Confidence-Calibrated Adversarial Training: Generalizing to Unseen Attacks

David Stutz [1] Matthias Hein [2] Bernt Schiele [1]

Abstract

Adversarial training yields robust models against a specific threat model, e.g., L_∞ adversarial examples. Typically robustness does *not* generalize to previously unseen threat models, e.g., other L_p norms, or larger perturbations. Our **confidence-calibrated adversarial training (CCAT)** tackles this problem by biasing the model towards low confidence predictions on adversarial examples. By allowing to reject examples with low confidence, robustness generalizes beyond the threat model employed during training. CCAT, trained *only* on L_∞ adversarial examples, increases robustness against larger L_∞, L_2, L_1 and L_0 attacks, adversarial frames, distal adversarial examples and corrupted examples and yields better clean accuracy compared to adversarial training. For evaluation, we consider 7 attacks directly attacking CCAT by maximizing confidence, allowing up to 50 restarts and 5000 iterations each.

1. Introduction

Deep networks were shown to be susceptible to adversarial examples (Szegedy et al., 2014): adversarially perturbed examples that cause mis-classification while being nearly “imperceptible”, i.e., close to the original example. Recently, adversarial training (Goodfellow et al., 2015; Madry et al., 2018), i.e., training on adversarial examples, became the de-facto state-of-the-art in obtaining adversarially robust models. However, following Fig. 1, adversarial training is known to “overfit” to the threat model *“seen”* during training, e.g., L_∞ adversarial examples. Thus, robustness does not extrapolate to larger L_∞ perturbations, cf. Fig. 1 (top left), or generalize to *“unseen”* attacks, cf. Fig. 1 (bottom left), e.g., other L_p threat models (Sharma & Chen, 2018; Tramèr & Boneh, 2019; Li et al., 2019; Kang et al., 2019; Maini et al., 2019). We hypothesize this to be a result of enforcing high-confidence predictions on adversarial examples, which makes it difficult to extrapolate beyond the adversarial examples seen during training. Moreover, adversarial training often hurts accuracy, resulting in a trade-off between robustness and accuracy (Tsipras et al., 2019; Stutz et al., 2019; Raghunathan et al., 2019; Zhang et al., 2019).

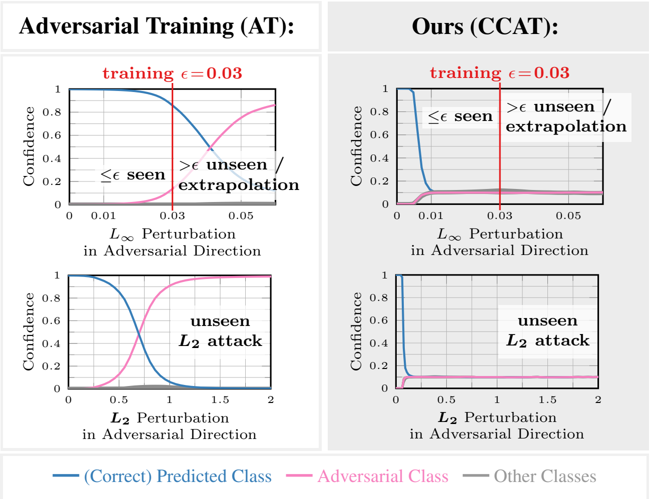

Figure 1: Adversarial Training (AT) versus our CCAT. We plot the confidence in the direction of an adversarial example. AT enforces high confidence predictions for the correct class on the L_∞-ball of radius ϵ (*“seen”* attack during training, top left). As AT enforces no particular bias beyond the ϵ-ball, adversarial examples can be found right beyond this ball. In contrast CCAT enforces a decaying confidence in the correct class up to uniform confidence within the ϵ-ball (top right). Thus CCAT biases the model to extrapolate uniform confidence beyond the ϵ-ball. This behavior also extends to *“unseen”* attacks during training, e.g., L_2 attacks (bottom), such that adversarial examples can be rejected via confidence-thresholding.

Contributions: We propose **confidence-calibrated adversarial training (CCAT)** which trains the network to predict a convex combination of uniform and (correct) one-hot distribution on adversarial examples that becomes more uniform as the distance to the attacked example increases. This is illustrated in Fig. 1. Thus, CCAT implicitly biases the network to predict a uniform distribution beyond the threat model *seen* during training, cf. Fig. 1 (top right). Robustness is obtained by rejecting low-confidence (adversarial) examples through confidence-thresholding. As a result, having

[1] Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken [2] University of Tübingen, Tübingen. Correspondence to: David Stutz <david.stutz@mpi-inf.mpg.de>.

