This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Low Curvature Activations Reduce Overfitting in Adversarial Training

Vasu Singla Sahil Singla Soheil Feizi David Jacobs
University of Maryland
{vsingla, ssingla, sfeizi, djacobs}@cs.umd.edu

Abstract

Adversarial training is one of the most effective defenses against adversarial attacks. Previous works suggest that overfitting is a dominant phenomenon in adversarial training leading to a large generalization gap between test and train accuracy in neural networks. In this work, we show that the observed generalization gap is closely related to the choice of the activation function. In particular, we show that using activation functions with low (exact or approximate) curvature values has a regularization effect that significantly reduces both the standard and robust generalization gaps in adversarial training. We observe this effect for both differentiable/smooth activations such as SiLU as well as non-differentiable/non-smooth activations such as LeakyReLU. In the latter case, the "approximate" curvature of the activation is low. Finally, we show that for activation functions with low curvature, the double descent phenomenon for adversarially trained models does not occur.

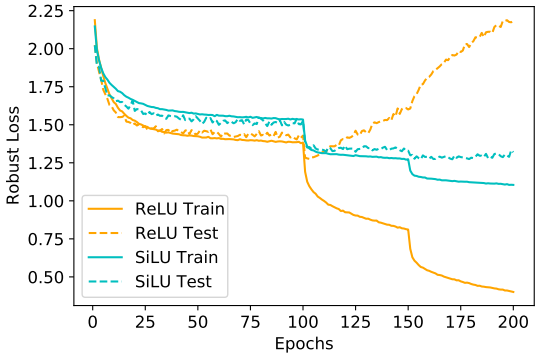

Figure 1: Learning curves for a robustly trained ResNet-18 model on CIFAR-10. Using an activation function with low curvature such as SiLU prevents robust overfitting, achieving and maintaining low test robust loss, even compared to the best early-stop checkpoint of a network with ReLU activation function. The learning rate is decreased by a factor of 10 at the 100th and 150th epoch.

1. Introduction

Deep Neural Networks can be readily fooled by adversarial examples, which are computed by imposing small perturbations on clean inputs [65]. Adversarial attacks have been well studied in the machine learning community in recent years [10, 43, 23, 54, 21, 36, 37]. There have been several defenses proposed against adversarial attacks in the literature [53, 64, 7]. In our work we focus on adversarial training [43, 23, 35], one of the most effective empirical defenses.

Adversarial training involves training the network on adversarially perturbed data instead of clean data to produce a classifier with better robustness on the test set. However, it has been shown that networks produced through vanilla adversarial training do not robustly generalize well [59, 56, 18]. The gap between robust train and test accuracy for adversarially trained neural networks i.e. the *robust generalization gap* can be far greater than the generalization gap achieved during standard empirical risk minimization. In this work, we show that the robust generalization gap is significantly impacted by the curvature of the activation function, and activations with low curvature can act as efficient regularizers for adversarial training, effectively mitigating this phenomenon.

Rice *et al*. [56] showed for adversarially trained ReLU networks, the best robust test accuracy is not achieved by allowing models to train until convergence. Adversarial training has the characteristic that, after a certain point, further training will continue to decrease the robust training loss, while the robust test loss starts increasing. This phenomenon is referred to as *robust overfitting* and ultimately leads to poor robust accuracy on the test set. Rice *et al*. also showed that while traditional approaches against overfitting such as l_1, l_2 regularization can mitigate robust overfitting, no approach works better than simple early stopping. Since standard accuracy continues to improve even after the network overfits to adversarial examples, early stopping leads to trade-off between selecting a model with high robust accuracy versus a model with high standard accuracy [11].

In this work, we systematically study the impact of activation functions on generalization. We first theoretically analyze the relation between maximum curvature of the activation function and adversarial robustness. A key observation of our paper is that for smooth activation functions the maximum value of the second derivative of the function, i.e. the maximum curvature has a significant impact on robust generalization. Specifically by using activations with low curvature the robust generalization gap can be reduced, whereas with high curvature the robust generalization gap increases. For instance, in Figure 1 for an adversarially trained CIFAR-10 model, test error on adversarial examples for the ReLU activation function decreases after the first learning rate drop, and keeps increasing afterwards. However, for SiLU [55] a smooth activation function with low curvature, robust test loss keeps decreasing. We also show that the choice of activation has a similar effect on the standard generalization gap. In other words, activations that show a large robust generalization gap also have a large standard generalization gap, and vice versa. Our work therefore provides novel insights to the robust overfitting phenomenon. The main objective of our work is to understand the relation between curvature of the activation function and adversarial training, and highlight findings which can be useful for training adversarially robust models.

Xie *et al*. [72] showed that replacing ReLU, a widely used activation function, by "smooth" [1] activation functions such as Softplus or SiLU with a weak adversary (single step PGD), improves adversarial robustness on Imagenet [14] for "free". They posit smooth activations allow adversarial training to find harder adversarial examples and compute better gradient updates to weight parameters. Further works have however demonstrated that while smooth activation functions can positively affect clean and robust accuracy, the trend is not as clear as the one observed by Xie *et al*. Thus, ReLU networks remains a prominent choice for robust classification [26, 51].

In contrast to Xie *et al*. [72], we consider a strong adversary for training and show that smoothness of activations is not required to obtain a regularization effect on adversarial training. In our experiments, we show that the same regularization can be achieved using non-smooth activations with low "approximate"" curvature. For non-smooth activations however, curvature is not well-defined. We consider LeakyReLU which is a non-smooth activation function and use the difference of activation slopes in positive and negative domains as the approximate maximum curvature of the activation function. Even for such a non-smooth activation function, we observe that if the approximate curvature is low, the robust overfitting phenomenon does not occur. Also in contrast to Xie *et al*. [72] we empirically show that smooth activations can perform worse than ReLU, if the smooth activation has high curvature.

