NON-SINGULAR ADVERSARIAL ROBUSTNESS OF NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Adversarial robustness has become an ever accelerating challenge for neural network owing to its over-sensitivity to small input perturbations. While being critical, we argue that solving this singular issue alone fails to provide a comprehensive robustness assessment. Even worse, the conclusions drawn from singular robustness may give a false sense of overall model robustness. Specifically, our findings show that adversarially trained models that are robust to input perturbations are still (or even more) vulnerable to weight perturbations when compared to standard models. In this paper, we formalize the notion of non-singular adversarial robustness for neural networks through the lens of joint perturbations to data inputs as well as model weights. To our best knowledge, this study is the first work considering simultaneous input-weight adversarial perturbations. Based on a multi-layer feed-forward neural network model with ReLU activation functions and standard classification loss, we establish error analysis for quantifying the loss sensitivity subject to ℓ_∞-norm bounded perturbations on data inputs and model weights. Based on the error analysis, we propose novel regularization functions for robust training and demonstrate improved non-singular robustness against joint input-weight adversarial perturbations.

1 INTRODUCTION

In spite of accomplishments achieved by machine learning in many tasks such as object recognition, speech recognition and so on, predictors remain to fail miserably under the presence of imperceptible perturbations, often known as "adversarial examples" (Szegedy et al., 2014; Goodfellow et al., 2014). These examples have long been the essence of many algorithms concerning adversarial robustness. The formal notion and framework gradually developed with the rise of such algorithms (Fawzi et al., 2017; Biggio & Roli, 2018).

Concretely, adversarial examples are often created from unperturbed data within a norm-ball of radius ϵ. Moreover, the robustness of a model is largely defined as the minimum perturbation that the input could change to alter the network's correct output (Hein & Andriushchenko, 2017; Weng et al., 2018). In Weng et al. (2020), the definition is to fit weight (model parameters) perturbation, another type of attack that could cause model to ill-perform. We note that considering input or weight perturbation alone is myopic and incomplete, as it only contributes to *singular* adversarial robustness assessment. Furthermore, in Section 4 (Fig.1), we show that models trained under only input perturbation would still suffer when encountering weight perturbation, and vice versa, which suggests that those two singular robustness results poses the risk of offering limited, or even false, sense of the comprehensive model robustness.

This paper resolves the gap by formalizing *non-singular adversarial robustness* of neural networks and studying simultaneous input-weight perturbations. We develop theorems bounds concerning pairwise margin on the classification loss for multi-layer neural networks with ReLU activations. Moreover, we propose a new loss function based on previous margin bounds for training a robust neural network against simultaneous perturbations and validate its effectiveness via experiments.

2 Related Works

Recent results discovered that the performance of a well-trained model could be devastated by adversarial examples using either gradient-based approaches (Goodfellow et al., 2014; Kurakin et al., 2017; Moosavi-Dezfooli et al., 2016; Carlini & Wagner, 2017; Chen et al., 2018; Xu et al., 2019) or prediction outputs (Chen et al., 2017; Tu et al., 2019; Cheng et al., 2019). Several methods were proposed later for investigating adversarial robustness. The state-of-the-art model by (Madry et al., 2018) makes avail of a procedure known as adversarial training, where the model weights are updated by minimizing the worst-case adversarial perturbations, forming a min-max training objective. Wang et al. (2019) further proves the convergence of such training process. On the other hand, Wu et al. (2020) demonstrated that by taking the landscape of loss-weight surface into account, the robust generalization gap under adversarial training would be narrowed. We note that this is different from our setting where we consider robustness from both the input and parameter aspects.

