprop; Graves (2011); Blundell et al. (2015)). These methods enable stochastic (i.e., mini-batch-based) variational inference and can be scaled to large neural networks (Hoffman et al., 2013). However, both methods have been shown to result in poor posterior approximations (Izmailov et al., 2021) and poor predictive uncertainty estimates (Band et al., 2021). More recent work on *Function-Space Variational Inference* in BNNs (FSVI; Rudner et al. (2022)) frames variational inference as optimization over induced functions and has been demonstrated to result in state-of-the-art predictive uncertainty estimates on benchmarking tasks such as MNIST, FashionMNIST, and CIFAR-10 as well as on safety-critical real-world problems.

Gaussian processes (GPs; Rasmussen and Williams (2006)) are a non-parametric alternative to BNNs. They correspond to infinitely-wide stochastic neural networks (Neal, 1996) and allow for exact posterior inference in small-data regression tasks but require approximate inference methods to be applied to classification tasks and to datasets with more than a few ten thousand data points (Snelson and Ghahramani, 2006; Hensman et al., 2013, 2014).

While recent work has enabled exact posterior inference for GPs in larger datasets using approximate matrix inversion methods (Wang et al., 2019), classification, and especially prediction tasks with high-dimensional input data such as images, require parametric feature extractors and approximate methods, for which there are no guarantees that the approximate posterior distribution faithfully represents the exact posterior, and underperform neural networks and BNNs.

1.3. Uncertainty in Bayesian Neural Networks

To reason about the predictive uncertainty of Bayesian neural networks in classification settings, we decompose the total uncertainty of a predictive distribution into its constituent parts: The aleatoric uncertainty of a model's predictive distribution is the uncertainty inherent in the data (according to the model), and a model's epistemic uncertainty (or *model* uncertainty) denotes its uncertainty based on constraints on the model (e.g., the number of parameters, inductive biases, optimization routines, etc.). Mathematically, we can then express predictive uncertainty as

$$\underbrace{\mathcal{H}(\mathbb{E}_{q_{\boldsymbol{\Theta}}}[p_{\mathbf{Y}|\mathbf{X},\boldsymbol{\Theta}}(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}; f)])}_{\text{Total Uncertainty}} = \underbrace{\mathbb{E}_{q_{\boldsymbol{\Theta}}}[\mathcal{H}(p_{\mathbf{Y}|\mathbf{X},\boldsymbol{\Theta}}(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}; f))]}_{\text{Expected Data Uncertainty}} + \underbrace{\mathcal{I}(\mathbf{Y}; \boldsymbol{\Theta})}_{\text{Model Uncertainty}}, \tag{1.4}$$

where $\mathcal{H}(\cdot)$ is the entropy functional and $\mathcal{I}(\mathbf{Y}; \boldsymbol{\Theta})$ is the mutual information (Shannon and Weaver, 1949; Cover and Thomas, 1991; Depeweg et al., 2018). The expected data uncertainty is also commonly referred to irreducible or aleatoric uncertainty, and the model uncertainty is commonly referred to as reducible or epistemic uncertainty.

1.4. Selective Prediction

Selective prediction modifies the standard prediction pipeline by introducing a rejection class, $\perp$, via a gating mechanism defined by a selection function $s : \mathcal{X} \to \mathbb{R}$ that determines whether a prediction should be made for a given input point $\mathbf{x} \in \mathcal{X}$ (El-Yaniv and Wiener, 2010). For a rejection threshold τ, the prediction model is then given by

$$(p(\mathbf{y} \mid \cdot, \boldsymbol{\theta}; f), s)(\mathbf{x}) = \begin{cases} p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}; f) & s \leq \tau \\ \perp & \text{otherwise.} \end{cases} \tag{1.5}$$

A variety of methods have been proposed to find a selection function s (Rabanser et al., 2022). Bayesian neural networks offer an automatic mechanism for doing so, since their posterior predictive distributions do not only reflect the level of noise in the data distribution via the model's aleatoric uncertainty—which can also be captured by deterministic neural networks—but also the level of uncertainty due to the model itself, for example, due to limited accesses training data or an inflexible model class, via the model's epistemic uncertainty. As such, the total uncertainty of a BNN's posterior predictive distribution reflects both uncertainty that can be derived from the training data and uncertainty about a model's limitations. The selective prediction model is then

$$(p(\mathbf{y} \mid \cdot, \boldsymbol{\theta}; f), \mathcal{H}(\mathbb{E}_{q_{\boldsymbol{\Theta}}}[p(\mathbf{y} \mid \cdot, \boldsymbol{\theta}; f)]))(\mathbf{x}) = \begin{cases} p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}; f) & \mathcal{H}(\mathbb{E}_{q_{\boldsymbol{\Theta}}}[p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}; f)]) \leq \tau \\ \perp & \text{otherwise,} \end{cases} \tag{1.6}$$

that is, a point $\mathbf{x} \in \mathcal{X}$ will be placed into the rejection class if the model's predictive uncertainty is above a certain threshold. To evaluate the predictive performance of a prediction model $(p(\mathbf{y} \mid \cdot, \boldsymbol{\theta}; f), s)(\mathbf{x})$, we compute the predictive performance of the classifier $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}; f)$ over a range of thresholds τ. Successful selective prediction models obtain high cumulative accuracy over many thresholds.