Presented at the ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning. Copyright 2020 by the author(s).

seen *only* L_∞ adversarial examples during training, CCAT improves robustness against previously *unseen* attacks, cf. Fig. 1 (bottom right), e.g., L_2, L_1 and L_0 adversarial examples, larger L_∞ perturbations, adversarial frames (Zajac et al., 2019) or distal adversarial examples (Hein et al., 2019) and accuracy of normal training is preserved better. For each threat model, i.e., L_p with $p \in \{\infty, 2, 1, 0\}$, we adapt 7 different attacks to CCAT and allow up to 50 random restarts and 5000 iterations. We report worst-case robust test error, extended to our confidence-thresholded setting, and compare with standard adversarial training and adversarial training using multiple threat models (Maini et al., 2019).

This paper is a short version of our ICML'20 work. While it is self-contained, we refer to (Stutz et al., 2020) and its supplementary material for further details and results. Our code will be made available at `davidstutz.de/ccat`.

2. Generalizable Robustness by Confidence Calibration of Adversarial Training

We briefly review adversarial training on L_∞ adversarial examples (Madry et al., 2018). Here, robustness does not generalize to larger perturbations or unseen attacks due to enforcing high-confidence predictions on adversarial examples. CCAT addresses this issue with minimal modifications, cf. Alg. 1, by encouraging low-confidence predictions on adversarial examples. During testing, adversarial examples can be rejected by confidence thresholding.

Problems of Adversarial Training (AT): Following (Madry et al., 2018), adversarial training is given as the following min-max problem:

$$\min_w \mathbb{E}\left[\max_{\|\delta\|_\infty \le \epsilon} \mathcal{L}(f(x+\delta; w), y)\right] \tag{1}$$

with $f : \mathbb{R}^d \mapsto \mathbb{R}^K$ being a classifier with K classes and weights w, (x, y) being training examples and $\mathcal{L}$ the cross-entropy loss. The inner maximization problem for computing adversarial examples $\tilde{x} := x + \delta$ is solved approximately using projected gradient ascent ensuring the L_∞-constraint as well as a box constraint for images, i.e., $\tilde{x}_i \in [0, 1]$. Maximizing the cross-entropy loss is equivalent to finding adversarial examples with *minimal* confidence in the true class. In contrast to training *only* on adversarial examples, others compute adversarial examples only for *half* the examples of each mini-batch (Szegedy et al., 2014). Compared to Eq. (1), 50%/50% adversarial training effectively minimizes

$$\underbrace{\mathbb{E}\Big[\max_{\|\delta\|_\infty \le \epsilon} \mathcal{L}(f(x+\delta; w), y)\Big]}_{\text{50\% adversarial training}} + \underbrace{\mathbb{E}\big[\mathcal{L}(f(x; w), y)\big]}_{\text{50\% ``clean'' training}}. \tag{2}$$

Intuitively, by balancing both terms in Eq. (2), the trade-off between accuracy and robustness can already be optimized to some extent (Stutz et al., 2019).

Algorithm 1 Confidence-Calibrated Adversarial Training (CCAT). The only changes compared to standard adversarial training are the attack (line 4) and the probability distribution over the classes (lines 6 and 7), which becomes more uniform as distance $\|\delta\|_\infty$ increases. During testing, low-confidence (adversarial) examples are rejected.

```
 1: while true do
 2:   choose random batch $(x_1, y_1), \ldots, (x_B, y_B)$.
 3:   for $b = 1, \ldots, B/2$ do
 4:     $\delta_b := \operatorname{argmax}_{\|\delta\|_\infty \le \epsilon} \max_{k \neq y_b} f_k(x_b + \delta)$  (see Eq. (3))
 5:     $\tilde{x}_b := x_b + \delta_b$
 6:     $\lambda := (1 - \min(1, \|\delta_b\|_\infty/\epsilon))^\rho$  (see Eq. (5))
 7:     $\tilde{y}_b := \lambda\, \text{one\_hot}(y_b) + (1 - \lambda)1/K$  (see Eq. (4))
 8:   end for
 9:   update parameters using Eq. (2):
10:       $\sum_{b=1}^{B/2} \mathcal{L}(f(\tilde{x}_b), \tilde{y}_b) + \sum_{b=B/2}^{B} \mathcal{L}(f(x_b), y_b)$
11: end while
```

