Supplementary Material for Confidence-Calibrated Adversarial Training: Generalizing to Unseen Attacks

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

Abstract

This document provides supplementary material for **confidence-calibrated adversarial training (CCAT)**. First, in Sec. B, we provide the proof of Proposition 1, showing that there exist problems where standard adversarial training (AT) is unable to reconcile robustness and accuracy, while CCAT is able to obtain *both* robustness *and* accuracy. In Sec. C, to promote reproducibility and emphasize our thorough evaluation, we discuss details regarding the used attacks, training procedure, baselines and evaluation metrics. Furthermore, Sec. C includes additional experimental results in support of the observations in the main paper. For example, we present results for confidence threshold at 95% and 98% true positive rate (TPR), results for the evaluated detection baselines as well as per-attack and per-corruption results for in-depth analysis. We also include qualitative results highlighting how CCAT obtains robustness through confidence thresholding. Code and pre-trained models are available at davidstutz.de/ccat.

A. Introduction

Confidence-calibrated adversarial training (CCAT) biases the network towards low-confidence predictions on adversarial examples. This is achieved by training the network to predict a uniform distribution between (correct) one-hot and uniform distribution which becomes more uniform as the distance to the attacked example increases. In the main paper, we show that CCAT addresses two problems of standard adversarial training (AT) as, e.g., proposed in (Madry et al., 2018): the poor generalization of robustness to attacks not employed during training, e.g., other L_p attacks or larger perturbations, and the reduced accuracy. We show that CCAT, trained only on L_∞ adversarial examples, improves robustness against previously unseen attacks through confidence thresholding, i.e., rejecting low-confidence (adversarial) examples. Furthermore, we demonstrate that CCAT is able to improve accuracy compared to adversarial training. In this document, Sec. B provides the proof of Proposition 1. Then, Sec. C includes details on our experimental setup, emphasizing our efforts to thoroughly evaluate CCAT, and additional experimental results allowing an in-depth analysis of the robustness obtained through CCAT.

In Sec. B, corresponding to the proof of Proposition 1, we show that there exist problems where standard adversarial training is indeed unable to reconcile robustness and accuracy. CCAT, in contrast, is able to obtain *both* robustness and accuracy, given that the "transition" between one-hot and uniform distribution used during training is chosen appropriately.

In Sec. C, we discuss our thorough experimental setup to facilitate reproducibility and present additional experimental results in support of the conclusions of the main paper. In the first part of Sec. C, starting with Sec. C.1, we provide a detailed description of the employed projected gradient descent (PGD) attack with momentum and backtracking, including pseudo-code and used hyper-parameters. Similarly, we discuss details of the used black-box attacks. In Sec. C.2, we include details on our training procedure, especially for CCAT. In Sec. C.3, we discuss the evaluated baselines, i.e., (Maini et al., 2020; Madry et al., 2018; Zhang et al., 2019; Lee et al., 2018; Ma et al., 2018). Then, in Sec. C.4, we discuss the employed evaluation metrics, focusing on our *confidence-thresholded* robust test error (RErr). In the second part of Sec. C, starting with Sec. C.5, we perform ablation studies considering our attack and CCAT. Regarding the attack, we demonstrate the importance of enough iterations, backtracking and appropriate initialization to successfully attack CCAT. Regarding CCAT, we consider various values for the hyper-parameter ρ which controls the transition from one-hot to uniform distribution

[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>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

during training. In Sec. C.6, we analyze how CCAT achieves robustness by considering its behavior in adversarial directions as well as in between clean examples. Finally, we provide additional experimental results in Sec. C.7: our main results, i.e., robustness against seen and unseen adversarial examples, for a confidence threshold at 95% and 98% true positive rate (TPR), per-attack results on all datasets and per-corruption results on MNIST-C (Mu & Gilmer, 2019) and Cifar10-C (Hendrycks & Dietterich, 2019).

B. Proof of Proposition 1

Adversarial training usually results in reduced accuracy on clean examples. In practice, training on 50% clean and 50% adversarial examples instead of training *only* on adversarial examples allows to control the robustness-accuracy trade-off to some extent, i.e., increase accuracy while sacrificing robustness. The following proposition shows that there exist problems where adversarial training is unable to reconcile robustness and accuracy, while CCAT is able to obtain *both*:

Proposition 1. *We consider a classification problem with two points $x = 0$ and $x = \epsilon$ in $\mathbb{R}$ with deterministic labels, i.e., $p(y = 2|x = 0) = 1$ and $p(y = 1|x = \epsilon) = 1$, such that the problem is fully determined by the probability $p_0 = p(x = 0)$. The Bayes error of this classification problem is zero. Let the predicted probability distribution over classes be $\tilde{p}(y|x) = \frac{e^{g_y(x)}}{e^{g_1(x)}+e^{g_2(x)}}$, where $g : \mathbb{R}^d \to \mathbb{R}^2$ is the classifier and we assume that the function $\lambda : \mathbb{R}_+ \to [0, 1]$ used in CCAT is monotonically decreasing and $\lambda(0) = 1$. Then, the error of the Bayes optimal classifier (with cross-entropy loss) for*

- *adversarial training on 100% adversarial examples is $\min\{p_0, 1 - p_0\}$.*
- *adversarial training on 50%/50% adversarial/clean examples per batch is $\min\{p_0, 1 - p_0\}$.*
- *CCAT on 50% clean and 50% adversarial examples is* zero *if $\lambda(\epsilon) < \min\left\{{p_0}/{1-p_0}, {1-p_0}/{p_0}\right\}$.*

Proof. First, we stress that we are dealing with three different probability distributions over the labels: the true one $p(y|x)$, the imposed one during training $\hat{p}(y|x)$ and the predicted one $\tilde{p}(y|x)$. We also note that $\hat{p}$ depends on λ as follows:

$$\hat{p}(k) = \lambda p_y(k) + (1 - \lambda)u(k) \tag{1}$$

where $p_y(k)$ is the original one-hot distribution, i.e., $p_y(k) = 1$ iff $k = y$ and $p_y(k) = 0$ otherwise with y being the true label, and $u(k) = {1}/{K}$ is the uniform distribution. We note that this is merely an alternative formulation to the target distribution as outlined in the main paper. Also note that λ itself is a function of the norm $\|\delta\|$; here, this dependence is made explicit by writing $\hat{p}(\lambda)(y|x)$. This makes the expressions for the expected loss of CCAT slightly more complicated. We first derive the Bayes optimal classifier and its loss for CCAT. We introduce

$$a = g_1(0) - g_2(0), \quad b = g_1(\epsilon) - g_2(\epsilon). \tag{2}$$

and express the logarithm of the predicted probabilities (confidences) of class 1 and 2 in terms of these quantities.

$$-\log \tilde{p}(y = 2|x = x) = -\log\left(\frac{e^{g_2(x)}}{e^{g_1(x)} + e^{g_2(x)}}\right) = \log\left(1 + e^{g_1(x)-g_2(x)}\right) = \begin{cases} \log\left(1 + e^{a}\right) & \text{if } x = 0 \\ \log(1 + e^{b}) & \text{if } x = \epsilon \end{cases}.$$

$$-\log \tilde{p}(y = 1|x = x) = -\log\left(\frac{e^{g_1(x)}}{e^{g_1(x)} + e^{g_2(x)}}\right) = \log\left(1 + e^{g_2(x)-g_1(x)}\right) = \begin{cases} \log(1 + e^{-a}) & \text{if } x = 0 \\ \log\left(1 + e^{-b}\right) & \text{if } x = \epsilon \end{cases}.$$