Finally, we study the phenomenon of double descent generalization curves seen in standard training [4] and robust training [48]. Double descent describes the following phenomenon. Increasing model complexity causes test accuracy to first increase and then decrease. Then upon reaching a critical point known as interpolation threshold, test accuracy starts increasing again. We show that double descent curves reported by [56] for robust overfitting using ReLU do not hold for activation functions with low curvature such as SiLU.

2. Related Works

Goodfellow *et al*. [23] provided one of the first approaches for adversarial training based on generating adversarial examples through the fast sign gradient method (FGSM). Building on this, a stronger adversary known as the basic iterative method [35] was proposed in subsequent work, using multiple smaller steps for generating adversarial examples. Madry *et al*. [43] extended this adversary with multiple random restarts to train models on adversarial data, referred to as projected gradient descent (PGD) adversarial training. Further works have focused on improving the performance of the adversarial training procedure with methods such as feature denoising [74], hypersphere embedding [52], balancing standard and robust error [80] and using friendly adversarial data [81]. A separate line of works has focused on speeding up adversarial training due to its increased time complexity, by reducing attack iterations and computational complexity for calculating gradients [78, 60, 69]. Another tangent line of work focuses on adversarial training for universal attacks [61, 5].

Besides adversarial training, several other defenses have been proposed such as defensive distillation [53], preprocessing techniques [27, 64, 7] and randomized transformations [73, 16, 41] or detection of adversarial examples [44, 19]. However these methods were later broken by stronger adversaries [3, 67, 9]. These defense methods were shown to rely on obfuscated gradients (gradient masking), which provided a false sense of security. Due to the bitter history of gradient masking as a defense, Xie *et al*. [72] proposed use of smooth activations with a single step PGD attack, reaching state of the art robust performance on ImageNet [14]. Xie *et al*. hypothesize that using smooth activations provides networks with better gradient updates and allows adversaries to find harder examples.

Since many defenses proposed in the literature have been broken, another separate line of work has focused on certified defenses, which can guarantee robustness against adversarial attacks. These methods use techniques such as mixed-integer programming methods [66, 42, 20, 8] and satisfiability modulo theories [33, 17, 31]. Some certifi-

[1] We use the same definition of smoothness as Xie *et al*., that the function is C^1 smooth, that is, that the first derivative is continuous everywhere.

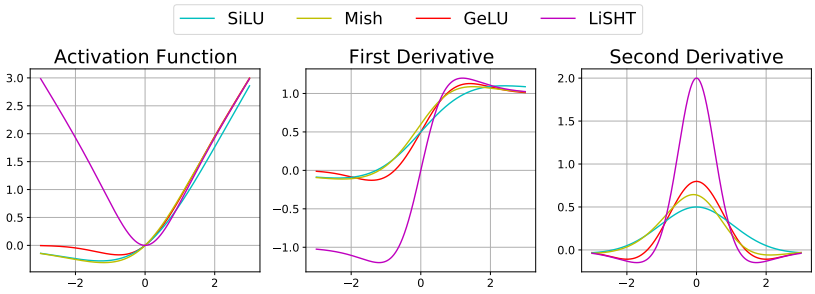

Figure 2: Activation functions along with their first and second derivatives.

cation methods bound the global Lipschitz constant of the network, which are usually loose for large neural networks with multiple layers [2, 24]. Another line of work has focused on providing loose certificates using other techniques such as randomized smoothing [13, 38, 40, 1], abstract representations [22, 45, 62], interval bound propagation, second-order information [63], [25] and duality and linear programs [58, 68, 70].

Lack of overfitting in overparameterized deep learning models is an intriguing phenomenon for deep learning [77]. These models can be trained to effectively zero training error, without having impact on test time performance. Hence, it is now standard practice in deep learning to train longer and use large overparameterized models, since test accuracy generally improves past an interpolation point also known as double descent generalization [4, 48]. Schmidt *et al*. [59] however have shown that sample complexity required for adversarially robust generalization is significantly higher than sample complexity for standard generalization. In a recent work, Rice *et al*. [56] have shown the overfitting phenomenon to be dominant in adversarial training and show that training longer decrease robustness on test data. Rice *et al*. also show that double descent generalization curves seem to hold with increase in model size but not by training longer. A recent work shows that robust overfitting may be mitigated [12] using a combination of previously proposed techniques such as knowledge-distillation [75] and stochastic weight averaging [32]. Another recent work, proposes the use of adversarial weight perturbations [71] to mitigate robust overfitting, which may also increase the training time. AVMixup [39] also discussed the idea of robust overfitting and proposed a combination of AVMixup, Label smoothing and Feature Scatter to alleviate robust overfitting on CIFAR-10. In contrast to these works, we discover a novel way to mitigate this phenomenon without using complex regularization techniques that may lead to additional hyper-parameters and increased training time; we only modify the activation function of the network.

3. Background

3.1. Adversarial Training

To train networks that are robust to adversarial examples, the following robust optimization framework is used:

$$\min_{w} \mathbb{E}_{(x,y)\sim\mathbb{D}} \left[\max_{d(x,\hat{x})\leq\epsilon} l(f_w(\hat{x}), y) \right]$$

where x is a training sample with ground truth label y sampled from the underlying data distribution $\mathbb{D}$, $l(.,.)$ is the loss function, f_w is the model parameterized by w parameters, $d(.,.)$ is a distance function and ϵ is the maximum distance allowed. Typically, the distance function is chosen to be an l_p-norm ball such as the l_2 and l_∞-norm balls though other non l_p threat models have been considered in [36, 37]. Adversarial training thus consists of two optimization problems, the inner maximization problem to construct adversarial samples, and the outer minimization problem to update weight parameters w. To solve the inner maximization problem, different types of attacks have been used in the literature, such as projected gradient descent (PGD) [43] or fast gradient sign method (FSGM) [69]. For example, an l_∞ PGD adversary starts with a random initial perturbation drawn from a uniform distribution $\mathcal{U}$, and iteratively adjusts the perturbation with α step-size towards the l_∞ gradient direction, followed by projection back onto the l_∞ norm ball with maximum radius ϵ:

$$\begin{aligned} \hat{x}_0 &= x + \mathcal{U}(-\epsilon, \epsilon) \\ \bar{x}_t &= \hat{x}_t + \alpha \cdot \text{sign}\big(\nabla_{\hat{x}_t} l(f(\hat{x}_t), y)\big) \\ \hat{x}_{t+1} &= \max(\min(\bar{x}_t, x+\epsilon), x-\epsilon) \end{aligned}$$