2. Prior Work

A growing body of somewhat contradictory work on the adversarial robustness of BNNs has left the community without concensus on this scientific question. Claims of inherent adversarial robustness of Bayesian inference have propagated a belief that BNNs enjoy enhanced adversarial robustness compared to their deterministic counterparts, or that BNNs allow for successful detection of adversarially perturbed data. Other works show how Bayesian methods can be broken by adversarial perturbations. Adversarial attacks have been adapted to the stochastic nature of BNNs, some of them plagued by pitfalls resembling those of the adversarial ML community a couple of years earlier. Here, we aim to survey and classify existing work. To align this survey with our own results, we first outline prior work on adversarial example (AE) *detection* with BNNs, before proceeding to robustness of *classification* with BNNs. Note that while AE detection seems an easier task than robust classification, recent work (Tramer, 2022) shows that there is a reduction from detection to classification (albeit a computationally inefficient one), which means that claims of a high-confidence AE detector should receive equal scrutiny as robust classification claims would. In particular, after nearly a decade of work by an ever-growing community, only robustness results achieved with adversarial training (Madry et al., 2018) have stood the test of time and constitute today's benchmark (Croce et al., 2021).

Adversarial Example Detection with BNNs. A first set of early works has investigated model confidence on adversarial samples by looking at Bayesian uncertainty estimates using the intuition that adversarial examples lie off the true data manifold. Feinman et al. (2017) give a first scheme using uncertainty estimates in dropout neural networks, claiming AE detection, which is subsequently broken in Carlini and Wagner (2017a) (who, incidentally break most AE detection schemes of their time). Rawat et al. (2017) analyze four Bayesian methods and claim good AE detection using various uncertainty estimates, but analyze only weak FGSM attacks on MNIST[1]. Smith and Gal (2018) claim to provide empirical evidence that epistemic uncertainty of MCD could help detect stronger adversarial attacks (FGSM and BIM, a variant of PGD) on a more sophisticated cats-and-dogs dataset (refuted in Section 2). Bekasov and Murray (2018) evaluate AE detection ability of MCMC and PSVI, but do so only in a simplified synthetic data setting. The first to demonstrate adversarial vulnerability of Bayesian AE detection (though not for BNNs) are Grosse et al. (2018), who attack both accuracy and uncertainty of the Gaussian Process classifier. Several works leave the BNN-inference framework and thus are not our focus: by way of example Deng et al. (2021) design a Bayesian tack-on module for AE detection and Li et al. (2021) add Gaussian noise to all parameters in deterministic networks to generate distributions on each hidden representation for AE detection. To the best of our knowledge, our work is the

1. It is widely accepted that many proposed adversarial defenses fail to generalize from MNIST (Carlini and Wagner, 2017a).

first to demonstrate adversarial vulnerability of *modern* Bayesian inference methods that enjoy inference guarantees (while examining and refuting previous claims of robustness of BNNs).

Robustness of BNNs. Several recent works have hypothesized that BNN posteriors enjoy enhanced robustness to adversarial attacks, compared to their deterministic counterparts. Carbone et al. (2020); Bortolussi et al. (2022), guided by considerations of vanishing gradients on the data manifold in the infinite data limit for BNNs, claim to observe robustness to gradient-based attacks like FGSM and PGD for HMC and PSVI using a simple CNN and a fully-connected network (reevaluated in Section 2). Uchendu et al. (2021) examine the robustness of VGG and DenseNet with Variational Inference and claim marginal improvement over their deterministic counterparts. Pang et al. (2021) evaluate Bayesian VGG networks for two inference methods (standard Bayes-by-Backprop and Flipout), claiming evidence for surprising adversarial robustness[2]. Cardelli et al. (2019), De Palma et al. (2021), Grosse et al. (2021), and Patane et al. (2022) study the adversarial robustness of GPs. None of these works benchmark recent Bayesian inference methods like FSVI, which are state-of-the-art for prediction and semantic shift detection. Zhang et al. (2021) propose a regularization that could improve adversarial robustness. Our evaluation both in their and in the standard attack parameter setting shows no significant robustness gain with their regularization (see Section 2). The first in the literature to give negative evidence for robustness of BNNs is Blaas (2021), comparing a BNN trained with HMC to a deterministic network on MNIST and FashionMNIST but observing no difference in robustness.

Robustness of OOD Detection. To the best of our knowledge, there is no work prior to ours examining Bayesian OOD pipelines for their robustness against adversarial attacks. Appendix **??** surveys some related work.

Investigating the BNN prior. Blaas and Roberts (2021) ask how the *prior* could affect adversarial robustness for BNNs. For PSVI on a three-layer fully-connected network, they observe a trade-off between accuracy and robustness (reminiscent of the accuracy-robustness trade-off for deterministic NNs (Zhang et al., 2019)): small priors yield a small Lipschitz constant and thus better robustness but cannot fit the data; for large priors the opposite is true.

Robustness of OOD Detection. The profound advantage of a Bayesian neural network is that it can provide both aleatoric and epistemic uncertainty estimations because of the probabilistic representation of the model. In particular, a large body of work leverages this to create Bayesian pipelines for OOD detection. To the best of our knowledge, there is no work prior to ours examining Bayesian OOD pipelines for their robustness against adversarial attacks and our work is the first to do so. There *is* prior work on robustness of OOD detection with *deterministic* models which we briefly survey here: Sehwag et al. (2019) attack the OOD detection of deterministic models with calibration and temperature scaling. Kopetzki et al. (2021) attack both accuracy and OOD detection for Dirichlet-based models. Zeng et al. (2022) attack out-domain uncertainty estimation for deep ensembles and

2. Both Uchendu et al. (2021) and Pang et al. (2021) have no code released, and the claims in the latter work seem too good to be true. Pang et al. (2021) also distinguishes between "variational inference" and "Bayes-by-Backprop" although Bayes-by-Backprop is a variational inference method. We adversarially break Bayes-by-Backprop (i.e., PSVI) (see Section 3).

uncertainty estimation using radial basis function kernels and GPs. All these works break the OOD detection capabilities of the models.