We consider the approach by (Madry et al., 2018) with 100% adversarial training. The expected loss can be written as

$$\begin{aligned}
&\mathbb{E}\Big[\max\nolimits_{\|\delta\|_\infty \le \epsilon} L(y, g(x + \delta)\Big] = \mathbb{E}\Big[\mathbb{E}\Big[\max\nolimits_{\|\delta\|_\infty \le \epsilon} L(y, g(x + \delta)|x\Big]\Big]\\
=&p(x = 0)p(y = 2|x = 0) \max\left\{-\log\left(\tilde{p}(y = 2|x = 0)\right), -\log\left(\tilde{p}(y = 2|x = \epsilon)\right)\right\}\\
&\ + (1 - p(x = 0))p(y = 1|x = \epsilon) \max\left\{-\log\left(\tilde{p}(y = 1|x = 0)\right), -\log\left(\tilde{p}(y = 1|x = \epsilon)\right)\right\}\\
=&p(x = 0) \max\left\{-\log\left(\tilde{p}(y = 2|x = 0)\right), -\log\left(\tilde{p}(y = 2|x = \epsilon)\right)\right\}\\
&\ + (1 - p(x = 0)) \max\left\{-\log\left(\tilde{p}(y = 1|x = 0)\right), -\log\left(\tilde{p}(y = 1|x = \epsilon)\right)\right\}
\end{aligned}$$

This yields in terms of the parameters a, b the expected loss:

$$L(a,b) = \max\Big\{\log(1+e^a), \log(1+e^b)\Big\}p_0 + \max\Big\{\log(1+e^{-a}), \log(1+e^{-b})\Big\}(1-p_0)$$

The expected loss is minimized if $a = b$ as then both maxima are minimal. This results in the expected loss

$$L(a) = \log(1+e^a)p_0 + \log(1+e^{-a})(1-p_0).$$

The critical point is attained at $a^* = b^* = \log\left(\frac{1-p_0}{p_0}\right)$. Thus

$$a^* = b^* = \begin{cases} > 0 & \text{if } p_0 < \frac{1}{2}, \\ < 0 & \text{if } p_0 > \frac{1}{2}. \end{cases}$$

Thus, we classify $x = 0$ correctly, if $p_0 > \frac{1}{2}$ and $x = \epsilon$ correctly if $p_0 < \frac{1}{2}$. As result, the error of 100% adversarial training is given by $\min\{p_0, 1-p_0\}$ whereas the Bayes optimal error is zero as the problem is deterministic.

Next we consider 50% adversarial plus 50% clean training. The expected loss

$$\mathbb{E}\Big[\max_{\|\delta\|_\infty \le \epsilon} L(y, g(x+\delta)\Big] + \mathbb{E}\Big[L(y, g(x+\delta)\Big],$$

can be written as

$$\begin{aligned} L(a,b) = &\max\Big\{\log(1+e^a), \log(1+e^b)\Big\}p_0 \\ &+ \max\Big\{\log(1+e^{-a}), \log(1+e^{-b})\Big\}(1-p_0) \\ &+ \log(1+e^a)p_0 + \log(1+e^{-b})(1-p_0) \end{aligned}$$

We make a case distinction. If $a \ge b$, then the loss reduces to

$$\begin{aligned} L(a,b) = &\log(1+e^a)p_0 + \log(1+e^{-b})(1-p_0) \\ &+ \log(1+e^a)p_0 + \log(1+e^{-b})(1-p_0) \\ &\ge L(a,a) \\ &= 2\log(1+e^a)p_0 + 2\log(1+e^{-a})(1-p_0) \end{aligned}$$

Solving for the critical point yields $a^* = \log\left(\frac{1-p_0}{p_0}\right) = b^*$. Next we consider the set $a \le b$. This yields the loss

$$\begin{aligned} L(a,b) = &\log(1+e^b)p_0 + \log(1+e^{-a})(1-p_0) \\ &+ \log(1+e^a)p_0 + \log(1+e^{-b})(1-p_0) \end{aligned}$$

Solving for the critical point yields $a^* = \log\left(\frac{1-p_0}{p_0}\right) = b^*$ which fulfills $a \le b$. Actually, it coincides with the solution found already for 100% adversarial training and thus resulting error of 50% adversarial, 50% clean training is again $\min\{p_0, 1-p_0\}$ which is not equal to the Bayes optimal error.

For our confidence-calibrated adversarial training one first has to solve

$$\delta_x^*(j) = \underset{\|\delta\|_\infty \le \epsilon}{\text{argmax}} \max_{k \ne j} \tilde{p}(y = k \,|\, x + \delta).$$

We get with the expressions above (note that we have a binary classification problem):

$$\delta_0^*(1) = \underset{\|\delta\|_\infty \leq \epsilon}{\text{argmax}}\ \tilde{p}(y=2|0+\delta) = \begin{cases} 0 & \text{if } a < b \\ \epsilon & \text{else.} \end{cases}.$$

$$\delta_0^*(2) = \underset{\|\delta\|_\infty \leq \epsilon}{\text{argmax}}\ \tilde{p}(y=1|0+\delta) = \begin{cases} \epsilon & \text{if } a < b \\ 0 & \text{else.} \end{cases}.$$

$$\delta_\epsilon^*(1) = \underset{\|\delta\|_\infty \leq \epsilon}{\text{argmax}}\ \tilde{p}(y=2|\epsilon+\delta) = \begin{cases} -\epsilon & \text{if } a < b \\ 0 & \text{else.} \end{cases}.$$

$$\delta_\epsilon^*(2) = \underset{\|\delta\|_\infty \leq \epsilon}{\text{argmax}}\ \tilde{p}(y=1|\epsilon+\delta) = \begin{cases} 0 & \text{if } a < b \\ -\epsilon & \text{else.} \end{cases}.$$

Note that the imposed distribution $\hat{p}$ over the classes depends on the true label y of x and $\delta_x^*(y)$ and then $\lambda(\|\delta_x^*(y)\|_\infty)$. Due to the simple structure of the problem it holds that $\|\delta_x^*(y)\|_\infty$ is either 0 or ϵ.

In CCAT we use for 50% of the batch the standard cross-entropy loss and for the other 50% we use the following loss:

$$L\left(\hat{p}_y\big(\lambda\big(\,\|\delta_x^*(y)\|\,\big)\big)(x), \tilde{p}(x)\right) = -\sum_{j=1}^{2} \hat{p}_y\big(\lambda\big(\,\|\delta_x^*(y)\|\,\big)\big)(y=j\,|\,x=x) \log\left(\tilde{p}\big(y=j\,|\,x=x+\delta_x^*(j)\big)\right).$$