Aside from input perturbation, Liu et al. (2017) and Zhao et al. (2019) proposed fault-injection attacks which randomize the model parameters stored in memory by physically changing the logical bits of the memory storage. Widrow & Lehr (1990) and Cheney et al. (2017) studied weight perturbations applied on the internal architecture for generalization. Weng et al. (2020) showed that by taking weight sensitivity into account, the model could maintain its performance after weight quantization. Furthermore, Zhao et al. (2020) demonstrated that by taking advantages of mode connectivity of the model's parameters, one could mitigate or preclude the attacks based on weight perturbations. As above results have shown that either part of the perturbations have been studied explicitly while joint perturbation remains equivocal. Moreover, we note that adversarial training subject to weight perturbations is not meaningful since the min-max procedure would fail in only the parameter space. In this work, we consider when input and weight are both perturbed and prove bounds for training a non-singular robust neural network against joint perturbations.

3 Main Results

We first define in Section 3.1 the mathematical notation and preliminary. In Section 3.2, we introduce the main theorem. Finally, in Section 3.3, we proceed to develop a theory-driven loss function.

3.1 Notations and Preliminary

We start by offering some mathematical notations used in this paper. Let $[L]$ be the set containing all positive integers smaller than L. As for the notation of vectors, boldface lowercase letter are used (e.g. $\mathbf{x}$) and the i-th element is marked as $[\mathbf{x}]_i$. Meanwhile, matrices are denoted by boldface uppercase letter(e.g. $\mathbf{W}$). Given a matrix $\mathbf{W} \in \mathcal{R}^{m\times n}$, we write its i-th row, j-th column and (i,j) element as $W_{i,:}$, $W_{:,j}$,and $W_{i,j}$ respectively. The matrix (α,β) norm is written as $\|\mathbf{W}\|_{\alpha,\beta}$. In the following sections, we would adopt the notion of vector-induced norm upon mentioning (α,β) norm of a given matrix $\mathbf{W}$; namely, we have $\|\mathbf{W}\|_{\alpha,\beta} = \max_{\mathbf{x}\neq 0} \frac{\|\mathbf{W}\mathbf{x}\|_\alpha}{\|\mathbf{x}\|_\beta}$. We may use the shorthand notation $\|\cdot\|_p := \|\cdot\|_{p,p}$. Furthermore, we use the notion of $\mathbb{B}^\infty_{\mathbf{W}}(\epsilon)$ to express an element-wise ℓ_∞ norm ball for both matrix and vector. Specifically, given a matrix $\mathbf{W} \in \mathcal{R}^{m\times n}$ and vector $\mathbf{x} \in \mathcal{R}^n$, we could define the norm ball as $\mathbb{B}^\infty_{\mathbf{W}}(\epsilon) := \{\hat{\mathbf{W}} \,|\, |\hat{W}_{i,j} - W_{i,j}| \leq \epsilon, \forall i \in [m], j \in [n]\}$ and $\mathbb{B}^\infty_{\mathbf{x}}(\epsilon) := \{\hat{\mathbf{x}} \,|\, |[\hat{\mathbf{x}}]_j - [\mathbf{x}]_j| \leq \epsilon, \forall j \in [n]\}$.

Preliminary We study K-class classification problem and consider an input vector $\mathbf{x} \in \mathcal{R}^d$ and an L-layer neural network defined as $f_{\boldsymbol{W}}(\mathbf{x}) = \mathbf{W}^L\rho(\mathbf{W}^{L-1}...\rho(\mathbf{W}^1\mathbf{x})) \in \mathcal{R}^K$ with $\boldsymbol{W}$ being the set containing all weight matrices (i.e. $\boldsymbol{W} := \{\mathbf{W}^i \,|\, \forall i \in [L]\}$ while $\rho(\cdot)$ stands for non-negative monotone activation functions and are assumed 1-Lipschitz. Moreover, the i-th component of neural network's output is written as $[f_{\boldsymbol{W}}(\mathbf{x})]_i$ and the pairwise margin is denoted as $f^{ij}_{\boldsymbol{W}}(\mathbf{x}) := [f_{\boldsymbol{W}}(\mathbf{x})]_i - [f_{\boldsymbol{W}}(\mathbf{x})]_j$. The output of k-th ($k \in [L-1]$) layer given a certain set of matrices $\boldsymbol{W}$ under both unperturbed and input-perturbed setting is $\mathbf{z}^k_{\boldsymbol{W}} := \rho(\mathbf{W}^k...\rho(\mathbf{W}^1\mathbf{x})), \mathbf{W}^m \in \boldsymbol{W}, \forall m \in [k]$ and $\hat{\mathbf{z}}^k_{\boldsymbol{W}} := \rho(\mathbf{W}^k...\rho(\mathbf{W}^1\hat{\mathbf{x}}))$ where $\hat{\mathbf{x}} \in \mathbb{B}^\infty_{\mathbf{x}}(\epsilon_x)$.