In a different though related work, Galil and El-Yaniv (2021) attack *in-distribution* data to increase the uncertainty of correctly classified images and decrease the uncertainty of incorrectly classified images, thus targeting selective accuracy in an in-distribution setting for deterministic networks as well as MCD. Note that their attack requires knowledge of the ground truth for the instances it attacks, which might be difficult to attain.

Adversarial training and bnns. Ye and Zhu (2018) introduce uncertainty over the generation of adversarial examples to improve adversarial training. For example, Liu et al. (2018) propose to create robust models with a version of AT for BNNs guided by the intuition that randomness in the model parameters could enhance AT optimization against adversarial attacks. However, Zimmermann (2019) refutes these claims by observing that robustness significantly diminishes when using expected gradients to attack. Uchendu et al. (2021) directly combine adversarial training with BNN by training on iteratively generated adversarial examples, and observe small improvements in robustness. Doan et al. (2022) make the interesting observation that direct AT of BNNs as in Liu et al. (2018) might lead to mode collapse of the posterior and propose a new "information gain" objective for more principled AT of BNNs.

3. Details of Evaluation of Prior Work

Here we provide further details complementing Section 2.

Attacking AE detection in Smith and Gal (2018): In addition to implementing the intended PGD 10 attack on the BNNs, we additionally provide evaluations for two stronger attacks, PGD (=PGD 40) and TransferPGD +. We use the same l_∞ perturbation as in their results, 10/255. PGD directly attacks the BNN accuracy for 40 iterations with the gradient of the loss of the expectation, as discussed in Section 1.1. At each iteration we use 10 samples from the posterior. Figure 1 shows that PGD 10, and more so PGD already fool the AE detector leading it to reject more clean samples than adversarial samples, as witnessed by the decreasing selective accuracy curve. However, when tracking gradients for PGD 40 we find some degree of gradient diminishing. Therefore, we design a stronger attack called TransferPGD +. As its name indicates, it is a two phase attack that consists of a transfer attack that warm-starts a subsequent PGD +. That is, we attack the deterministic non-dropout version of the BNN first (by turning off dropout when computing the loss), and then perform 40 iterations of PGD on the uncertainty estimation of the network starting from the transferred adversarial examples generated in the first phase, again with 10 posterior samples per iteration. We use the label predicted by the BNN to avoid label leakage. TransferPGD + can break the accuracy of the BNN to 2% and fools the AE detector successfully.

Code issues: After investigating the code privided by Smith and Gal (2018), we find two mutually reinforcing problems. First, they evaluate the clean, adversarial, and noisy data separately but do not set batch-normalization layers to evaluation mode. As a result, it is possible that the detector may use batch statistics among the different sample groups to distinguish them. Secondly, when creating the *noisy* data, re-centered images with initial values in [0, 255] are mistakenly clipped to [0, 1], making all noisy data points very similar to each other (since most information is clipped to either 0 or 1), and leading to

very small epistemic uncertainty for noisy images. This leads the BNN to accept the noisy images and reject the clean and adversarial ones, resulting in misleading, overly optimistic ROC curves (this effect was further amplified once we fixed the batch-normalization issue). Moreover, care needs to be applied when using standard packages, developed for vanilla NNs, to BNNs. Deterministic models tend to operate on logits and the softmax and cross-entropy calculations are combined, and therefore standard attack packages apply a softmax function on these pre-loss outputs. BNNs, on the other hand, average the probability predictions from posterior samples after the softmax function, and hence directly produce probabilities. A direct application of standard attack packages to BNNs would apply the softmax function to class probabilities (a "double softmax problem"), thereby making the class probabilities more uniform and weakening the attack strength. We find this problem in the implementation provided by (Smith and Gal, 2018).

Note that Smith and Gal (2018) show the ROC curve and provide AUROC values, while we have chosen to show slective accuracy curves throughout our work. AUROC and ASA are incomparable, so we do not show a comparison of these metrics here. Both curves quantify the performance of the AE detector and our findings (see Figure 1) show failure to detect AE.

Attacking Robustness in Carbone et al. (2020); Bortolussi et al. (2022): As we discuss in Section 2, the BNNs trained in this evaluation tend to have large values before the softmax layer, resulting in gradient vanishing. Therefore, we renormalize the logits by 100 to fix numerical issues when attacking the trained model. We evaluate these adversarial examples on the unnormalized network. With the double softmax corrected, one could use standard PGD on the loss of the expectation as in Eq. (1.3) to break the accuracy to nearly 0% on MNIST with perturbation radius $\epsilon = 0.3$.

Attacking Robustness via Regularization in Zhang et al. (2021): We use the publicly available code to train a Bayesian MLP on MNIST and then use the evaluation code to assess its adversarial robustness for different values of perturbation budget ϵ. Table 4 summarizes the results. We note that once we fix the "double-softmax" problem (by replacing `torch.nn.CrossEntropyLoss` with `torch.nn.NLLLoss` in the source code of the attack), the model essentially becomes non-robust for the standard value of $\epsilon = 0.3$ (the loss effect is more pronounced with an FGSM attack). In Table 4, using their code, we have reevaluated the results in Zhang et al. (2021) in their setting (with the double-softmax) and after fixing it, for $\epsilon = 0.16$, where their work claimed the largest benefit of regularization, and the more standard $\epsilon = 0.3$.