The corresponding expected loss is then given by

$$\begin{aligned}
&\mathbb{E}\Big[L\Big(\hat{p}_y\big(\lambda\big(\,\|\delta_x^*(y)\|\,\big)\big)(x), \tilde{p}(x)\Big)\Big] = \mathbb{E}\Big[\mathbb{E}\Big[L\Big(\hat{p}_y\big(\lambda\big(\,\|\delta_x^*(y)\|\,\big)\big)(x), \tilde{p}(x)\Big)\Big|\,x\Big]\Big] \\
=&p(x=0)\,\mathbb{E}\Big[L\Big(\hat{p}_y\big(\lambda\big(\,\|\delta_0^*(y)\|\,\big)\big)(0), \tilde{p}(0)\Big)\Big|\,x=0\Big] + p(x=\epsilon)\,\mathbb{E}\Big[L\Big(\hat{p}_Y\big(\lambda\big(\,\|\delta_\epsilon^*(y)\|\,\big)\big)(\epsilon), \tilde{p}(\epsilon)\Big)\Big|\,x=\epsilon\Big],
\end{aligned}$$

where $p(x=0) = p_0$ and $p(x=\epsilon) = 1 - p(x=0) = 1 - p_0$. With the true conditional probabilities $p(y=s\,|\,x)$ we get

$$\begin{aligned}
\mathbb{E}\Big[L\Big(\hat{p}_y\big(\lambda(\delta)\big)(x), \tilde{p}(x)\Big)\Big|\,x\Big] &= \sum_{s=1}^{2} p(y=s\,|\,x)\, L\Big(\hat{p}_s\big(\lambda\big(\,\|\delta_x^*(s)\|\,\big)\big)(x), \tilde{p}(x)\Big) \\
= -\sum_{s=1}^{2} p(y=s\,|\,x) &\sum_{j=1}^{2} \hat{p}_s\big(\lambda\big(\,\|\delta_x^*(s)\|\,\big)\big)(y=j\,|\,x) \log\left(\tilde{p}(y=j\,|\,x=x+\delta_x^*(s))\right)
\end{aligned}$$

For our problem it holds with $p(y=2\,|\,x=0) = p(y=1\,|\,x=\epsilon) = 1$ (by assumption). Thus,

$$\begin{aligned}
\mathbb{E}\Big[L\Big(\hat{p}_y\big(\lambda(\delta)\big)(x), \tilde{p}(x)\Big)\Big|x=0\Big] &= -\sum_{j=1}^{2} \hat{p}_2\big(\lambda\big(\,\|\delta_0^*(2)\|\,\big)\big)(y=j\,|\,x=0) \log\left(\tilde{p}(y=j\,|\,x=x+\delta_0^*(2))\right) \\
\mathbb{E}\Big[L\Big(\hat{p}_y\big(\lambda(\delta)\big)(x), \tilde{p}(x)\Big)\Big|\,x=\epsilon\Big] &= -\sum_{j=1}^{2} \hat{p}_1\big(\lambda\big(\,\|\delta_\epsilon^*(1)\|\,\big)\big)(y=j\,|\,x=\epsilon) \log\left(\tilde{p}(y=j\,|\,x=x+\delta_\epsilon^*(1))\right)
\end{aligned}$$

As $\|\delta_x^*(y)\|_\infty$ is either 0 or ϵ and $\lambda(0) = 1$ we use in the following for simplicity the notation $\lambda = \lambda(\|\epsilon\|_\infty)$. Moreover, note that

$$\hat{p}_y(\lambda)(y=j|x=x) = \begin{cases} \lambda + \frac{(1-\lambda)}{K} & \text{if } y=j, \\ \frac{(1-\lambda)}{K} & \text{else} \end{cases},$$

where K is the number of classes. Thus, $K=2$ in our example and we note that $\lambda + \frac{(1-\lambda)}{2} = \frac{1+\lambda}{2}$.

With this we can write the total loss (remember that we have half normal cross-entropy loss and half the loss for the adversarial part with the modified “labels”) as

$$\begin{aligned}L(a,b) =&p_0\Big[\log(1+e^a)\mathbb{1}_{a\geq b}+\mathbb{1}_{a<b}\Big(\frac{(1+\lambda)}{2}\log(1+e^b)+\frac{(1-\lambda)}{2}\log(1+e^{-b})\Big)\Big]\\&+(1-p_0)\Big[\log(1+e^{-b})\mathbb{1}_{a\geq b}+\mathbb{1}_{a<b}\Big(\frac{(1+\lambda)}{2}\log(1+e^{-a})+\frac{(1-\lambda)}{2}\log(1+e^{a})\Big)\Big]\\&+\log(1+e^a)p_0+\log(1+e^{-b})(1-p_0),\end{aligned}$$

where we have omitted a global factor $\frac{1}{2}$ for better readability (the last row is the cross-entropy loss). We distinguish two sets in the optimization. First we consider the case $a \geq b$. Then it is easy to see that in order to minimize the loss we have $a = b$.

$$\partial_a L = 2\frac{e^a}{1+e^a}p_0 - \frac{e^{-a}}{1+e^{-a}}(1-p_0)$$

This yields $e^a = \frac{1-p_0}{p_0}$ or $a = \log\left(\frac{1-p_0}{p_0}\right)$ and the minimum for $a \geq b$ is attained on the boundary of the domain of $a \leq b$. The other case is $a \leq b$. We get

$$\begin{aligned}\partial_a L =&\Big[\frac{(1+\lambda)}{2}\frac{-e^{-a}}{1+e^{-a}}+\frac{(1-\lambda)}{2}\frac{e^a}{1+e^a}\Big](1-p_0)+p_0\frac{e^a}{1+e^a}\\\partial_b L =&\Big[\frac{(1+\lambda)}{2}\frac{e^b}{1+e^b}+\frac{(1-\lambda)}{2}\frac{-e^b}{1+e^{-b}}\Big]p_0+(1-p_0)\frac{-e^{-b}}{1+e^{-b}}\end{aligned}$$

This yields the solution

$$a^* = \log\left(\frac{\frac{1+\lambda}{2}(1-p_0)}{p_0+\frac{1-\lambda}{2}(1-p_0)}\right), \qquad b^* = \log\left(\frac{\frac{1-\lambda}{2}p_0+(1-p_0)}{\frac{1+\lambda}{2}p_0}\right)$$

It is straightforward to check that $a^* < b^*$ for all $0 < p_0 < 1$, indeed we have

$$\frac{\frac{1+\lambda}{2}(1-p_0)}{p_0+\frac{1-\lambda}{2}(1-p_0)} = \frac{\frac{1+\lambda}{2}(1-p_0)}{p_0\frac{1+\lambda}{2}+\frac{1-\lambda}{2}} = \frac{1-p_0-\frac{(1-\lambda)}{2}(1-p_0)}{p_0\frac{1+\lambda}{2}+\frac{1-\lambda}{2}} < \frac{\frac{1-\lambda}{2}p_0+(1-p_0)}{\frac{1+\lambda}{2}p_0}$$

if $0 < p_0 < 1$ and note that $\lambda < 1$ by assumption. We have $a^* < 0$ and thus $g_2(0) > g_1(0)$ (Bayes optimal decision for $x = 0$) if

$$1 > \frac{1-p_0}{p_0}\lambda,$$

and $b^* > 0$ and thus $g_1(\epsilon) > g_2(\epsilon)$ (Bayes optimal decision for $x = \epsilon$) if

$$1 > \frac{p_0}{1-p_0}\lambda.$$

Thus we recover the Bayes classifier if

$$\lambda < \min\left\{\frac{1-p_0}{p_0}, \frac{p_0}{1-p_0}\right\}.$$

□

C. Experiments

In the following, we provide additional details on our experimental setup regarding (a) the used attacks, especially our PGD-Conf attack including pseudo-code in Sec. C.1, (b) training of AT and CCAT in Sec. C.2, (c) the evaluated baselines in Sec. C.3 and (d) the used evaluation metrics in Sec. C.4. Afterwards, we include additional experimental results, including ablation studies in Sec. C.5, qualitative results for analysis in Sec. C.6, and further results for 95% and 98% true positive rate (TPR), results per attack and results per corruption on MNIST-C (Mu & Gilmer, 2019) and Cifar10-C (Hendrycks & Dietterich, 2019) in Sec. C.7.