3.2 Main Theorem and Results

Theorem 1 (all-layer and input joint perturbation) *Let $f_{\boldsymbol{W}}(\boldsymbol{x}) = \boldsymbol{W}^L\rho(..\rho(\boldsymbol{W}^1\boldsymbol{x})...)$ denotes an L-layer neural network and let $f_{\widehat{\boldsymbol{W}}}(\hat{\boldsymbol{x}}) = \hat{\boldsymbol{W}}^L\rho(..\rho(\hat{\boldsymbol{W}}^1\hat{\boldsymbol{x}})...)$ with $\hat{\boldsymbol{W}}^m \in \mathbb{B}^\infty_{\boldsymbol{W}^m}(\epsilon_m), \forall m \in [L]$ and $\hat{\boldsymbol{x}} \in \mathbb{B}^\infty_{\boldsymbol{x}}(\epsilon_x)$, furthermore, let ξ be the set containing possible perturbations, i.e. $\xi := \epsilon_{\boldsymbol{x}} \cup \{\epsilon_m\}_{m=1}^L$ and d_m representing the dimension of matrix $\boldsymbol{W}^m$'s row vector, then for any set of pairwise margin bound between natural and joint perturbed settings, we have*

$$f^{ij}_{\widehat{\boldsymbol{W}}}(\hat{\boldsymbol{x}}) \leq f^{ij}_{\boldsymbol{W}}(\boldsymbol{x}) + \tau^{ij}_{\boldsymbol{W}}(\xi) + \zeta^{ij}_{\boldsymbol{W}}(\boldsymbol{x}, \xi) \tag{1}$$

where $\tau^{ij}_{\boldsymbol{W}}(\xi)$ can be expressed as

$$\tau^{ij}_{\boldsymbol{W}}(\xi) = \epsilon_{\boldsymbol{x}}\left(\left\|W^L_{i,:} - W^L_{j,:}\right\|_1 + 2d_L\epsilon_L\right)\Pi^{L-1}_{m=1}(\|\boldsymbol{W}^m\|_\infty + d_m\epsilon_m) \tag{2}$$

while $\zeta^{ij}_{\boldsymbol{W}}(\boldsymbol{x}, \xi)$ possesses the following form

$$\begin{aligned}
\zeta^{ij}_{\boldsymbol{W}}(\boldsymbol{x}, \xi) := \left\|W^L_{i,:} - W^L_{j,:}\right\|_1 \Bigg\{ & \epsilon_1 \|\boldsymbol{x}\|_1 \Pi^{L-2}_{l=1} \left\|(\boldsymbol{W}^{L-l})\right\|_\infty \\
& + \sum_{k=1}^{L-3} \left(\Pi^{L-1}_{m=k+2} \|\boldsymbol{W}^m\|_\infty\right)\epsilon_{k+1} \left\|\boldsymbol{h}^{k^*}\right\|_1 + \epsilon_{L-1}\left\|\boldsymbol{h}^{L-2^*}\right\|_1 \Bigg\} + 2\epsilon_L \left\|\boldsymbol{h}^{L-1^*}\right\|_1 \\
& \textit{where } \boldsymbol{h}^{k^*} = \rho(\boldsymbol{W}^{k^*}...\rho(\boldsymbol{W}^{1^*}\boldsymbol{x}) \\
& \textit{with } \begin{cases} W^{m^*}_{i,j} = W^m_{i,j} + \epsilon_m, \ \forall i,j \textit{ and } \forall m \in [L] \setminus \{1\} \\ W^{1^*}_{i,j} = W^1_{i,j} + sgn([\boldsymbol{x}]_j)\,\epsilon_1, \ \forall i,j \end{cases}
\end{aligned} \tag{3}$$