Trained on L_∞ adversarial examples, the robustness of adversarial training does not generalize to previously unseen adversarial examples, including larger perturbations or other L_p adversarial examples. We hypothesize that this is because adversarial training explicitly enforces high-confidence predictions on L_∞ adversarial examples within the ϵ-ball seen during training ("seen" in Fig. 1). However, this behavior is difficult to extrapolate to arbitrary regions in a meaningful way. Thus, it is not surprising that adversarial examples can often be found right beyond the ϵ-ball used during training, cf. Fig. 1 (top left). This can be described as "overfitting" to the L_∞ adversarial examples used during training. Also, larger ϵ-balls around training examples might include (clean) examples from other classes. Then, the inner maximization problem of Eq. (1) will focus on these regions and reduce accuracy (Jacobsen et al., 2019b;a).

Confidence-calibrated adversarial training (CCAT): We address these problems with minimal modifications, as outlined in Alg. 1. During training, we train the network to predict a convex combination of (correct) one-hot distribution (on clean examples) and uniform distribution (on adversarial examples) as target distribution within the cross-entropy loss. We found that the model extrapolates the low-confidence predictions on adversarial examples beyond the ϵ-ball, i.e., to larger perturbations, unseen attacks or distal adversarial examples. During testing, these adversarial examples can be rejected by confidence thresholding.

As CCAT enforces low-confidence on adversarial examples, our adaptive attack explicitly maximizes the confidence in any other label $k \neq y$, given example x with true label y:

$$\max_{\|\delta\|_\infty \le \epsilon} \max_{k \neq y} f_k(x + \delta; w) \tag{3}$$

where f_k denotes the confidence in class k. Then, given an adversarial example from Eq. (3) during training, CCAT uses the following combination of uniform and one-hot distribution as target for the cross-entropy loss:

$$\tilde{y} = \lambda \text{ one\_hot}(y) + (1 - \lambda)^{1}/_{K}, \tag{4}$$

with $\lambda \in [0, 1]$ and $\text{one\_hot}(y) \in \{0, 1\}^K$ denoting the one-hot vector corresponding to class y. Thus, we enforce a convex combination of the original label distribution and the uniform distribution which is controlled by the parameter λ. We choose λ to decrease with the distance $\|\delta\|_\infty$ of the adversarial example to the attacked example x with the intention to enforce uniform predictions when $\|\delta\|_\infty = \epsilon$:

$$\lambda = \left(1 - \min\left(1, \frac{\|\delta\|_\infty}{\epsilon}\right)\right)^\rho \tag{5}$$

Here, the speed of decay is controlled by the parameter ρ. For $\rho = 10$, Fig. 1 (top right) shows the transition as approximated by the network. We train on 50% clean and 50% adversarial examples in each batch, as in Eq. (2), such that the network has an incentive to predict correct labels.

3. Experiments

We evaluate CCAT on MNIST (LeCun et al., 1998), SVHN (Netzer et al., 2011) and Cifar10 (Krizhevsky, 2009) as well as MNIST-C (Mu & Gilmer, 2019) and Cifar10-C (Hendrycks & Dietterich, 2019) with corrupted examples (e.g., blur, noise, compression, transforms etc.). As CCAT allows to reject (adversarial) examples by confidence-thresholding before classifying them, we adapt attacks to CCAT by directly maximizing confidence. Additionally, we generalize the commonly used robust test error (Madry et al., 2018) to our confidence-thresholded setting. As we will see, this "reject option", is also beneficial for standard adversarial training (AT).

Adaptive Attacks: We consider 7 different L_p attacks, $p \in \{\infty, 2, 1, 0\}$, to maximize the confidence of adversarial examples, cf. Eq. (3), as effective adaptive attack against CCAT. We use a backtracking scheme, momentum (Dong et al., 2018) and run exactly T iterations, choosing the iterate corresponding to the best objective. For Eq. (3), $T = 1000$ iterations and 10 random restarts with random initialization plus one restart with zero initialization are used; $T = 200$ with 50 random restarts are used for maximizing cross-entropy loss as in Eq. (1). As black-box attacks, we use (Ilyas et al., 2018) with $T = 1000$ iterations and 10 restarts, (Narodytska & Kasiviswanathan, 2017) for $T = 1000$ iterations and 10 restarts, and (Andriushchenko et al., 2019) (for L_∞ and L_2) with $T = 5000$ iterations. In the case of L_0 we also use (Croce & Hein, 2019). For L_∞, L_2, L_1 and L_0 attacks, we set **ϵ to $\mathbf{0.3, 3, 18, 15}$ (MNIST) or $\mathbf{0.03, 2, 24, 10}$ (SVHN/Cifar10)**.