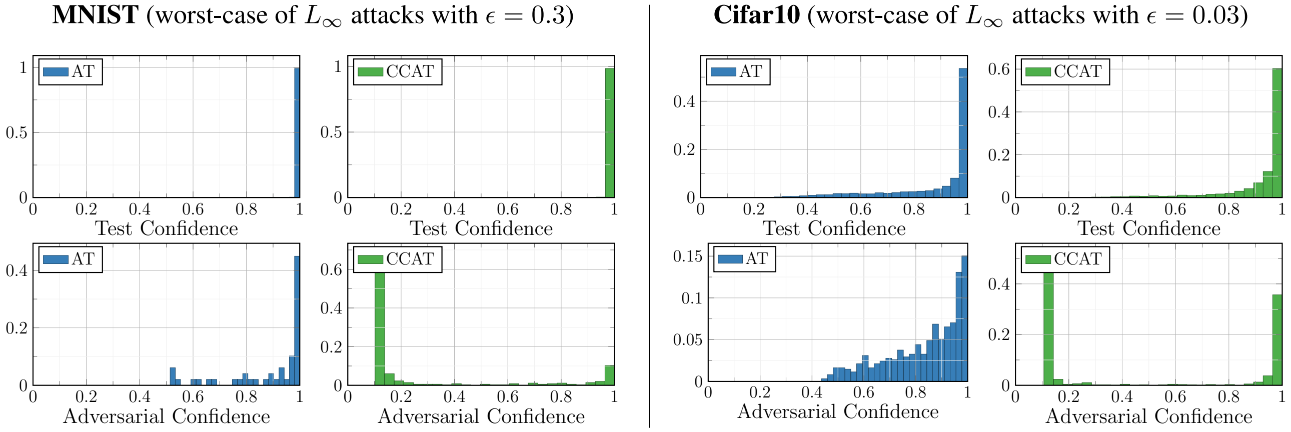

Figure 1: Confidence Histograms. We show histograms of confidences on correctly classified test examples (top) and on adversarial examples (bottom) for both AT and CCAT. Note that for AT, the number of successful adversarial examples is usually lower than for CCAT. For CCAT in contrast, nearly all adversarial examples are successful, while only a part has high confidence. Histograms obtained for the worst-case adversarial examples across all tested L_∞ attacks with $\epsilon = 0.3$ and $\epsilon = 0.03$ on MNIST and Cifar10, respectively.

C.1. Attacks

Projected Gradient Descent (PGD): Complementary to the description of the projected gradient descent (PGD) attack by (Madry et al., 2018) and our adapted attack, we provide a detailed algorithm in Alg. 1. We note that the objective maximized in (Madry et al., 2018) is

$$\mathcal{F}(x + \delta, y) = \mathcal{L}(f(x + \delta; w), y) \tag{3}$$

where $\mathcal{L}$ denotes the cross-entropy loss, $f(\cdot; w)$ denotes the model and (x, y) is an input-label pair from the test set. Our adapted attack, in contrast, maximizes

$$\mathcal{F}(x + \delta, y) = \max\nolimits_{k \neq y} f_k(x + \delta; w) \tag{4}$$

where f_k denotes the confidence of f in class k. Note that the maximum over labels, i.e., $\max_{k \neq y}$, is explicitly computed during optimization; this means that in contrast to (Goodfellow et al., 2019), we do not run $(K - 1)$ targeted attacks and subsequently take the maximum-confidence one, where K is the number of classes. We denote these two variants as PGD-CE and PGD-Conf, respectively. Deviating from (Madry et al., 2018), we initialize δ uniformly over directions and norm (instead of uniform initialization over the volume of the ϵ-ball):

$$\delta = u\epsilon \frac{\delta'}{\|\delta'\|_\infty}, \quad \delta' \sim \mathcal{N}(0, I), u \sim U(0, 1) \tag{5}$$

where δ' is sampled from a standard Gaussian and $u \in [0, 1]$ from a uniform distribution. We also consider zero initialization, i.e., $\delta = 0$. For random initialization we always consider multiple restarts, 10 for PGD-Conf and 50 for PGD-CE; for zero initialization, we use 1 restart. Finally, in contrast to (Madry et al., 2018), we run PGD for exactly T iterations, taking the perturbation corresponding to the best objective value obtained throughout the optimization.

PGD for $L_p, p \in \{0, 1, 2\}$: Both PGD-CE and PGD-Conf can also be applied using the L_2, L_1 and L_0 norms following the description above. Then, gradient normalization in Line 10 of Alg. 1, the projection in Line 6, and the initialization in Eq. (5) need to be adapted. For the L_2 norm, the gradient is normalized by dividing by the L_2 norm; for the L_1 norm only the 1% largest values (in absolute terms) of the gradient are kept and normalized by their L_1 norm; and for the L_0 norm, the gradient is normalized by dividing by the L_1 norm. We follow the algorithm of (Duchi et al., 2008) for the L_1 projection; for the L_0 projection (onto the ϵ-ball for $\epsilon \in \mathbb{N}_0$), only the ϵ largest values are kept. Similarly, initialization for L_2 and L_1 are simple by randomly choosing a direction (as in Eq. (5)) and then normalizing by their norm. For L_0, we randomly choose pixels with probability $(\frac{2}{3}\epsilon)/(HWD)$ and set them to a uniformly random values $u \in [0, 1]$, where $H \times W \times D$ is the image size. In experiments, we found that tuning the learning rate for PGD with L_1 and L_0 constraints (independent of

Algorithm 1 Projected Gradient Descent (PGD) with Backtracking. Pseudo-code for the used PGD procedure to maximize Eq. (3) or Eq. (4) using momentum and backtracking subject to the constraints $\tilde{x}_i = x_i + \delta_i \in [0,1]$ and $\|\delta\|_\infty \leq \epsilon$; in practice, the procedure is applied on batches of inputs. The algorithm is easily adapted to work with arbitrary L_p-norm; only the projections on Line 6 and 24 as well as the normalized gradient in Line 10 need to be adapted.

```
    input: example $x$ with label $y$
    input: number of iterations $T$
    input: learning rate $\gamma$, momentum $\beta$, learning rate factor $\alpha$
    input: initial $\delta^{(0)}$, e.g., Eq. (5) or $\delta^{(0)} = 0$
 1: $v := 0$ {saves the best objective achieved}
 2: $\tilde{x} := x + \delta^{(0)}$ {best adversarial example obtained}
 3: $g^{(-1)} := 0$ {accumulated gradients}
 4: for $t = 0, \ldots, T$ do
 5:   {projection onto $L_\infty$ $\epsilon$-ball and on $[0,1]$:}
 6:   clip $\delta_i^{(t)}$ to $[-\epsilon, \epsilon]$
 7:   clip $x_i + \delta_i^{(t)}$ to $[0,1]$
 8:   {forward and backward pass to get objective and gradient:}
 9:   $v^{(t)} := \mathcal{F}(x + \delta^{(t)}, y)$ {see Eq. (3) or Eq. (4)}
10:   $g^{(t)} := \text{sign}\left(\nabla_{\delta^{(t)}} \mathcal{F}(x + \delta^{(t)}, y)\right)$
11:   {keep track of adversarial example resulting in best objective:}
12:   if $v^{(t)} > v$ then
13:     $v := v^{(t)}$
14:     $\tilde{x} := x + \delta^{(t)}$
15:   end if
16:   {iteration $T$ is only meant to check whether last update improved objective:}
17:   if $t = T$ then
18:     break
19:   end if
20:   {integrate momentum term:}
21:   $g^{(t)} := \beta g^{(t-1)} + (1 - \beta) g^{(t)}$
22:   {“try” the update step and see if objective increases:}
23:   $\hat{\delta}^{(t)} := \delta^{(t)} + \gamma g^{(t)}$
24:   clip $\hat{\delta}_i^{(t)}$ to $[-\epsilon, \epsilon]$
25:   clip $x_i + \hat{\delta}_i^{(t)}$ to $[0,1]$
26:   $\hat{v}^{(t)} := \mathcal{F}(x + \hat{\delta}^{(t)}, y)$
27:   {only keep the update if the objective increased; otherwise decrease learning rate:}
28:   if $\hat{v}^{(t)} \geq v^{(t)}$ then
29:     $\delta^{(t+1)} := \hat{\delta}^{(t)}$
30:   else
31:     $\gamma := \gamma/\alpha$
32:   end if
33: end for
34: return $\tilde{x}$, $\tilde{v}$
```