4. Recommendations for Evaluating Robustness of bnn pipelines

Having examined (see Section 2) the three robustness claims, we draw several conclusions about pitfalls and failure modes and list recommendations to avoid them when attacking Bayesian pipelines, similar (and sometimes overlapping) with earlier recommendations in the context of non-Bayesian methods (Carlini and Wagner, 2017a; Athalye et al., 2018a).

1. When considering randomness, use the strongest attack appropriate for the model, all other things being equal. In the case of stochastic models, attack the loss of the average (Equation (1.3)), rather than the average loss (at least for convex losses).

Table 4: A revised evaluation of the robustness of the robust BNN of Zhang et al. (2021) in their setting, and with the double softmax problem fixed. Mean accuracy (± one standard deviation) of a Bayesian MLP trained on MNIST against FGSM and PGD generated data on MNIST.

Radius	**Method**	FGSM	PGD 10
$\epsilon = 0.16$	Original setting	41.23 ±0.32	6.38 ±0.05
	Reevaluated	10.04 ±0.05	7.58 ±0.11
$\epsilon = 0.3$	Original setting	22.30 ±0.11	0.64 ±0.02
	Reevaluated	0.23 ±0.02	0.04 ±0.00

2. Beware of double softmax. All prior works examined in this paper provide implementations that apply the softmax function twice. Nearly all BNNs output the probability prediction, and it is necessary to remove the softmax function from the loss to effectively apply standard attack implementations.

3. Fix all normalization layers but enable all stochastic network components (such as dropout) at test time.

4. Monitor gradient values throughout the attack to avoid numerical issues. The gradients when attacking accuracy should never be vanishing to zero.

5. Increase the radius to check whether the model can be broken. If so, examine for obfuscated gradients (Athalye et al., 2018a). If the model remains robust, attempt to attack a deterministic network using the parameters from the posterior or several fixed samples.

6. If PGD attacks fail, consider using refined attack benchmarks like AutoAttack (Croce and Hein, 2020).

7. Design attacks appropriate for the model pipeline. Consider how to break the model based on its design, such as targeting both accuracy and uncertainty estimation (like the PGD+ attack we introduce in Section 3). Adaptiving attacks to the model quantities can provide more confidence in robustness results.

5. Adversarial Attack Methods

Threat model and principled generation of adversarial perturbations with FGSM, PGD and PGD+. We consider a *full white-box* attack: The adversary has knowledge of the entire model and its parameters, and can obtain samples from its posterior, which is the white-box model considered in most prior work (Carbone et al., 2020; Bortolussi et al., 2022; Zhang et al., 2021). We apply FGSM and PGD with 40 iterations to attack expected accuracy (to break robustness in Section 3.1) or uncertainty (to fool OOD detectors in Section 3.3) as per Equation (1.3). To devise a stronger attack on AE detectors, we also create a combined attack, PGD+: the idea is to produce adversarial examples that both fool

Table 5: Average selective accuracy for semantic shift detection with BNNs.

		CNN				ResNet-18			
		Clean	Noisy	FGSM	PGD	Clean	Noisy	FGSM	PGD
MNIST	HMC	83.92±0.02	83.65±0.11	61.02±1.54	47.16±2.05	-	-	-	-
	MCD	83.88±0.10	83.84±0.17	55.22±0.04	20.01±0.96	82.43±0.94	78.26±4.03	56.62±7.42	16.49±0.45
	PSVI	83.92±0.01	83.83±0.04	48.21±2.37	18.03±0.54	81.67±2.27	77.98±5.62	53.58±4.10	16.37±0.66
	FSVI	84.10±0.01	84.17±0.03	60.00±11.80	15.45±0.01	80.35±5.48	79.16±7.27	71.81±13.25	15.54±0.49
FMNIST	HMC	77.57±0.22	79.04±0.13	61.90±0.54	45.14±1.54	-	-	-	-
	MCD	77.14±0.61	78.26±0.54	40.52±3.42	21.40±0.77	80.32±0.12	81.70±0.08	73.28±0.91	15.39±0.02
	PSVI	75.26±0.40	76.66±0.39	46.54±2.23	16.81±0.17	80.54±0.23	81.62±0.03	77.65±1.36	15.21±0.01
	FSVI	76.75±0.55	79.97±0.27	28.94±2.66	15.15±0.03	81.67±0.08	81.87±0.04	80.00±0.37	15.21±0.00
CIFAR-10	MCD	-	-	-	-	77.51±0.42	78.18±0.26	75.31±1.01	15.31±0.00
	PSVI	-	-	-	-	77.51±0.42	78.18±0.26	75.31±1.01	15.31±0.00
	FSVI	-	-	-	-	80.54±0.17	80.72±0.08	78.15±0.54	15.24±0.01