Proof: Please see Appendix A.1

One could inspect that in equation (1), the term $\zeta^{ij}_{\boldsymbol{W}}(\mathbf{x}, \xi)$ stands for the worst-case error when we consider weight perturbation. Secondly, the second term $\tau^{ij}_{\boldsymbol{W}}(\xi)$ represents the error induced by input perturbation when weight perturbation is taken into presumptions of the models.

3.3 Theory-inspired loss towards non-singular robustness

With our theoretical insights on margin bound, we proceed to construct a new regularization function towards training a non-singular adversarial robust neural network. Specifically, consider the new loss function in the following form:

$$\ell'(f_{\boldsymbol{W}}(\mathbf{x}), y) = \ell_{\text{cls}}(f_{\boldsymbol{W}}(\mathbf{x}), y) + \alpha \max_{y' \neq y}\{\tau^{y'y}_{\boldsymbol{W}}(\xi)\} + \beta \max_{y' \neq y}\{\zeta^{y'y}_{\boldsymbol{W}}(\mathbf{x}, \xi)\} \tag{4}$$

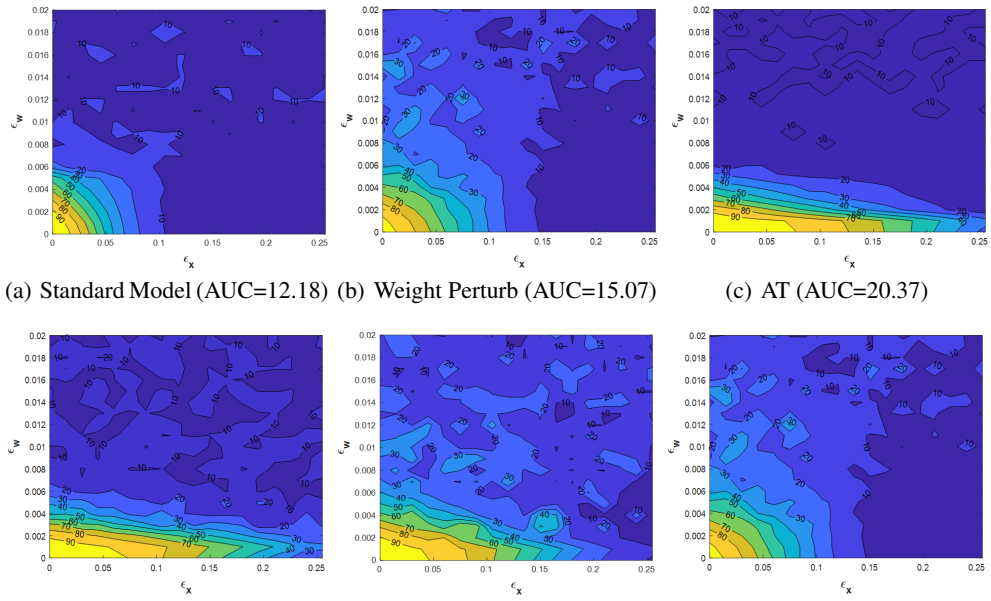
In the above equation, the classification loss function is accompanied with two extra regularizers which can improve model robustness in both input and parameter spaces.

4 Experiments

In this section, we conduct experiments to demonstrate the performance of singular and non-singular models. The detail of experiment setup sees in Appendix A.2. For comparison, five different training methods using the training loss in (4) are presented in our experiments: (i) Standard Model, (ii) Weight Perturb, (iii) Adversarial Training (AT) Madry et al. (2018), (iv) Adversarial Training with additional β-term regularizaiton (AT+β), and (v) Joint Input-Weight Perturb (JIWP).