the objective, i.e., Eq. (3) or Eq. (4)) is much more difficult. Additionally, PGD using the L_0 norm seems to get easily stuck in sub-optimal local optima.

Backtracking: Alg. 1 also gives more details on the employed momentum and backtracking scheme. These two “tricks” add two additional hyper-parameters to the number of iterations T and the learning rate γ, namely the momentum parameter β and the learning rate factor α. After each iteration, the computed update, already including the momentum term, is only applied if this improves the objective. This is checked through an additional forward pass. If not, the learning rate is divided by α, and the update is rejected. Alg. 1 includes this scheme as an algorithm for an individual test example x with label y for brevity; however, extending it to work on batches is straight-forward. However, it is important to note that the learning rate is updated per test example individually. In practice, for PGD-CE, with $T = 200$ iterations, we use $\gamma = 0.05$, $\beta = 0.9$ and $\alpha = 1.25$; for PGD-Conf, with $T = 1000$ iterations, we use $\gamma = 0.001$, $\beta = 0.9$ and $\alpha = 1.1$.

Black-Box Attacks: We also give more details on the used black-box attacks. For random sampling, we apply Eq. (5) $T = 5000$ times. We also implemented the Query-Limited (QL) black-box attack of (Ilyas et al., 2018) using a population

MNIST: Attack Ablation with confidence-thresholded RErr in % for τ@99%TPR (L_∞ attack with $\epsilon = 0.3$ for training and testing)									
Optimization	momentum+backtrack					mom		–	
Initialization	zero				rand	zero		zero	
Iterations	40	200	1000	2000	2000	60	300	60	300
AT	0.4	0.4	0.4	0.3	0.6	0.4	0.6	0.4	0.4
AT Conf (AT trained with PGD-Conf)	0.8	0.8	0.8	0.8	1.1	1.0	1.1	1.0	1.0
CCAT	0.2	1.4	4.6	4.6	3.7	0.7	0.7	0.2	0.2

SVHN: Attack Ablation with confidence-thresholded RErr in % for τ@99%TPR (L_∞ attack with $\epsilon = 0.03$ for training and testing)									
Optimization	momentum+backtrack					mom		–	
Initialization	zero				rand	zero		zero	
Iterations	40	200	1000	2000	2000	60	300	60	300
AT	38.4	46.2	49.9	50.1	51.8	37.7	38.1	29.9	30.8
AT Conf (AT trained with PGD-Conf)	27.4	40.5	46.9	47.3	48.1	27.1	28.5	21.1	23.8
CCAT	4.0	5.0	22.8	23.3	5.2	2.6	2.6	2.6	2.6

CIFAR10: Attack Ablation with confidence-thresholded RErr in % for τ@99%TPR (L_∞ attack with $\epsilon = 0.03$ for training and testing)									
Optimization	momentum+backtrack					mom		–	
Initialization	zero				rand	zero		zero	
Iterations	40	200	1000	2000	2000	60	300	60	300
AT	60.9	60.8	60.8	60.8	60.9	60.9	60.9	57.4	57.6
AT Conf (AT trained with PGD-Conf)	60.4	60.6	60.5	60.5	60.9	60.4	60.6	56.2	56.6
CCAT	14.8	16.2	40.2	41.3	34.9	7.2	7.2	7.2	7.2

Table 1: Detailed Attack Ablation Studies. We compare our L_∞ PGD-Conf attack with T iterations and different combinations of momentum, backtracking and initialization on all three datasets. We consider AT, AT trained with PGD-Conf (AT Conf), and CCAT; we report RErr for confidence threshold τ@99%TPR. As backtracking requires an additional forward pass per iteration, we use $T = 60$ and $T = 300$ for attacks without backtracking to be comparable to attacks with $T = 40$ and $T = 200$ with backtracking. Against CCAT, $T = 1000$ iterations or more are required and backtracking is essential to achieve high RErr. AT, in contrast, is "easier" to attack, requiring less iterations and less sophisticated optimization (i.e., without momentum and/or backtracking).

of 50 and variance of 0.1 for estimating the gradient in Line 10 of Alg. 1; a detailed algorithm is provided in (Ilyas et al., 2018). We use a learning rate of 0.001 (note that the gradient is signed, as in (Madry et al., 2018)) and also integrated a momentum with $\beta = 0.9$ and backtracking with $\alpha = 1.1$ and $T = 1000$ iterations. We use zero and random initialization; in the latter case we allow 10 random restarts. For the Simple black-box attack we follow the algorithmic description in (Narodytska & Kasiviswanathan, 2017) considering only axis-aligned perturbations of size ϵ per pixel. We run the attack for $T = 1000$ iterations and allow 10 random restarts. Following, (Khoury & Hadfield-Menell, 2018), we further use the Geometry attack for $T = 1000$ iterations. Random sampling, QL, Simple and Geometry attacks are run for arbitrary L_p, $p \in \{\infty, 2, 1, 0\}$. For L_∞, we also use the Square attack proposed in (Andriushchenko et al., 2019) with $T = 5000$ iterations with a probability of change of 0.05. For all attacks, we use Eq. (4) as objective. Finally, for L_0, we also use Corner Search (Croce & Hein, 2019) with the cross-entropy loss as objective, for $T = 200$ iterations. We emphasize that, except for QL, these attacks are not gradient-based and do not approximate the gradient. Furthermore, we note that all attacks except Corner Search are adapted to explicitly attack CCAT by maximizing Eq. (4).