3.2. Robust Overfitting

A surprising characteristic of overparametrized models is their good generalization behavior observed in practice. [4, 50]. Although overparameterized models have enough model complexity to memorize the dataset even on random labels [77], they can be trained to zero error on the training

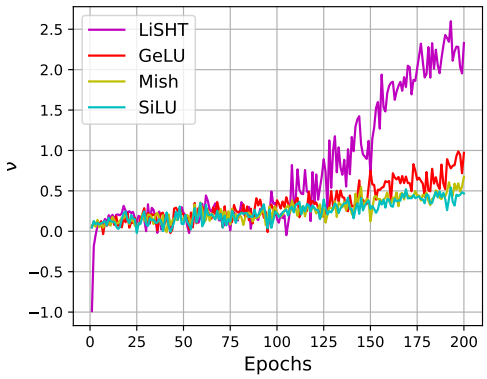

Figure 3: Maximum eigenvalues for a batch of test examples for Resnet-18 models with different smooth activations. Eigenvalues are larger for activations with high curvature.

set with no detrimental effects on generalization. For the standard (non-adversarial) empirical risk minimization setting, modern convergence curves indicate that while training for long periods of time, test loss continues to decrease [48]. This has led to the practice of training models for as long as possible to achieve better generalization [30]. However, in adversarial training it was noted that training longer can cause overfitting and result in worse test performance [56]. This phenomenon has been referred to as "robust overfitting" and shown to occur with a variety of datasets, model architectures and different threat models.

Regularizers are standard tools in practice to mitigate the effects of overfitting, especially in the regime when the number of parameters are larger than the number of data points. The standard regularization techniques such as l_1 and l_2 regularization and data augmentation methods such as Cutout [15] and Mixup [79] have been shown to be ineffective against robust overfitting phenomena [56]. Only early-stopping using a validation dataset and semi-supervised learning methods that augment the dataset with unlabelled data have been shown to be effective and reduce the generalization gap for adversarially robust learning. Data augmentation using semi-supervised methods however requires the use of additional data that may not be available. Early stopping leads to selection of an earlier checkpoint and causes a trade off between robust accuracy and standard accuracy, as training longer leads to better standard test accuracy.

4. Impact of Activation Curvature on Adversarial Training

In this section we consider the effects of curvature for smooth activation functions on standard and robust generalization gaps. We define curvature for smooth activation functions by the maximum of the second derivative[2] i.e $\max_x f''(x)$. We consider the following smooth activation functions, which are ranked by decreasing curvature as follows (see Figure 2 for functions and their first and second derivatives):

1. **Linearly Scaled Hyperbolic Tangent** (LiSHT) [57]: $f(x) = x * \tanh(x)$, this function has highest curvature among activations considered.

2. **Gaussian Error Linear Unit** (GeLU) [29]: $f(x) = x * \Phi(x)$, where $\Phi(x)$ is gaussian cummulative distribution function.

3. **Mish** [46]: $f(x) = x * \tanh(ln(1+\exp(x)))$ is a smooth continuous function similar to SiLU.

4. **SiLU** [55]: $f(x) = x * sigmoid(x)$ is a smooth approximation to ReLU but has a non-monotonic "bump" for $x < 0$.

We also conduct experiments for non-smooth ReLU activation as a baseline. Code for reproducing our experiments can be found at https://github.com/vasusingla/low_curvature_activations.

4.1. Analyzing the influence of activations on robustness

In this section, we analyze the theoretical relationship between curvature of the activation function and adversarial robustness. The motivation behind our analysis is to provide an intuition for our observations, we do not rigorously prove a monotonic relationship between robustness and activation curvature. To elucidate this, we first consider the relation between the input Hessian (ie., the second derivatives of the output with respect to the input) and adversarial robustness. We consider a simple binary classifier f, implemented as a two-layer neural network. Let w_1, w_2 be weight matrices for the first and second layers respectively. Let $\sigma(\cdot)$ be a twice differentiable activation function and $\sigma''(\cdot)$ denote the second derivative of the activation function. The two layer neural network can then be represented as $f(x) = w_2^T \sigma(w_1 x)$. Assume the final layer of the network outputs a single logit, which is transformed to probability using a sigmoid function. In other words, the probability of a sample being in class 0 is given as $p(x) = sigmoid(f(x))$. Assuming a sample is classified into class 1 if $p(x) < 0.5$, then a sample x is classified into class 1 iff $f(x) < 0$ and class 0 otherwise. In other words, we use a probability threshold of 0.5, to classify an example into class 1. We assume that the neural network can be locally well approximated using the second order Taylor expansion. We now use the results by [47] about the relation

[2]Note that this definition of curvature is different from the standard definition of curvature used for twice-differentiable functions.