the classifier (to drive accuracy to zero) and have low uncertainty (lower than the clean samples), resulting in poor predictive accuracy, worsening for higher rejection rates. To this end, PGD+ first attacks BNN accuracy in the first 40 iterates (using the BNN prediction as its label) and, from this starting point, computes another 40 iterates to attack uncertainty within the allowed ϵ-ball around the *original* point. PGD+ does not need the ground truth to attack uncertainty (unlike the uncertainty attack in Galil and El-Yaniv (2021)). We settle on PGD+ after observing empirically that it is stronger than PGD80 targeting accuracy or targeting a linear combination of accuracy and uncertainty, as its two stages are specifically designed to fool AE detectors. Throughout, we use 10 samples from the posterior for each iteration of the attack (see Eq. 1.3). We find it unnecessary to increase the attack sample size for our attacks and hence adopt this number for computational efficiency. We only apply gradient-based attacks; since these are sufficient for breaking all BNNs, we do not need to implement AutoAttack (Croce and Hein, 2020). We showcase ℓ_∞ attacks with standard attack parameters: $\epsilon = 0.3$ for MNIST, $\epsilon = 0.1$ for FashionMNIST, $\epsilon = 8/255$ for CIFAR-10. We only consider *total* uncertainty in our attack and for selective prediction, as we found no evidence for an advantage of using *epistemic* uncertainty or other variants.

5.1. Robustness of Semantic Shift Detection

Setting: Semantic shift detection is a common application of BNNs, where they outperform state-of-the-art deterministic uncertainty quantification methods for NNs (van Amersfoort et al., 2020; Band et al., 2021; Rudner et al., 2022; Tran et al., 2022). Our OOD semantic shift datasets for MNIST, FashionMNIST and CIFAR-10 are FashionMNIST, MNIST, and SVHN, respectively, each of them giving zero accuracy. The test set contains half in-distribution (ID) and half OOD samples, hence selective accuracy curves start at 50% accuracy. We attack only the OOD samples with FGSM and PGD, targeting the uncertainty. Since this is already sufficient to reject most ID samples, we do not also attack the ID samples, though we could conceivably do so.

Results: Complete ASA results are given in Table 5 and selective accuracy curves are shown in Figures 3 (for MNIST, using a CNN) (and 7, 4, 5, 6 in Appendix 6 for the other datasets and architectures). Our results resemble what we have seen for AE detection. The PGD attack nearly completely fools the detector to reject all ID samples, reaching close to

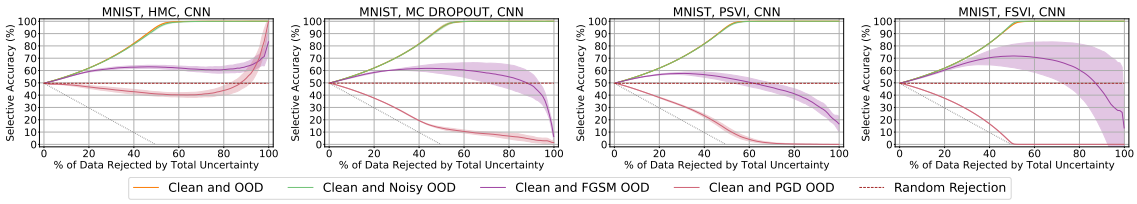

Figure 3: OOD detection statistics for HMC, MCD, PSVI and FSVI for MNIST with a CNN.

0% accuracy for 50% rejection rate and indeed most methods give ASA close to the lower bound of 12.5%, with only HMC showing higher ASA. Still, even for HMC ASA is below 50%, meaning it performs worse than random rejection. As before, FGSM attacks are too weak to turn the direction of the selective accuracy curve.

6. Additional Figures and Tables for Results Section

Table 6 shows the average neagative log likelihood (ANLL) for AE detection on Section 3.2. The NLL is an evaluation metric of interest, since it reflects the degree of confidence in a prediction and penalizes underconfident correct predictions as well as overconfidenyt wrong predictions. For classification models, the NLL is given by the cross-entropy loss between the one-hot labels and the predicted class probabilities.

Table 6: Detecting Adversarial Examples with BNNs- Average Negative Log-Likelihood

			Clean	Noisy	FGSM	PGD	PGD +
MNIST	CNN	HMC	0.20±0.00	0.10±0.04	1.51±0.15	4.06±0.23	4.45±0.51
		MCD	0.00 ±0.00	0.00 ±0.00	1.70 ±0.04	10.42 ±0.17	10.03 ±0.24
		PSVI	0.00±0.00	0.00±0.00	2.48±0.12	8.75±0.29	10.84±0.12
		FSVI	0.02 ±0.00	0.08 ±0.03	0.42 ±0.09	7.48 ±0.36	10.34 ±0.22
	ResNet-18	MCD	0.00 ±0.00	2.34 ±0.37	1.60 ±0.11	10.60 ±0.09	11.56 ±0.02
		PSVI	0.00 ±0.00	1.09 ±0.17	1.53 ±0.17	11.11 ±0.05	11.58 ±0.65
		FSVI	0.04±0.05	0.59±0.53	0.84±0.37	9.12±0.66	10.29±0.75
FMNIST	CNN	MCD	0.05 ±0.00	0.05 ±0.00	2.77 ±0.07	9.34 ±0.11	9.78 ±0.07
		PSVI	0.05 ±0.00	0.05 ±0.00	4.15 ±0.30	9.23 ±0.07	10.08 ±0.06
		FSVI	0.08 ±0.00	0.12 ±0.01	1.83 ±0.12	8.75 ±0.28	9.73 ±0.25
	ResNet-18	MCD	0.06 ±0.00	0.09 ±0.00	3.58 ±0.27	11.01 ±0.06	11.06 ±0.03
		PSVI	0.10 ±0.01	0.15 ±0.01	1.63 ±0.71	9.84 ±0.82	10.01 ±0.72
		FSVI	0.08 ±0.01	0.12 ±0.01	1.69 ±0.19	10.65 ±0.03	10.86 ±0.06
CIFAR10	ResNet-18	MCD	0.04 ±0.00	0.04 ±0.00	2.67 ±0.03	10.64 ±0.07	11.08 ±0.03
		PSVI	0.04 ±0.01	0.05 ±0.01	1.61 ±0.35	10.18 ±1.38	10.44 ±1.08
		FSVI	0.06 ±0.01	0.06 ±0.01	1.08 ±0.03	10.18 ±0.09	10.62 ±0.07

In addition to the selective prediction curves shown in Sections 3.2 and 3.3, we also generate the full sets for each dataset-method-architecture setting. The results are shown in Figure 4 for MNIST+ResNet-18, Figure 5 for FashionMNIST+CNN, Figure 6 for FashionMNIST+ResNet-18 and Figure 7 for CIFAR-10+ResNet-18.