C.2. Training

We follow the ResNet-20 architecture by (He et al., 2016) implemented in PyTorch (Paszke et al., 2017). For training we use a batch size of 100 and train for 100 and 200 epochs on MNIST and SVHN/Cifar10, respectively: this holds for normal training, adversarial training (AT) and confidence-calibrated adversarial training (CCAT). For the latter two, we use PGD-CE and PGD Conf, respectively, for $T = 40$ iterations including momentum and backtracking ($\beta = 0.9$, $\alpha = 1.5$). For PGD-CE we use a learning rate of 0.05, 0.01 and 0.005 on MNIST, SVHN and Cifar10. For PGD-Conf we use a learning rate of 0.005. For CCAT, we randomly switch between the initialization in Eq. (5) and zero initialization. For training, we use standard stochastic gradient descent, starting with a learning rate of 0.1 on MNIST/SVHN and 0.075 on

SVHN: Training Ablation for Detection (τ@99%TPR) and Standard Settings ($\tau = 0$)
(L_∞ attack with $\epsilon = 0.03$ during training and testing)

	Detection Setting τ@99%TPR			Standard Setting $\tau = 0$	
	ROC AUC	Err in %	RErr in %	Err in %	RErr in %
Normal	0.17	2.6	99.9	3.6	99.9
AT	0.55	2.5	54.9	3.4	56.9
AT Conf (AT trained with PGD-Conf)	0.61	2.8	52.5	3.7	58.7
CCAT, $\rho = 1$	0.74	2.2	43.0	2.7	82.4
CCAT, $\rho = 2$	0.68	2.1	44.2	2.9	79.6
CCAT, $\rho = 4$	0.68	1.8	35.8	2.7	80.4
CCAT, $\rho = 6$	0.64	1.8	32.8	2.9	72.1
CCAT, $\rho = 8$	0.63	2.2	42.3	2.9	84.6
CCAT, $\rho = 10$	0.67	2.1	38.5	2.9	91.0
CCAT, $\rho = 12$	0.67	1.9	36.3	2.8	81.8

CIFAR10: Training Ablation for Detection (τ@99%TPR) and Standard Settings ($\tau = 0$)
(L_∞ attack with $\epsilon = 0.03$ during training and testing)

	Detection Setting τ@99%TPR			Standard Setting $\tau = 0$	
	ROC AUC	Err in %	RErr in %	Err in %	RErr in %
Normal	0.20	7.4	100.0	8.3	100.0
AT	0.65	15.1	60.9	16.6	61.3
AT Conf (AT trained with PGD-Conf)	0.63	15.1	61.5	16.1	61.7
CCAT, $\rho = 1$	0.63	8.7	72.4	9.7	95.3
CCAT, $\rho = 2$	0.60	8.4	70.6	9.7	95.1
CCAT, $\rho = 4$	0.61	8.6	66.3	9.8	93.5
CCAT, $\rho = 6$	0.54	8.0	69.8	9.2	94.1
CCAT, $\rho = 8$	0.58	8.5	65.3	9.4	93.2
CCAT, $\rho = 10$	0.60	8.7	63.0	10.1	95.0
CCAT, $\rho = 12$	0.62	9.4	63.0	10.1	96.6

Table 2: Training Ablation Studies. We report unthresholded RErr and Err, i.e., $\tau = 0$ ("Standard Setting"), and τ@99%TPR as well as ROC AUC ("Detection Setting") for CCAT with various values for ρ. The models are tested against our L_∞ PGD-Conf attack with $T = 1000$ iterations and zero as well as random initialization. On Cifar10, $\rho = 10$ works best and performance stagnates for $\rho > 10$. On SVHN, we also use $\rho = 10$, although $\rho = 6$ shows better results.

Cifar10. The learning rate is multiplied by 0.95 after each epoch. We do not use weight decay; but the network includes batch normalization (Ioffe & Szegedy, 2015). On SVHN and Cifar10, we use random cropping, random flipping (only Cifar10) and contrast augmentation during training. We always train on 50% clean and 50% adversarial examples per batch, i.e., each batch contains both clean and adversarial examples which is important when using batch normalization.

C.3. Baselines

As baseline, we use the multi-steepest descent (MSD) adversarial training of (Maini et al., 2020), using the code and models provided in the official repository[1]. The models correspond to a LeNet-like (LeCun et al., 1998) architecture on MNIST, and the pre-activation version of ResNet-18 (He et al., 2016) on Cifar10. The models were trained with L_∞, L_2 and L_1 adversarial examples and ϵ set to $0.3, 1.5, 12$ and $0.03, 0.5, 12$, respectively. We attacked these models using the same setup as used for standard AT and our CCAT.

Additionally, we compare to TRADES (Zhang et al., 2019) using the code and pre-trained models from the official repository[2]. The models correspond to a convolutional architecture with four convolutional and three fully-connected layers (Carlini & Wagner, 2017) on MNIST, and a wide ResNet, specifically WRN-10-28 (Zagoruyko & Komodakis, 2016), on

[1] https://github.com/locuslab/robust_union
[2] https://github.com/yaodongyu/TRADES

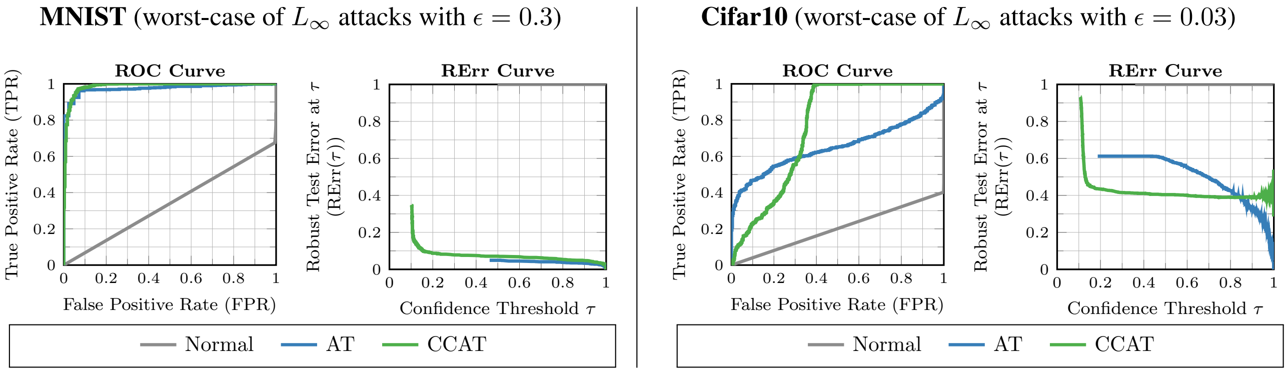

Figure 2: ROC and RErr curves. ROC curves, i.e. FPR plotted against TPR for all possible confidence thresholds τ, and (confidence-thresholded) RErr curves, i.e., RErr over confidence threshold τ for AT and CCAT, including different ρ parameters. Worst-case adversarial examples across all L_∞ attacks with $\epsilon = 0.3$ (MNIST) and $\epsilon = 0.03$ (Cifar10) were tested. For evaluation, the confidence threshold τ is fixed at 99%TPR, allowing to reject at most 1% correctly classified clean examples. Thus, we also do not report the area under the ROC curve in the main paper.

Cifar10. Both are trained using *only* L_∞ adversarial examples with $\epsilon = 0.3$ and $\epsilon = 0.03$, respectively. The evaluation protocol follows the same setup as used for standard AT and CCAT.

On Cifar10, we also use the pre-trained ResNet-50 from (Madry et al., 2018) obtained from the official repository[3]. The model was trained on L_∞ adversarial examples with $\epsilon = 0.03$. The same evaluation as for CCAT applies.