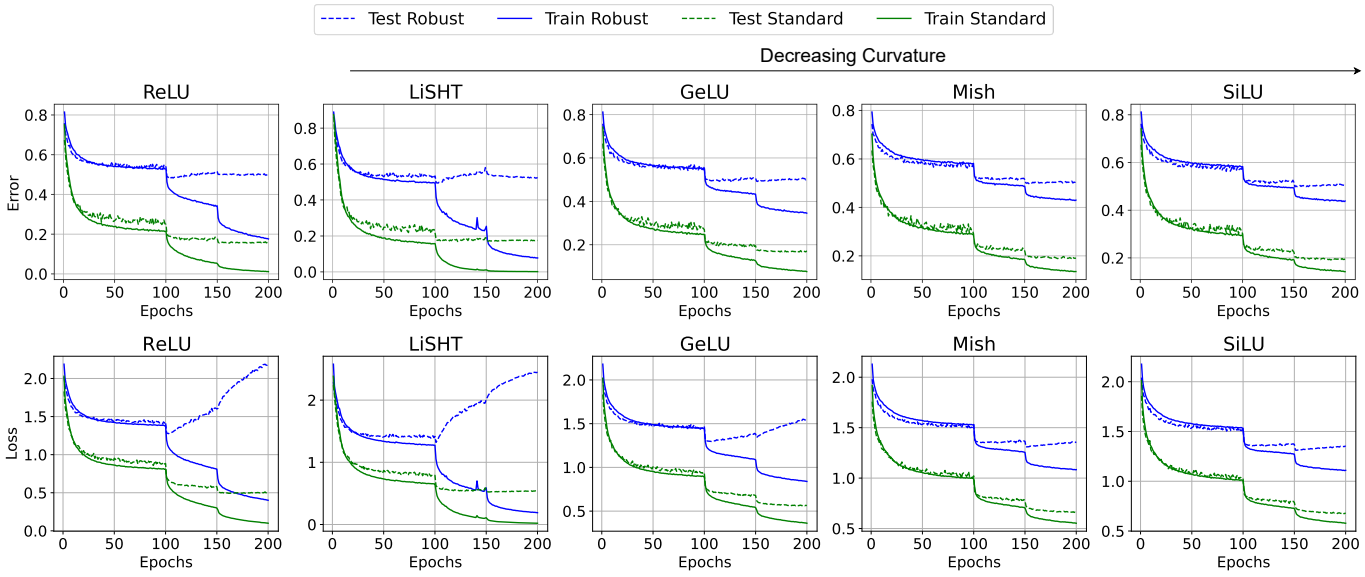

Figure 4: Learning curves for CIFAR-10 dataset on Resnet-18 for different activation functions. ReLU activation is non-smooth and included as a baseline, all the other activations are ordered by decreasing curvature from left to right. Top graphs show standard and robust error, and bottom graphs represent loss curves for both train and test data.

between the input Hessian and robustness. Let x belong to class 1, then for $x+\delta$ to be classified as class 0, the minimal l_2 perturbation that fools the classifier can be written as:

$$\delta^* = \arg\min_{\delta} \|\delta\|$$
$$\text{s.t. } f(x) + \nabla_x f(x)^T \delta + \frac{1}{2}\delta^T \nabla_x^2 f(x) \delta \geq 0$$

It can be shown under these assumptions the magnitude of δ^* can be upper and lower bounded with respect to input curvature. We use the following lemma -

Lemma 1. *[47] Let x be such that $c = -f(x) \geq 0$, and let $g = \nabla_x f(x)$. Assume that $\nu = \lambda_{max}\left(\nabla_x^2 f(x)\right) \geq 0$, denotes the largest eigenvalue and let u be the eigenvector corresponding to ν. Then,*

$$\frac{\|g\|}{\nu}\left(\sqrt{1+\frac{2\nu c}{\|g\|^2}}-1\right) \leq \|\delta^*\| \leq \frac{\|g^T u\|}{\nu}\left(\sqrt{1+\frac{2\nu c}{(g^T u)^2}}-1\right) \quad (1)$$

This lemma shows that upper and lower bounds on the magnitude of δ^* increase as ν decreases keeping all other factors constant [47]. An increase in $\|\delta^*\|$ therefore increases the minimum l_2 ball required to find an adversarial example for input x, leading to increased robustness. Therefore, a low maximum eigenvalue of the input Hessian leads to higher adversarial robustness.

We now show the relation between activation functions and input curvature. For the considered two layer neural network, the Hessian with respect to the input x is given as:

$$\nabla_x^2 f(x) = w_1^T diag\Big(\sigma''(w_1 x) \odot w_2\Big) w_1 \quad (2)$$

where $\odot$ denotes the Hadamard product between two vectors. Equation 2 shows that the Hessian of the input directly depends on $\sigma''(.)$, which suggests that an increase in the curvature of the activation function leads to an increase in the norm of the input Hessian. Finally, although we assume our activation to be smooth we expect similar results for non-smooth activations.

We empirically show the relation between ν and activation curvature holds for adversarially trained Resnet-18 models. The learning curves presented in Fig. 3 show that for activations with high curvature, the maximum eigenvalue of the input Hessian indeed is larger. This result combined with our previous observation therefore suggests high activation curvature indeed leads to lower robustness.

4.2. Activation Curvature and Generalization Gap

In this section we show results for the adversarial training for different smooth activation functions. We hypothesize that for adversarial trained networks, activations with low curvature are more robust and have a small generalization gap.