We also evaluate adversarial frames (Zajac et al., 2019), which allow a 2 (MNIST) or 3 (SVHN/Cifar10) pixel border to be manipulated to maximize Eq. (3). And distal adversarial examples apply PGD, L_∞-constrained with $\epsilon = 0.3$ (MNIST) or $\epsilon = 0.03$ (SVHN/Cifar10), to maximize confidence in *any* class starting from a (uniform) random image.

Confidence-Thresholded Robust Test Error (RErr): We consider the widely used robust test error (RErr), defined as the test error when all test examples are allowed to be attacked, i.e., modified within the chosen threat model. However, standard RErr does not take into account the option of rejecting (adversarial) examples. Therefore, we propose a generalized definition adapted to our confidence-thresholded setting: For fixed confidence threshold τ, the **confidence-thresholded RErr** is defined as the **test error on test examples that can be modified within the chosen threat model *and* pass confidence thresholding**. For $\tau = 0$ (i.e., all examples pass confidence thresholding) this reduces to the standard RErr, comparable to related work. In the following, we always report the per-example worst-case (confidence-thresholded) RErr, i.e., per example, the adversarial example with highest confidence is used across all attacks. A (clean) **confidence-thresholded test error (Err)** is obtained similarly.

The confidence threshold τ is chosen following a simple detection setting: adversarial example are *negatives* and correctly classified clean examples are *positives*. Then, we take the confidence threshold τ corresponding to the extremely conservative choice of **99% true positive rate (TPR)**: at most 1% of correctly classified clean examples can be rejected. Thus, τ is determined *only* by correctly classified clean examples, independent of adversarial examples.

Training and Baselines: We train 50%/50% AT and CCAT with L_∞ attacks using $T = 40$ iterations and $\epsilon = 0.3$ (MNIST) or $\epsilon = 0.03$ (SVHN/Cifar10). We use ResNet-20 (He et al., 2016), implemented in PyTorch (Paszke et al., 2017), trained using stochastic gradient descent. For CCAT, we use $\rho = 10$. We also compare to multi-steepest descent (MSD) adversarial training (Maini et al., 2019) using L_∞, L_2 and L_1 attacks during training. We use the provided pre-trained LeNet (MNIST) and pre-activation ResNet-18 (Cifar10) with ϵ set to $0.3, 1.5, 12$ and $0.03, 0.5, 12$, respectively. The L_2, L_1 attacks in Tab. 1 (larger ϵ) are unseen.

3.1. Main Results (Tab. 1)

Robustness Against Unseen L_p Attacks: Considering Tab. 1 and L_∞ adversarial examples as seen during training, CCAT exhibits comparable robustness to AT. With 7.4%/67.9% RErr on MNIST/Cifar10, CCAT lacks behind AT (1.7%/62.7%) only slightly. However, regarding unseen attacks, AT's robustness deteriorates quickly. On Cifar10, for example, RErr goes up to 93.7%, 98.4%, 98.4% and 72.4% for larger L_∞, L_2, L_1 and L_0 attacks. Except for larger L_∞ perturbations, CCAT's robustness generalizes to

MNIST: FPR and **RErr** ↓ in % for τ@99%TPR							**FPR**↓
L_p	L_∞	L_∞	L_2	L_1	L_0	adv.	distal
ϵ	0.3	0.4	3	18	15	frames	
	seen	**unseen**	**unseen**	**unseen**	**unseen**	**unseen**	**unseen**
Norm	100.0	100.0	100.0	100.0	92.3	87.7	100.0
AT	**1.7**	100.0	81.5	24.6	23.9	73.7	100.0
CCAT	7.4	**11.9**	**0.3**	**1.8**	**14.8**	**0.2**	**0**
MSD	34.3	98.9	59.2	55.9	66.4	8.8	100.0