Furthermore, we evaluate two detection baseline: the Mahalanobis detector (MAHA) of (Ma et al., 2018) and the local intrinsic dimensionality (LID) detector of (Lee et al., 2018). We used the code provided by (Lee et al., 2018) from the official repository[4]. For evaluation, we used the provided setup, adding *only* PGD-CE and PGD-Conf with $T = 1000$, $T = 200$ and $T = 40$. For $T = 1000$, we used 5 random restarts, for $T = 200$, we used 25 restarts, and for $T = 40$, we used one restart. These were run for L_∞, L_2, L_1 and L_0. We also evaluated distal adversarial examples as in the main paper. While the hyper-parameters were chosen considering our L_∞ PGD-CE attack ($T = 40$, one restart) and kept fixed for other threat models, the logistic regression classifier trained on the computed statistics (e.g., the Mahalanobis statistics) is trained for each threat model individually, resulting in an advantage over AT and CCAT. For worst-case evaluation, where we keep the highest-confidence adversarial example per test example for CCAT, we use the obtained detection score instead. This means, for each test example individually, we consider the adversarial example with worst detection score for evaluation.

C.4. Evaluation Metrics

Complementing the discussion in the main paper, we describe the used evaluation metrics and evaluation procedure in more detail. Adversarial examples are computed on the first 1000 examples of the test set; the used confidence threshold is computed on the last 1000 examples of the test set; test errors are computed on all test examples minus the last 1000. As we consider multiple attacks, and some attacks allow multiple random restarts, we always consider the worst case adversarial example per test example and across all attacks/restarts; the worst-case is selected based on confidence.

FPR and ROC AUC: To compute receiver operating characteristic (ROC) curves, and the area under the curve, i.e., ROC AUC, we define negatives as *successful* adversarial examples (corresponding to correctly classified test examples) and positives as the corresponding *correctly classified* test examples. The ROC AUC as well as the curve itself can easily be calculated using scikit-learn (Pedregosa et al., 2011). Practically, the generated curve could be used to directly estimate a threshold corresponding to a pre-determined true positive rate (TPR). However, this requires interpolation; after trying several interpolation schemes, we concluded that the results are distorted significantly, especially for TPRs close to 100%. Thus, we follow a simpler scheme: on a held out validation set of size 1000 (the last 1000 samples of the test set), we sorted the corresponding confidences, and picked the confidence threshold in order to obtain (at least) the desired TPR, e.g., 99%.

In the main paper, instead of reporting ROC AUC, we reported only confidence-thresholded robust test error (RErr), which

[3] https://github.com/MadryLab/robustness
[4] https://github.com/pokaxpoka/deep_Mahalanobis_detector

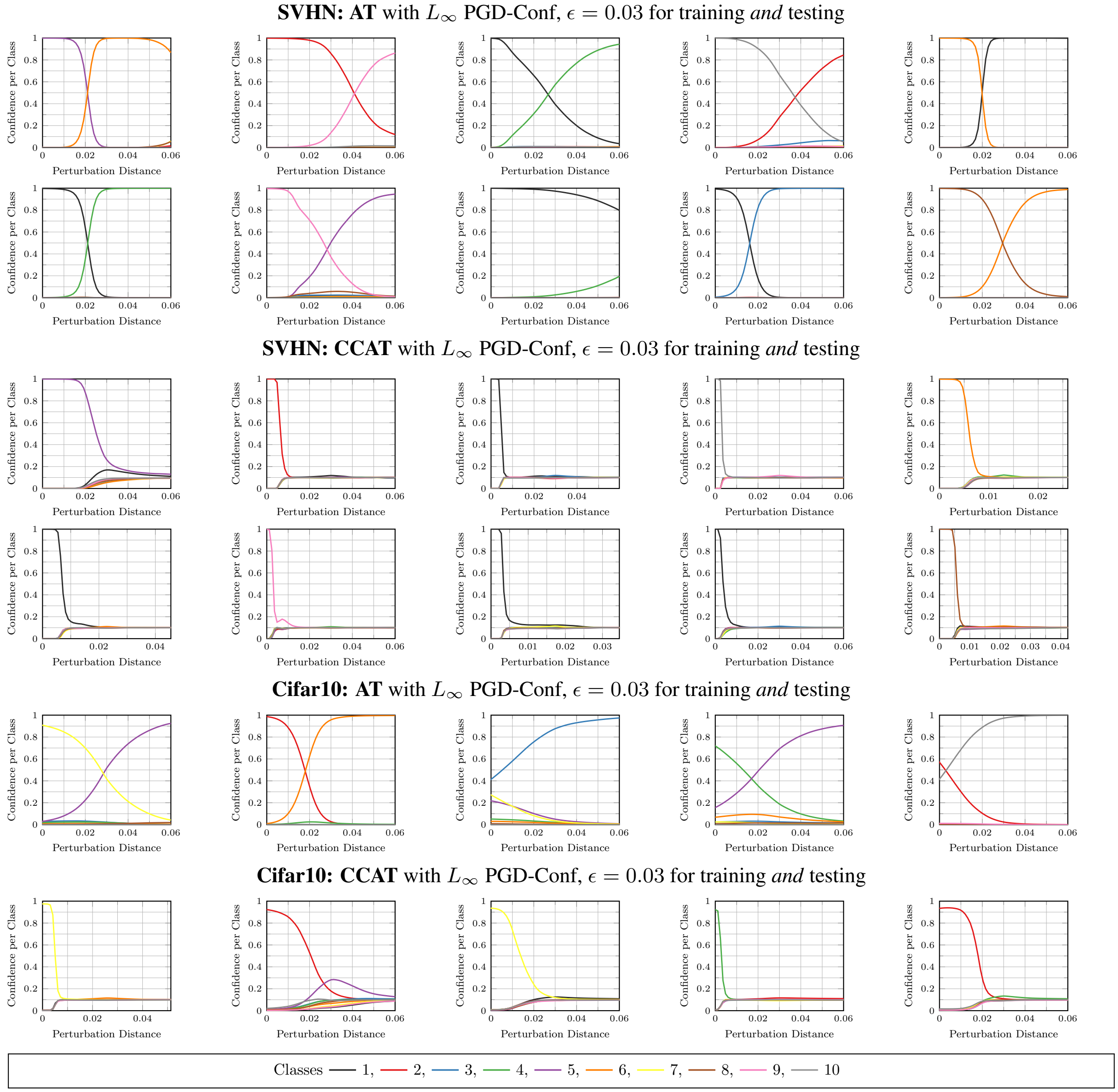

Figure 3: Effect of Confidence Calibration. Confidences for classes along adversarial directions for AT and CCAT. Adversarial examples were computed using PGD-Conf with $T = 1000$ iterations and zero initialization. For both AT and CCAT, we show the first ten examples of the test set on SVHN, and the first five examples of the test set on Cifar10. As can be seen, CCAT biases the network to predict uniform distributions beyond the ϵ-ball used during training ($\epsilon = 0.03$). For AT, in contrast, adversarial examples can usually be found right beyond the ϵ-ball.

implicitly subsumes the false positive rate (FPR), at a confidence threshold of 99%TPR. Again, we note that this is an extremely conservative choice, allowing to reject at most 1% correctly classified clean examples. In addition, comparison to other approaches is fair as the corresponding confidence threshold only depends on correctly classified clean examples, not on adversarial examples. As also seen in Fig. 2, ROC AUC is not a practical metric to evaluate the detection/rejection of adversarial examples. This is because rejecting a significant part of correctly classified clean examples is not acceptable. In this sense, ROC AUC measures how well positives and negatives can be distinguished in general, while we are only interested in the performance for very high TPR, e.g., 99%TPR as in the main paper. In this document, we also report FPR to complement our evaluation using (confidence-thresholded) RErr.