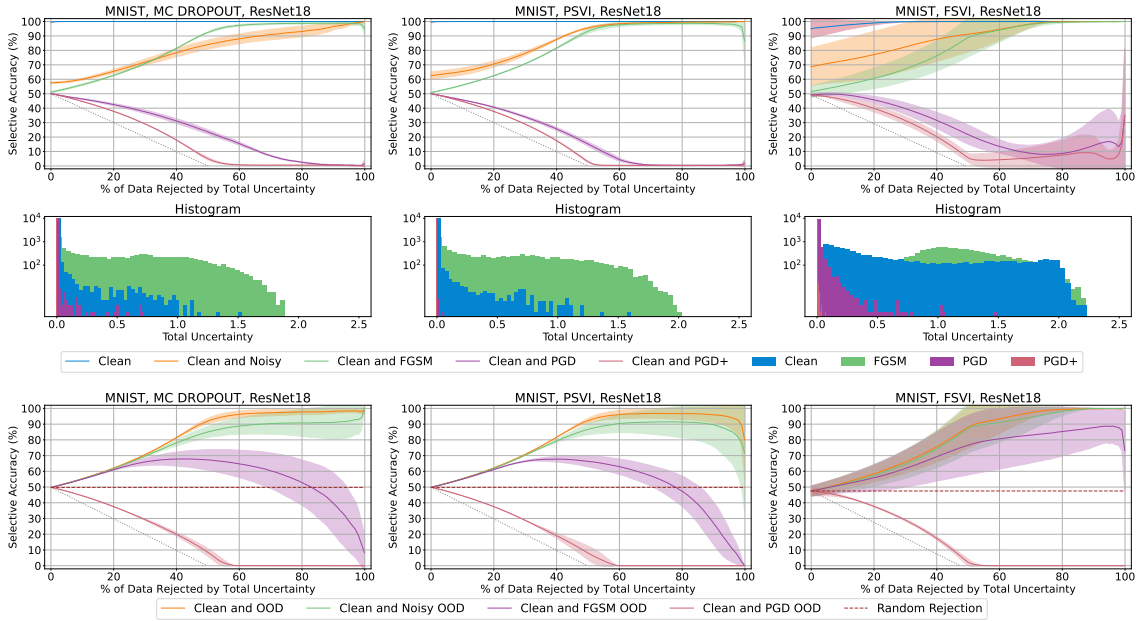

Figure 4: AE and OOD detection statistics for MCD, PSVI, and FSVI on Fashion MNIST with a ResNet-18 architecture

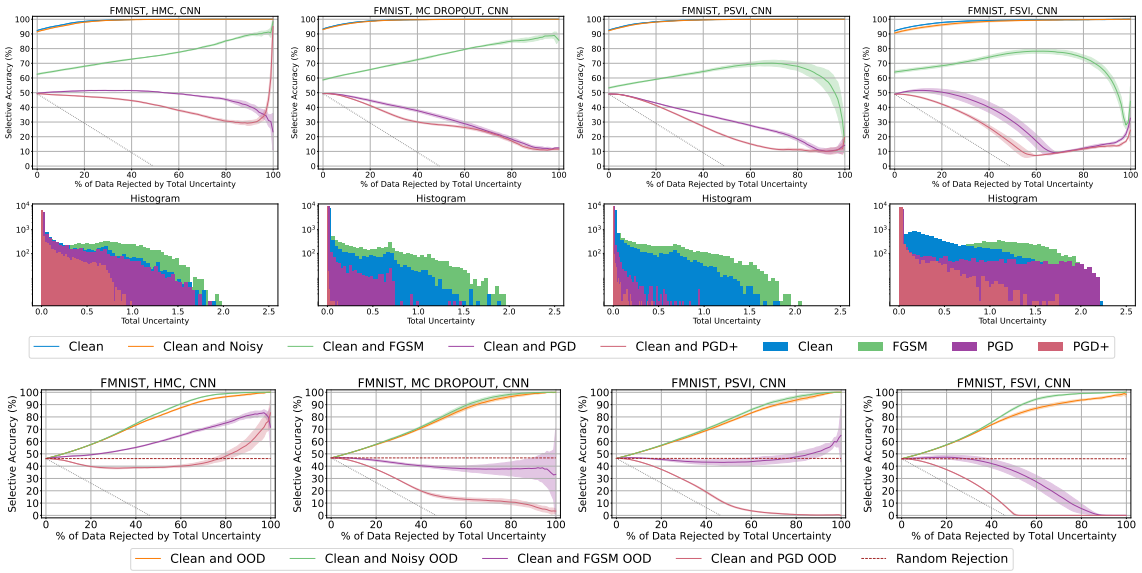

Figure 5: AE and OOD detection statistics for HMC, MCD, PSVI, and FSVI on Fashion MNIST with a CNN architecture