CIFAR10: FPR and **RErr** ↓ in % for τ@99%TPR							**FPR**↓
L_p	L_∞	L_∞	L_2	L_1	L_0	adv.	distal
ϵ	0.03	0.06	2	24	10	frames	
	seen	**unseen**	**unseen**	**unseen**	**unseen**	**unseen**	**unseen**
Norm	100.0	100.0	100.0	100.0	77.1	96.6	83.3
AT	62.7	93.7	98.4	98.4	72.4	78.7	75.0
CCAT	67.9	92.0	**51.8**	**58.5**	**20.3**	**65.1**	**0**
MSD	**53.0**	**89.4**	87.8	67.4	35.8	82.6	76.7

Table 1: Main Results: Generalizing Robustness. For L_∞, L_2, L_1, L_0 attacks and adversarial frames, we report per-example worst-case RErr at 99%TPR across all attacks; ϵ is reported in the corresponding columns. For distal adversarial examples, we report false positive rate (FPR). L_∞ attacks with ϵ=0.3 on MNIST and $\epsilon = 0.03$ on SVHN/Cifar10 were used for training (**seen**). The remaining attacks were not encountered during training (**unseen**). CCAT outperforms AT and MSD regarding robustness against unseen attacks. For MSD, we used pre-trained models with different architectures.

these unseen attacks significantly better, with 51.8%, 58.5%, and 20.3% for L_2, L_1 and L_0 attacks. On MNIST, AT generalizes better to L_1 and L_0 attacks, possibly due to the large L_∞-ball used during training ($\epsilon = 0.3$). Overall, CCAT improves robustness against arbitrary (unseen) L_p attacks.

Comparison to Baselines: MSD outperforms both AT and CCAT on Cifar10 on the L_∞ adversarial examples seen during training: 53% RErr compared to 67.9% and 62.7% for CCAT and AT. This might be a result of training on 100% adversarial examples (instead of 50%/50%, cf. Eq. (2)). However, regarding L_2, L_1 and L_0 attacks, CCAT outperforms MSD with 51.8%, 58.5% and 20.3% compared to 87.8%, 67.4% and 35.8% in terms of RErr. This is surprising, as MSD trains on both L_2 and L_1 attacks with smaller ϵ, while CCAT does not. On MNIST, due to the small network (LeNet), MSD is not able to compete with AT or CCAT.

Robustness Against Unconventional Attacks: Against adversarial frames, robustness of AT reduces to 73.7% RErr, even on MNIST, while CCAT achieves 0.2%. MSD, in contrast, is able to preserve robustness better with 8.8% RErr, which might be due to the L_2 and L_1 attacks seen during training. Regarding distal adversarial examples, we report **false positive rate (FPR)**, i.e., the fraction of distal adversarial examples *not* rejected due to high confidence. CCAT consistently reaches 0% FPR, in contrast to high FPRs for AT and MSD.

	MNIST: Err in %		**MNIST-C** Err in %	**CIFAR10:** Err in %		**CIFAR10-C** Err in %
	τ=0	99% TPR	99% TPR	τ=0	99% TPR	99% TPR
Norm	0.4	0.1	32.8	**8.3**	7.4	12.3
AT	0.5	**0**	12.6	16.6	15.5	16.2
CCAT	**0.3**	0.1	**5.7**	10.1	**6.7**	**8.5**
*MSD	1.8	0.9	6.0	18.4	17.6	19.3

Table 2: Test Error (Err) for $\tau = 0$, i.e., non-thresholded, and τ@99%TPR, i.e., confidence-thresholded, on test sets and corrupted examples. For corrupted examples, we report the mean across all corruptions. On Cifar10, CCAT reduces Err significantly. For MSD, we used pre-trained models with different architectures, not directly comparable.

Improved Test Error: In Tab. 2, CCAT also improves Err compared to AT, coming close to normal training. On Cifar10, CCAT achieves 6.7% Err (with confidence-thresholding) and outperforms normal training with 7.4% and AT with 15.5%, in spite of using 50%/50% AT. Furthermore, CCAT also improves results on corrupted examples in terms of mean Err across all corruptions. On Cifar10-C, for example, CCAT achieves 8.5% compared 16.2% for AT and 12.3% for normal training.

The full paper (Stutz et al., 2020) includes an ablation study regarding our adaptive attack and evaluation metrics as well as results on SVHN. The **supplementary material** additionally includes detailed descriptions of our PGD attack (including pseudo-code), an in-depth discussion of our (confidence-thresholded) RErr, more details regarding baselines, additional ablation studies and qualitative examples.