4.1 Performance Evaluation

For non-singular robustness evaluation, we generalize the projected gradient descent (PGD) attack Madry et al. (2018) for input perturbation to joint input-weight perturbation, by simultaneously computing the signed gradient of the CE loss with respect to the data input and the model weight, clipping the perturbation within their respective ℓ_∞ ball constraints, and iterate this process for 100 steps with step sizes $\alpha_{\boldsymbol{X}} = 0.01$ and $\alpha_{\boldsymbol{W}} = 0.0005$. We describe this joint PGD attack as



(a) Standard Model (AUC=12.18) (b) Weight Perturb (AUC=15.07) (c) AT (AUC=20.37)

(d) AT+β-1 (AUC=21.51) (e) AT+β-2 (AUC=25.23) (f) JIWP (AUC=17.74)

Figure 1: Comparison of test accuracy contour of neural networks under joint input-weight PGD attack (100 steps) with varying input (ϵ_x) and weight (ϵ_w) perturbation levels. AUC refers to the area under curve scores. Comparing to the the standard model (a), singular robust models (b) and (c) have comparable or even worse robustness under their respective untrained perturbation type. Non-singular robust models using our proposed regularization function, including (d), (e) and (f), show significantly better AUC scores.

follows. Given an input $\boldsymbol{X}$ and a trained neural network weight $\boldsymbol{W}$, the perturbed weight $\widetilde{\boldsymbol{W}}$ and input $\widetilde{\boldsymbol{X}}$ are crafted by iterative gradient ascent using the sign of gradient of the CE loss marked as $\text{sgn}(\nabla_{\boldsymbol{W},\boldsymbol{X}}\ell_{cls}(f_{\widetilde{\boldsymbol{W}}}(\widetilde{\boldsymbol{X}}),y))$. The attack iteration with step sizes $\alpha_{\boldsymbol{W}}$ of weight and $\alpha_{\boldsymbol{X}}$ of input is formalized as $\widetilde{\boldsymbol{W}}^{(0)}=\boldsymbol{W}$, $\widetilde{\boldsymbol{W}}^{(t+1)}=\text{Clip}_{\boldsymbol{W},\epsilon_w}\left\{\widetilde{\boldsymbol{W}}^{(t)}+\alpha_{\boldsymbol{W}}\text{sgn}(\nabla_{\boldsymbol{W},\boldsymbol{X}}\ell_{cls}(f_{\widetilde{\boldsymbol{W}}^{(t)}}(\widetilde{\boldsymbol{X}}^{(t)}),y))\right\}$ and $\widetilde{\boldsymbol{X}}^{(0)}=\boldsymbol{X}$, $\widetilde{\boldsymbol{X}}^{(t+1)}=\text{Clip}_{\boldsymbol{X},\epsilon_x}\left\{\widetilde{\boldsymbol{X}}^{(t)}+\alpha_{\boldsymbol{X}}\ \text{sgn}(\nabla_{\boldsymbol{W},\boldsymbol{X}}\ell_{cls}(f_{\widetilde{\boldsymbol{W}}^{(t)}}(\widetilde{\boldsymbol{X}}^{(t)}),y))\right\}$.

Fig.1 demonstrates the non-singular robustness performance for each model. The standard model (a) is vulnerable to both weight and input perturbations. Singular robust models (b) and (c) are only robust to the seen perturbation type, while they only have comparable or even worse robustness against unseen perturbation type. For example, AT (model (c)) is only trained on input perturbation and is observed to be less robust under weight perturbation compared to the standard model (a). Similarly, the robustness of weight perturb model (b) to input perturbation is only slightly better than the standard model. The results suggest the insufficiency of singular robustness analysis. Comparing the area under curve (AUC) score of test accuracy, non-singular robust models (bottom row, (d)-(f)) using our proposed loss significantly outperform standard and singular robust models (top row). The AUC of best AT+β model (e) improves that of AT by about 24%, validating the effectiveness of our proposed regularizer. AT+β also attains better AUC than JIWP, suggesting that min-max training is crucial to non-singular robustness.