Experimental Settings - We show our results on the CIFAR-10 and CIFAR-100 dataset [34]. For comparison with best early-stop checkpoint [56], we randomly split the

Dataset	Activation	Robust Accuracy				Standard Accuracy			
		Final Train	Final Test	Best Val	Diff.	Final Train	Final Test	Best Val	Diff.
CIFAR-10	LiSHT	92.27	47.21	50.31	**45.06**	99.9	82.53	82.44	**17.37**
	ReLU	82.46	49.25	51.06	**33.21**	98.9	83.73	81.62	**15.17**
	GeLU	65.45	49.31	50.15	**16.14**	92.41	82.81	79.25	**9.6**
	Mish	57	49.18	49.62	**7.82**	86.48	80.05	79.96	**6.43**
	SiLU	56.15	48.91	49.41	**7.24**	85.79	80.55	80.57	**5.24**
CIFAR-100	LiSHT	93.58	18.62	22.48	**74.96**	99.92	49.12	49.13	**50.8**
	ReLU	79.87	18.81	25.91	**61.06**	98.58	51.58	51.05	**47**
	GeLU	57.96	21.56	26.33	**36.4**	89.18	53.67	49.5	**35.51**
	Mish	39.65	24.27	25.88	**15.38**	71.5	53.43	48.37	**18.07**
	SiLU	37.81	24.29	25.82	**13.52**	68.73	52.65	52.18	**16.08**

Table 1: Performance of different activations on CIFAR-10 and CIFAR-100 with ResNet-18. We use the best checkpoint based on **best robust accuracy** on the validation set shown in "Best Val"" column. The generalization gap, i.e difference between final train and final test accuracy is shown in "Diff." column. Generalization gap for both standard and robust accuracy increases for activations with high curvature.

original set into training and validation set with 90% and 10% of the images respectively. We consider the l_∞ threat model and use PGD-10 step attack with a single restart for training and PGD-20 step attack with 5 restarts for reporting the test accuracy. For the attack hyper-parameters, we use $\epsilon = 8/255$ and $\alpha = 2/255$. We use the ResNet-18 [28] architecture for all our experiments except for experiments with double descent curves where we use Wide ResNet-28 [76]. We use the same training setup as [56] throughout the paper, an SGD optimizer with momentum of 0.9 and weight decay 5×10^{-4} for 200 epochs with batch size of 128.

We discover that choice of activation function has a large impact on robust overfitting. Figure 4 shows our results. First we reproduce the effect of robust overfitting observed by Rice *et al*. [56] for all the activations. The robust training loss keeps decreasing, however robust test loss rises shortly after the first learning rate drop. For standard training and standard test loss however, both keep decreasing throughout training. Training appears to proceed smoothly at the start, however at the learning rate drop on the 100th and 150th epochs, robust test error decreases briefly and then keeps increasing as training progresses. This phenomenon shows the best performance for robust test accuracy is **not** achieved by training till convergence, unlike standard training. In contrast the best standard accuracy for adversarial training is still reached by training till convergence. We show that for activation functions with lower curvature the robust overfitting phenomenon occurs to a lesser degree. In contrast to Xie *et al*. [72], we also show that LiSHT a smooth activation function performs worse than the non-smooth ReLU function and shows a larger robust generalization gap as shown in Fig. 4. We also note that *for activations that display a large robust generalization gap, the standard generalization gap is also higher*. Finally, the curvature of the activation function has a direct impact on both the robust and standard generalization gaps, as shown in the learning curves. For activations with high curvature such as LiSHT and GeLU the generalization gap is large and for activations with low curvature such as Mish and SiLU the generalization gap is much lower. Note that although the training loss/error is higher for activations with lower curvature, adversarial training is much more stable and allows training till convergence, achieving better standard accuracy and maintaining similar robust accuracy.

We show the quantitative results in Table 1. To show the gap due to robust overfitting (decay in performance from peak robust accuracy) we also show the best robust accuracy found using early stopping with a validation set. We also report the corresponding standard accuracy for the **best robust accuracy checkpoint** (not the best standard accuracy checkpoint). The robust and standard generalization gap decrease for CIFAR-10 and CIFAR-100 as seen in Table 1. The effects of robust overfitting, (i.e difference in best and final checkpoint on robust accuracy) also decreases for activations with smaller curvature. For example, the overfitting gap falls from 3.1% for LiSHT to 0.5% for SiLU on CIFAR-10. Standard accuracy however, either remains the same or improves by training longer (compared to the best checkpoint). On CIFAR-100 upon training till convergence, *SiLU simultaneously achieves both robust and standard accuracy higher than ReLU*. These results therefore validate our claim that low curvature activations reduce robust overfitting. Using the best validation checkpoint for CIFAR-100, *SiLU achieves nearly the same robust accuracy and higher standard accuracy than ReLU*. The results therefore show that for adversarial training, curvature of the activation function play an important role in obtaining high robust and standard accuracy.

4.3. Curvature effects with Parameteric Swish

To further understand the impact of activation curvature on standard and robust generalization gap, we conduct anal-

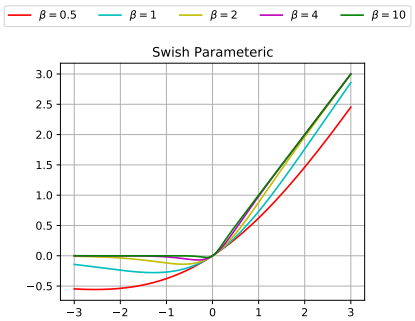

Figure 5: Visualization of PSwish with different β values.

β	Robust Accuracy			Standard Accuracy		
	Train	Test	Diff.	Train	Test	Diff.
0.5	47.00	45.24	**1.76**	75.39	73.57	**1.82**
1	56.15	48.91	**7.31**	85.79	80.55	**5.24**
2	69.65	49.6	**20.05**	94.57	83.39	**11.18**
4	83	49.92	**33.08**	98.82	84.48	**14.34**
10	89.2	50.63	**38.57**	99.7	83.57	**16.13**

Table 2: Performance of PSwish with different β values, higher β value indicates higher curvature. Results are shown for final checkpoint and show that for activations with high curvature, standard and robust generalization gap increases.

ysis with *Parameteric Swish (PSwish)* [6], defined as follows:

$$f(x) = x \cdot sigmoid(\beta x)$$

The SiLU function defined previously is a special case of PSwish, when $\beta = 1$. PSwish transitions from the identity function for $\beta = 0$, to ReLU for $\beta \to \infty$. The curvature of PSwish increases as β increases. Figure 5 shows the PSwish activation function for different values of β.