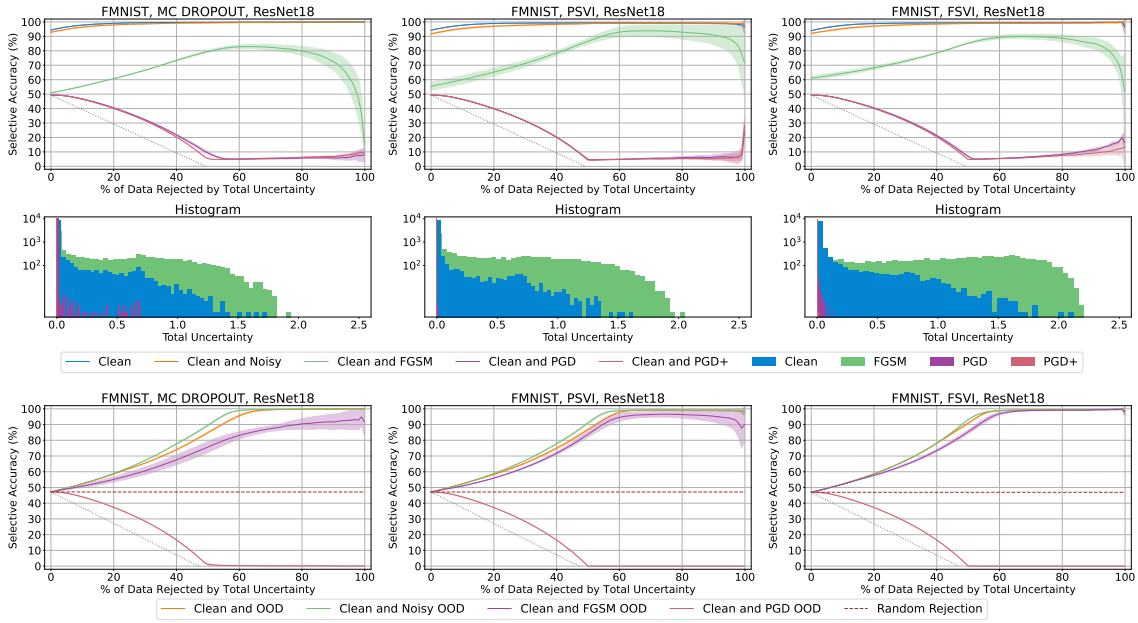

Figure 6: AE and OOD detection statistics for MCD, PSVI, and FSVI on Fashion MNIST with a ResNet-18 architecture

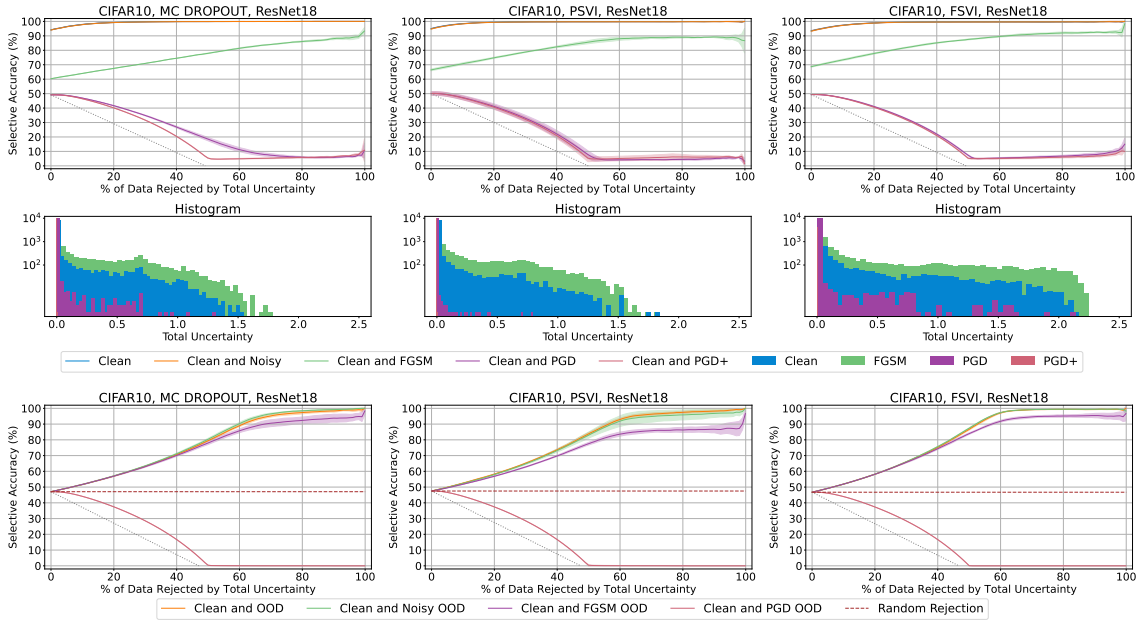

Figure 7: AE and OOD detection statistics for MCD, PSVI, and FSVI on CIFAR-10 with a ResNet-18 architecture.

7. Experimental Details for our Results

Here, we provide a detailed description of our experimental setup in Section 3.

Architectures For CNN models, we use a four-layer CNN for all the experiments. The architecture for CNN is shown in Table 7. We use the standard ResNet-18 from He et al. (2016). For MCD, we add dropout after every activation function for both CNN and ResNet-18.