5 Conclusion

In this paper, we analyze the robustness of pairwise class margin for neural networks against joint input-weight perturbations. A theory-inspired regularizer is proposed towards training comprehensive robust neural networks. Empirical results against joint input-weight perturbations show that singular robust models can give a false sense of overall robustness, while our proposal can significantly improve non-singular adversarial robustness and offer thorough evaluation.

REFERENCES

Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84:317–331, 2018.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, pp. 39–57, 2017.

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 *ACM Workshop on Artificial Intelligence and Security*, pp. 15–26, 2017.

Pin-Yu Chen, Yash Sharma, Huan Zhang, Jinfeng Yi, and Cho-Jui Hsieh. EAD: elastic-net attacks to deep neural networks via adversarial examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 10–17, 2018.

Nicholas Cheney, Martin Schrimpf, and Gabriel Kreiman. On the robustness of convolutional neural networks to internal architecture and weight perturbations. *arXiv preprint arXiv:1703.08245*, 2017.

Minhao Cheng, Thong Le, Pin-Yu Chen, Jinfeng Yi, Huan Zhang, and Cho-Jui Hsieh. Query-efficient hard-label black-box attack: An optimization-based approach. *International Conference on Learning Representations*, 2019.

Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. The robustness of deep networks: A geometrical perspective. *IEEE Signal Processing Magazine*, 34(6):50–62, 2017.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Matthias Hein and Maksym Andriushchenko. Formal guarantees on the robustness of a classifier against adversarial manipulation. In *Advances in Neural Information Processing Systems*, pp. 2263–2273, 2017.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *International Conference on Learning Representations*, 2017.

Yannan Liu, Lingxiao Wei, Bo Luo, and Qiang Xu. Fault injection attack on deep neural network. In *2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 131–138. IEEE, 2017.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *International Conference on Learning Representations*, 2018.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582, 2016.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *International Conference on Learning Representations*, 2014.

Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, and Shin-Ming Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 742–749, 2019.

Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. On the convergence and robustness of adversarial training. In *ICML*, volume 1, pp. 2, 2019.

Tsui-Wei Weng, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, Dong Su, Yupeng Gao, Cho-Jui Hsieh, and Luca Daniel. Evaluating the robustness of neural networks: An extreme value theory approach. *International Conference on Learning Representations*, 2018.

Tsui-Wei Weng, Pu Zhao, Sijia Liu, Pin-Yu Chen, Xue Lin, and Luca Daniel. Towards certificated model robustness against weight perturbations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 6356–6363, 2020.

Bernard Widrow and Michael A Lehr. 30 years of adaptive neural networks: perceptron, madaline, and backpropagation. *Proceedings of the IEEE*, 78(9):1415–1442, 1990.

Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33, 2020.

Kaidi Xu, Sijia Liu, Pu Zhao, Pin-Yu Chen, Huan Zhang, Quanfu Fan, Deniz Erdogmus, Yanzhi Wang, and Xue Lin. Structured adversarial attack: Towards general implementation and better interpretability. *International Conference on Learning Representations*, 2019.

Pu Zhao, Siyue Wang, Cheng Gongye, Yanzhi Wang, Yunsi Fei, and Xue Lin. Fault sneaking attack: A stealthy framework for misleading deep neural networks. In *2019 56th ACM/IEEE Design Automation Conference (DAC)*, pp. 1–6. IEEE, 2019.

Pu Zhao, Pin-Yu Chen, Payel Das, Karthikeyan Natesan Ramamurthy, and Xue Lin. Bridging mode connectivity in loss landscapes and adversarial robustness. In *International Conference on Learning Representations*, 2020.