We show the results with the CIFAR-10 dataset, for final checkpoints for training and testing set in Table 2. Interestingly, we observe that both the standard and robust generalization gap are extremely dependent on the choice of β. The robust generalization gap increases from 1.76 to 38.57 and the standard generalization gap increases from 1.82 to 16.13 for $\beta = 0.5$ and $\beta = 10$ respectively. We also observe that robust test accuracy for the final checkpoint increases from 45.24 to 50.63 for the same β values. For larger values of β i.e $\beta \to \infty$, PSwish behaves like ReLU and standard and robust final test accuracy start decreasing. The results are consistent with our previous experiments and show that the standard and robust generalization gap increases for activations with high curvature. Further using the early stopping checkpoint with the validation set, PSwish with $\beta = 10$ outperforms ReLU baseline by 0.33% on robust accuracy and 1.24% on standard accuracy, highlighting that the choice of activation function can improve standard and robust performance for adversarially trained models.

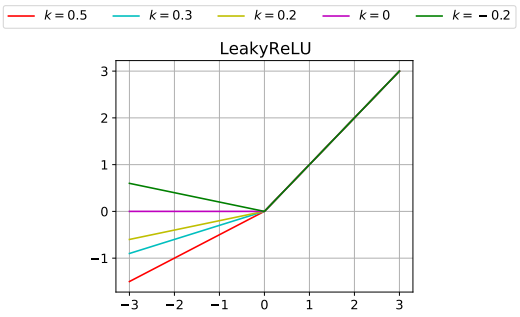

Figure 6: Visualization of LeakyReLU with different k values.

k	Robust Accuracy			Standard Accuracy		
	Train	Test	Diff.	Train	Test	Diff.
0.5	52.74	48.71	**4.03**	82.99	79.56	**3.43**
0.3	63.06	49.62	**13.44**	92.00	83.56	**8.44**
0.2	69.64	49.34	**20.3**	95.37	84.21	**11.16**
0	82.46	49.47	**32.99**	98.9	83.73	**15.17**
-0.2	85.89	48.16	**37.73**	99.47	83.01	**16.46**

Table 3: Performance of the LeakyReLU activation function with different slope values. The standard and robust generalization gap increases for slopes with larger approximate curvature.

5. Does smoothness matter?

Xie *et al*. [72] showed that using smooth activations, adversarial training can achieve better standard and robust accuracy on Imagenet [14]. They posit that using smooth activations can improve gradients, which can both strengthen the attacker and provide better gradient updates to weight parameters, thus achieving superior performance.

In contrast, we show that the relation of the generalization gap to activations can be observed for non-smooth activations as well. We use the non-smooth LeakyReLU activation function defined as follows:

$$\text{LeakyReLU}(k, x) = \begin{cases} x & \text{if } x \geq 0 \\ kx & \text{if } x < 0 \end{cases}$$

where k is a hyper-parameter that can be tuned. The first derivative of LeakyReLU is given as:

$$\frac{d}{dx}\text{LeakyReLU}(k, x) = \begin{cases} 1 & \text{if } x \geq 0 \\ k & \text{if } x < 0 \end{cases}$$

For non-smooth activations, curvature of the activation function however is not well defined. Therefore for LeakyReLU, we use the difference of slopes, i.e $|1 - k|$ as the "approximate" curvature of the function. Hence, for $k \leq 1$ the approximate curvature decreases with increasing value of k . We use the same setup as in previous experiments and show the results for final training and test

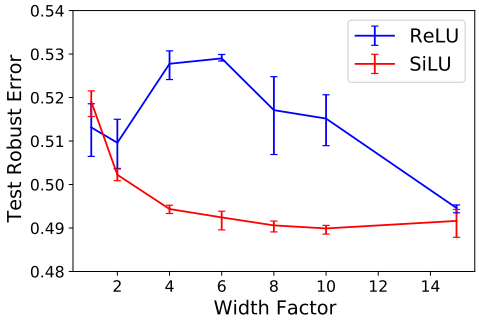

Figure 7: The generalization curves show the double descent phenomenon occurs for networks with ReLU activation but does not occur with SiLU activation. We use adversarially trained WideResnet models and the model complexity is controlled by the width of the architecture. Each data point shows the average for last 3 epochs.

checkpoints in Table 3 on CIFAR-10. We observe behavior similar to smooth activations for LeakyReLU. For $k = 0.5$, the approximate curvature is low, and both robust and standard generalization gap, 4.03 and 3.43 respectively is much smaller than for $k = -0.2$, for which robust and standard generalization gap, 37.73 and 16.46 is large. We therefore hypothesize for non-smooth activations, the "approximate" curvature of the activation function has impact on the generalization gap.

6. Double descent curves

The standard bias-variance trade-off from classical machine learning theory fails to explain why deep networks generalize well especially when they have far more parameters than the samples they are trained on [77]. Its now standard practice to use overparameterized models and allow models to train longer [30] since test time performance typically improves for increased model complexity beyond the data interpolation point, a phenomenon known as *double descent* [4]. It was further shown that both training longer and increasing architecture size can be viewed as increase in model complexity and the double descent phenomenon is observed for both settings [48]. The phenomenon of double descent generalization with increase in model width was also briefly noted for l_2 adversarially trained models [48].

Rice *et al*. [56] show that robust overfitting contradicts the double descent phenomenon observed with respect to training longer, since training longer harms test time performance. Although, they still observe the double descent phenomenon for ReLU networks with respect to the model size as shown in Fig. 7. They therefore posit that, training longer and increasing model size have separate effects on robust generalization.