Hyperparameters We implement HMC using the `Hamiltorch` package from (Cobb and Jalaian, 2021) and apply it to MNIST with a CNN model due to the lack of scalability. To optimize GPU memory usage, we use 10,000 training and 5,000 validation samples from the MNIST dataset. We deploy 100 samples burnin and generate another 100 samples from the posterior. For each sample, we train the model for 20 steps with 0.001 as the step size. Such configuration already yields a HMC BNN with around 96% accuracy on the test set. The hyperparameters for FSVI, PSVI, and MCD are shown in Table 8, Table 9, and Table 10.

Table 7: The Architecture of the four-layer CNN

```
nn.Conv(out_features=32, kernel_size=(3, 3))
ReLU()
max_pool(window_shape=(2, 2), strides=(2, 2), padding="VALID")
nn.Conv(out_features=64, kernel_size=(3, 3))
ReLU()
max_pool(window_shape=(2, 2), strides=(2, 2), padding="VALID")
reshape to flatten
nn.fc(out_features=256
ReLU()
nn.fc(out_features=num_classes)
```

Table 8: Hyperparameters for FSVI

	CIFAR10+ResNet-18	FMNIST+ResNet-18	FMNIST+CNN	MNIST+ResNet-18	MNIST+CNN
Prior Var	100,000	100,000	1,000,000	100,000	1,000,000
Prior Mean	0	0	0	0	0
Epochs	200	30	200	10	200
Batch Size	128	128	128	128	128
Context Batch Size	128	128	16	128	16
Learning Rate	0.005	0.005	0.05	0.005	0.05
Momentum	0.9	0.9	0.9	0.9	0.9
Weight Decay	0	0	0	0	0
Alpha	0.05	0.05	0.05	0.05	0.05
Reg Scale	1	1	1	1	1

Table 9: Hyperparameters for PSVI

	CIFAR10+ResNet-18	FMNIST+ResNet-18	FMNIST+CNN	MNIST+ResNet-18	MNIST+CNN
Prior Var	1	1	1	1	1
Prior Mean	0	0	0	0	0
Epochs	200	50	200	10	200
Batch Size	128	128	128	128	128
Learning Rate	0.005	0.005	0.05	0.005	0.05
Momentum	0.9	0.9	0.9	0.9	0.9
Weight Decay	0	0	0	0	0
Alpha	0.05	0.05	0.05	0.05	0.05
Reg Scale	1	1	1	1	1

Table 10: Hyperparameters for MCD

	CIFAR10+ResNet-18	FMNIST+ResNet-18	FMNIST+CNN	MNIST+ResNet-18	MNIST+CNN
Prior Precision	0.0005	0.0005	0.0001	0.0005	0.0001
Dropout Rate	0.1	0.1	0.1	0.1	0.1
Epochs	200	30	200	10	200
Batch Size	128	128	128	128	128
Learning Rate	0.005	0.005	0.05	0.005	0.05
Momentum	0.9	0.9	0.9	0.9	0.9
Weight Decay	0	0	0	0	0
Alpha	0.05	0.05	0.05	0.05	0.05
Reg Scale	1	1	1	1	1

8. Varying the strength of the adversarial attacks

Uncertainty estimates with BNNs perform correctly for clean and noisy samples, yet fail under adversarial attacks with standard perturbation strength. It would thus be interesting to explore, when varying the strength of the attack, at which point uncertainty estimates cannot be trusted. To investigate this, we apply the PGD attack with varying radius to understand the landscape of uncertainty estimates, for FSVI on CIFAR-10 with ResNet-18. As shown in the left plot of Figure 8, using total uncertainty for selective prediction is better than random rejection up to about a radius of 1/255. Beyond this radius, the uncertainty estimates are completely deceived by the attacks. The right plot compares the performance of the various inference methods, MCD, PSVI, and FSVI, in the same setting. MCD achieves the best Average Selective Accuracy (ASA) among the three methods, declining most gracefully with increased attack strength.

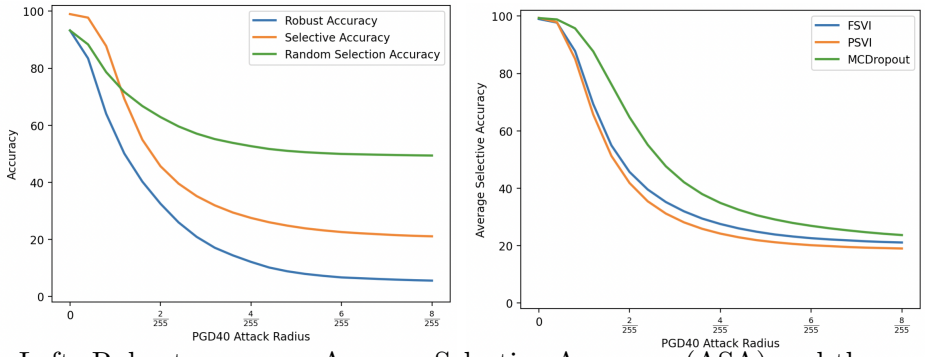

Figure 8: Left: Robust accuracy, Average Selective Accuracy (ASA) and the accuracy for Random Selection for varying PGD strength for FSVI on CIFAR10 with ResNet-18. We observe that for small radius up to $\approx \frac{1}{255}$ ASA exceeds Random Selection Accuracy, allowing for some AE detection. Right: ASA for FSVI, PSVI and MCD on CIFAR10 with ResNet-18. AE detection capability of MCD declines most gracefully with increasing attack strength.