4. Conclusion

Adversarial training results in robust models against the threat model *seen* during training, e.g., L_∞ adversarial examples. However, generalization to *unseen* attacks such as other L_p adversarial examples or larger L_∞ perturbations is insufficient. We propose **confidence-calibrated adversarial training (CCAT)** which biases the model towards low confidence predictions on adversarial examples and beyond. Then, adversarial examples can easily be rejected based on their confidence. Trained exclusively on L_∞ adversarial examples, CCAT improves robustness against unseen threat models such as larger L_∞, L_2, L_1 and L_0 adversarial examples, adversarial frames, distal adversarial examples and corrupted examples. Additionally, accuracy is improved in comparison to adversarial training. We thoroughly evaluated CCAT using 7 different white-and black-box attacks with up to 50 random restarts and 5000 iterations. These attacks where adapted to CCAT by directly maximizing confidence. We reported worst-case robust test error, extended to our confidence-thresholded setting, across *all* attacks.

References

Andriushchenko, M., Croce, F., Flammarion, N., and Hein, M. Square attack: a query-efficient black-box adversarial attack via random search. *arXiv.org*, 1912.00049, 2019.

Croce, F. and Hein, M. Sparse and imperceivable adversarial attacks. *arXiv.org*, abs/1909.05040, 2019.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. Boosting adversarial attacks with momentum. In *CVPR*, 2018.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *CVPR*, 2016.

Hein, M., Andriushchenko, M., and Bitterwolf, J. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. *CVPR*, 2019.

Hendrycks, D. and Dietterich, T. G. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

Ilyas, A., Engstrom, L., Athalye, A., and Lin, J. Black-box adversarial attacks with limited queries and information. In *ICML*, 2018.

Jacobsen, J., Behrmann, J., Carlini, N., Tramèr, F., and Papernot, N. Exploiting excessive invariance caused by norm-bounded adversarial robustness. *arXiv.org*, abs/1903.10484, 2019a.

Jacobsen, J., Behrmann, J., Zemel, R. S., and Bethge, M. Excessive invariance causes adversarial vulnerability. In *ICLR*, 2019b.

Kang, D., Sun, Y., Hendrycks, D., Brown, T., and Steinhardt, J. Testing robustness against unforeseen adversaries. *arXiv.org*, abs/1908.08016, 2019.

Krizhevsky, A. Learning multiple layers of features from tiny images. Technical report, 2009.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324, 1998.

Li, B., Chen, C., Wang, W., and Carin, L. On norm-agnostic robustness of adversarial training. *arXiv.org*, abs/1905.06455, 2019.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. *ICLR*, 2018.

Maini, P., Wong, E., and Kolter, J. Z. Adversarial robustness against the union of multiple perturbation models. *arXiv.org*, abs/1909.04068, 2019.

Mu, N. and Gilmer, J. Mnist-c: A robustness benchmark for computer vision. *ICML Workshops*, 2019.

Narodytska, N. and Kasiviswanathan, S. P. Simple black-box adversarial attacks on deep neural networks. In *CVPR Workshops*, 2017.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NeurIPS*, 2011.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. In *NeurIPS Workshops*, 2017.

Raghunathan, A., Xie, S. M., Yang, F., Duchi, J. C., and Liang, P. Adversarial training can hurt generalization. *arXiv.org*, abs/1906.06032, 2019.

Sharma, Y. and Chen, P. Attacking the madry defense model with l_1-based adversarial examples. In *ICLR Workshops*, 2018.

Stutz, D., Hein, M., and Schiele, B. Disentangling adversarial robustness and generalization. *CVPR*, 2019.

Stutz, D., Hein, M., and Schiele, B. Confidence-calibrated adversarial training: Generalizing to unseen attacks. *ICML*, 2020.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., and Fergus, R. Intriguing properties of neural networks. In *ICLR*, 2014.

Tramèr, F. and Boneh, D. Adversarial training and robustness for multiple perturbations. In *NeurIPS*, 2019.

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. In *ICLR*, 2019.

Zajac, M., Zolna, K., Rostamzadeh, N., and Pinheiro, P. O. Adversarial framing for image and video classification. In *AAAI Workshops*, 2019.

Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. Theoretically principled trade-off between robustness and accuracy. In *ICML*, 2019.