A recent work [49] suggests that the double descent phenomenon can be mitigated by optimal regularization. We explore whether activations with low curvature can mitigate double descent, by adversarially training Wide Resnets with different width factors. We show results for ReLU and SiLU activation functions in Figure 7. Experiments with other activations could not be conducted due to the high expense of training Wide Resnets. We use the SiLU activation function, because it has lowest curvature among all the activations considered. In Figure 7, we show the results for ReLU and the SiLU activation function with a PGD-10 adversary. While the double descent phenomenon is observed for ReLU activation, robust test performance continues to decrease for the SiLU activation function. Note that SiLU with *width-factor 4 attains equivalent performance to ReLU with width-factor 15*. None-the-less the final test error achieved by ReLU networks with large width factor is equivalent to the lowest test error achieved by SiLU networks with the same width. This suggest that low curvature activations may not be useful for models with large width. The results also indicate that use of activations with small curvature can act as a regularizer to mitigate the double descent phenomenon.

7. Conclusion

In this work, we first use both theoretical and empirical approaches to show the impact of curvature of the activation function on robustness. We further show that this property of regularization further extends to non-smooth activations as well. While results from Rice *et al*. show that classical regularization techniques are unable to prevent robust overfitting, our results show that activation functions with low curvature can largely mitigate that. Since robust overfitting is common in adversarial training, the properties of activation functions that we bring to light in this work can be useful for state of the art robust models. Finally our experiments also show that double descent, another phenomenon that has a significant impact on robust generalization, can be mitigated using activations with low curvature.

8. Acknowledgments

This project was supported in part by NSF CAREER AWARD 1942230, HR00112090132, HR001119S0026, NIST 60NANB20D134 and ONR GRANT13370299, Quantifying Ensemble Diversity for Robust Machine Learning (QED for RML) program from DARPA and the Guaranteeing AI Robustness Against Deception (GARD) program from DARPA. We are grateful to our colleagues Abhay Yadav, Songwei Ge, and Pedro Sandoval for their valuable inputs on the early draft of this manuscript.

References

[1] Alexander Levine 0001 and Soheil Feizi. (de)randomized smoothing for certifiable defense against patch attacks. 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.

[2] Cem Anil, James Lucas, and Roger Grosse. Sorting out lipschitz function approximation. In *International Conference on Machine Learning*, pages 291–301. PMLR, 2019.

[3] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pages 274–283. PMLR, 2018.

[4] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[5] Philipp Benz, Chaoning Zhang, Adil Karjauv, and In So Kweon. Universal adversarial training with class-wise perturbations. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2021.

[6] Garrett Bingham and Risto Miikkulainen. Discovering parametric activation functions. *CoRR*, abs/2006.03179, 2020.

[7] Jacob Buckman, Aurko Roy, Colin Raffel, and Ian Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In *International Conference on Learning Representations*, 2018.

[8] Rudy Bunel, Ilker Turkaslan, Philip H.S. Torr, Pushmeet Kohli, and M. Pawan Kumar. A unified view of piecewise linear neural network verification. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 4795–4804, Red Hook, NY, USA, 2018. Curran Associates Inc.

[9] 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*, pages 3–14, 2017.

[10] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57, 2017.

[11] Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 699–708, 2020.

[12] Tianlong Chen, Zhenyu Zhang, Sijia Liu, Shiyu Chang, and Zhangyang Wang. Robust overfitting may be mitigated by properly learned smoothening. In *International Conference on Learning Representations*, 2021.

[13] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR, 2019.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[15] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[16] Guneet S Dhillon, Kamyar Azizzadenesheli, Zachary C Lipton, Jeremy Bernstein, Jean Kossaifi, Aran Khanna, and Anima Anandkumar. Stochastic activation pruning for robust adversarial defense. *arXiv preprint arXiv:1803.01442*, 2018.

[17] Rüdiger Ehlers. Formal verification of piece-wise linear feed-forward neural networks. *CoRR*, abs/1705.01320, 2017.

[18] Farzan Farnia, Jesse Zhang, and David Tse. Generalizable adversarial training via spectral normalization. In *International Conference on Learning Representations*, 2019.

[19] Reuben Feinman, Ryan R Curtin, Saurabh Shintre, and Andrew B Gardner. Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*, 2017.

[20] Matteo Fischetti and Jason Jo. Deep neural networks as 0-1 mixed integer linear programs: A feasibility study. *CoRR*, abs/1712.06174, 2017.

[21] Songwei Ge, Vasu Singla, Ronen Basri, and David W. Jacobs. Shift invariance can reduce adversarial robustness. *CoRR*, abs/2103.02695, 2021.

[22] T. Gehr, M. Mirman, D. Drachsler-Cohen, P. Tsankov, S. Chaudhuri, and M. Vechev. Ai2: Safety and robustness certification of neural networks with abstract interpretation. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 3–18, 2018.

[23] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

[24] Henry Gouk, Eibe Frank, Bernhard Pfahringer, and Michael J Cree. Regularisation of neural networks by enforcing lipschitz continuity. *Machine Learning*, 110(2):393–416, 2021.

[25] Sven Gowal, Krishnamurthy Dj Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. Scalable verified training for provably robust image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4842–4851, 2019.

[26] Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020.

[27] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering adversarial images using input transformations. In *International Conference on Learning Representations*, 2018.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[29] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[30] Elad Hoffer, Itay Hubara, and Daniel Soudry. Train longer, generalize better: Closing the generalization gap in large batch training of neural networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 1729–1739, Red Hook, NY, USA, 2017. Curran Associates Inc.

[31] Xiaowei Huang, Marta Kwiatkowska, Sen Wang, and Min Wu. Safety verification of deep neural networks. *CoRR*, abs/1610.06940, 2016.

[32] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

[33] Guy Katz, Clark W. Barrett, David L. Dill, Kyle Julian, and Mykel J. Kochenderfer. Reluplex: An efficient SMT solver for verifying deep neural networks. *CoRR*, abs/1702.01135, 2017.

[34] Alex Krizhevsky. Learning multiple layers of features from tiny images. *University of Toronto*, 05 2012.

[35] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[36] Cassidy Laidlaw and Soheil Feizi. Functional adversarial attacks. 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 10408–10418. Curran Associates, Inc., 2019.

[37] Cassidy Laidlaw, Sahil Singla, and Soheil Feizi. Perceptual adversarial robustness: Defense against unseen threat models. In *ICLR*, 2021.

[38] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 656–672. IEEE, 2019.

[39] Saehyung Lee, Hyungyu Lee, and Sungroh Yoon. Adversarial vertex mixup: Toward better adversarially robust generalization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[40] Alexander Levine, Sahil Singla, and Soheil Feizi. Certifiably robust interpretation in deep learning. *CoRR*, abs/1905.12105, 2019.

[41] Xuanqing Liu, Minhao Cheng, Huan Zhang, and Cho-Jui Hsieh. Towards robust neural networks via random self-ensemble. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 369–385, 2018.

[42] Alessio Lomuscio and Lalit Maganti. An approach to reachability analysis for feed-forward relu neural networks. *CoRR*, abs/1706.07351, 2017.

[43] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[44] Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations. *arXiv preprint arXiv:1702.04267*, 2017.

[45] Matthew Mirman, Timon Gehr, and Martin Vechev. Differentiable abstract interpretation for provably robust neural networks. In *International Conference on Machine Learning*, pages 3578–3586. PMLR, 2018.

[46] Diganta Misra. Mish: A self regularized non-monotonic neural activation function. *CoRR*, abs/1908.08681, 2019.

[47] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard. Robustness via curvature regularization, and vice versa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9078–9086, 2019.

[48] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. In *International Conference on Learning Representations*, 2020.

[49] Preetum Nakkiran, Prayaag Venkat, Sham M. Kakade, and Tengyu Ma. Optimal regularization can mitigate double descent. In *International Conference on Learning Representations*, 2021.

[50] Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nathan Srebro. Exploring generalization in deep learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5949–5958, Red Hook, NY, USA, 2017. Curran Associates Inc.

[51] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. In *International Conference on Learning Representations*, 2021.

[52] Tianyu Pang, Xiao Yang, Yinpeng Dong, Taufik Xu, Jun Zhu, and Hang Su. Boosting adversarial training with hypersphere embedding. In *NeurIPS*, 2020.

[53] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597, Los Alamitos, CA, USA, may 2016. IEEE Computer Society.

[54] Nicolas Papernot, Patrick D. McDaniel, and Ian J. Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *CoRR*, abs/1605.07277, 2016.

[55] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[56] Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pages 8093–8104. PMLR, 2020.

[57] Swalpa Kumar Roy, Suvojit Manna, Shiv Ram Dubey, and Bidyut Baran Chaudhuri. Lisht: Non-parametric linearly scaled hyperbolic tangent activation function for neural networks. *arXiv preprint arXiv:1901.05894*, 2019.

[58] Hadi Salman, Greg Yang, Huan Zhang, Cho-Jui Hsieh, and Pengchuan Zhang. A convex relaxation barrier to tight robustness verification of neural networks. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché

Buc, Emily B. Fox, and Roman Garnett, editors, *NeurIPS*, pages 9832–9842, 2019.

[59] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 5019–5031, 2018.

[60] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! 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. Curran Associates, Inc., 2019.

[61] Ali Shafahi, Mahyar Najibi, Zheng Xu, John Dickerson, Larry S Davis, and Tom Goldstein. Universal adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5636–5643, 2020.

[62] Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin Vechev. An abstract domain for certifying neural networks. *Proc. ACM Program. Lang.*, 3(POPL), Jan. 2019.

[63] Sahil Singla and Soheil Feizi. Second-order provable defenses against adversarial attacks. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8981–8991. PMLR, 13–18 Jul 2020.

[64] Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *International Conference on Learning Representations*, 2018.

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

[66] Vincent Tjeng, Kai Y. Xiao, and Russ Tedrake. Evaluating robustness of neural networks with mixed integer programming. In *International Conference on Learning Representations*, 2019.

[67] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1633–1645. Curran Associates, Inc., 2020.

[68] Eric Wong and J. Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. 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 5283–5292. PMLR, 2018.

[69] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*, 2020.

[70] Eric Wong, Frank R. Schmidt, Jan Hendrik Metzen, and J. Zico Kolter. Scaling provable adversarial defenses. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 8410–8419, Red Hook, NY, USA, 2018. Curran Associates Inc.

[71] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. In *NeurIPS*, 2020.

[72] Cihang Xie, Mingxing Tan, Boqing Gong, Alan L. Yuille, and Quoc V. Le. Smooth adversarial training. *CoRR*, abs/2006.14536, 2020.

[73] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan L. Yuille. Mitigating adversarial effects through randomization. *CoRR*, abs/1711.01991, 2017.

[74] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 501–509, 2019.

[75] Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3903–3911, 2020.

[76] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In Edwin R. Hancock Richard C. Wilson and William A. P. Smith, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 87.1–87.12. BMVA Press, September 2016.

[77] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[78] Dinghuai Zhang, Tianyuan Zhang, Yiping Lu, Zhanxing Zhu, and Bin Dong. You only propagate once: Accelerating adversarial training via maximal principle. *arXiv preprint arXiv:1905.00877*, 2019.

[79] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[80] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, pages 7472–7482. PMLR, 2019.

[81] Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. Attacks which do not kill training make adversarial learning stronger. In *International Conference on Machine Learning*, pages 11278–11287. PMLR, 2020.