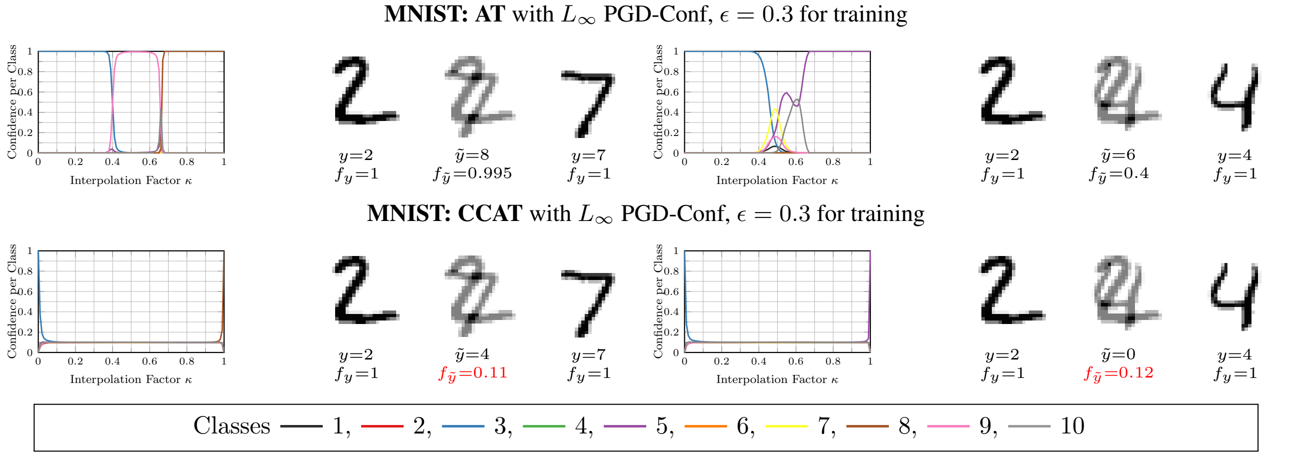

Figure 4: Confidence Calibration Between Test Examples. We plot the confidence for all classes when interpolating linearly between test examples: $(1 - \kappa)x_1 + \kappa x_2$ for two test examples x_1 and x_2 with $\kappa \in [0, 1]$; x_1 is fixed and we show two examples corresponding to different x_2. Additionally, we show the corresponding images for $\kappa = 0$, i.e., x_1, $\kappa = 0.5$, i.e., the mean image, and $\kappa = 1$, i.e., x_2, with the corresponding labels and confidences. As can be seen, CCAT is able to perfectly predict a uniform distribution between test examples. AT, in contrast, enforces high-confidence predictions, resulting in conflicts if x_1 and x_2 are too close together (i.e., within one ϵ-ball) or in sudden changes of the predicted class in between, as seen above.

Robust Test Error: The standard robust test error (Madry et al., 2018) is the model's test error in the case where all test examples are allowed to be attacked, i.e., modified within the chosen threat model, e.g., for L_p:

$$\text{“Standard” RErr} = \frac{1}{N}\sum_{n=1}^{N} \max_{\|\delta\|_p \le \epsilon} \mathbb{1}_{f(x_n+\delta)\neq y_n} \tag{6}$$

where $\{(x_n, y_n)\}_{n=1}^N$ are test examples and labels. In practice, RErr is computed empirically using several adversarial attacks, potentially with multiple restarts as the inner maximization problem is generally non-convex.

As standard RErr does not account for a reject option, we propose a generalized definition adapted to our confidence-thresholded setting. For fixed confidence threshold τ, e.g., at 99%TPR, the confidence-thresholded RErr is defined as

$$\text{RErr}(\tau) = \frac{\sum\limits_{n=1}^{N} \max\limits_{\|\delta\|_p \le \epsilon, c(x_n+\delta)\ge\tau} \mathbb{1}_{f(x_n+\delta)\neq y_n}}{\sum\limits_{n=1}^{N} \max\limits_{\|\delta\|_p \le \epsilon} \mathbb{1}_{c(x_n+\delta)\ge\tau}} \tag{7}$$

with $c(x) = \max_k f_k(x)$ and $f(x)$ being the model's confidence and predicted class on example x, respectively. This is the test error on test examples that can be modified within the chosen threat model *and* pass confidence thresholding. It reduces to standard RErr for $\tau = 0$, is in $[0, 1]$ and, thus, fully comparable to related work.

As both Eq. (6) and Eq. (7) cannot be computed exactly, we compute

$$\frac{\sum_{n=1}^{N} \max\{\mathbb{1}_{f(x_n)\neq y_n}\mathbb{1}_{c(x_n)\ge\tau}, \mathbb{1}_{f(\tilde{x}_n)\neq y_n}\mathbb{1}_{c(\tilde{x}_n)\ge\tau}\}}{\sum_{n=1}^{N} \max\{\mathbb{1}_{c(x_n)\ge\tau}, \mathbb{1}_{c(\tilde{x}_n)\ge\tau}\}} \tag{8}$$

which is an upper bound assuming that our attack is perfect. Essentially, this counts the test examples x_n that are either classified incorrectly with confidence $c(x_n) \ge \tau$ or that can be attacked successfully $\tilde{x}_n = x_n + \delta$ with confidence $c(\tilde{x}_n) \ge \tau$. This is normalized by the total number of test examples x_n that have $c(x_n) \ge \tau$ or where the corresponding adversarial example $\tilde{x}_n$ has $c(\tilde{x}_n) \ge \tau$. It can easily be seen that $\tau = 0$ reduces Eq. (8) to its unthresholded variant, i.e., standard RErr, ensuring full comparability to related work.

MNIST: FPR and **RErr** in % for τ@99%TPR

	L_∞ $\epsilon = 0.3$		L_∞ $\epsilon = 0.4$		L_2 $\epsilon = 3$		L_1 $\epsilon = 18$		L_0 $\epsilon = 15$		adv. frames	
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**	
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓
Normal	99.3	100.0	99.3	100.0	99.3	100.0	99.3	100.0	91.6	92.3	87.0	87.7
AT-50%	1.0	1.7	99.3	100.0	80.7	81.5	23.8	24.6	23.0	23.9	72.9	73.7
AT-100%	1.0	1.7	99.2	100.0	83.9	84.8	20.2	21.3	12.9	13.9	61.3	62.3
CCAT	6.9	7.4	11.4	11.9	0.0	0.3	1.3	1.8	14.2	14.8	0.0	0.2
* MSD	32.1	34.3	96.7	98.9	57.0	59.2	53.7	55.9	64.2	66.4	6.6	8.8
* TRADES	3.4	4.0	99.3	99.9	43.6	44.3	8.2	9.0	34.6	35.5	0.0	0.2

SVHN: FPR and **RErr** in % for τ@99%TPR

	L_∞ $\epsilon = 0.03$		L_∞ $\epsilon = 0.06$		L_2 $\epsilon = 2$		L_1 $\epsilon = 24$		L_0 $\epsilon = 10$		adv. frames	
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**	
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓
Normal	95.8	99.9	95.9	100.0	95.9	100.0	95.9	100.0	79.6	83.7	74.6	78.7
AT-50%	52.3	56.0	84.7	88.4	95.7	99.4	95.8	99.5	70.0	73.6	30.0	33.6
AT-100%	42.1	48.3	80.9	87.1	93.3	99.5	93.6	99.8	83.1	89.4	19.9	26.0
CCAT	35.5	39.1	49.5	53.1	25.4	29.0	28.1	31.7	0.4	3.5	1.0	3.7
* LID	87.1	91	89.2	93.1	96.1	92.2	85.7	90	37.6	41.6	85.1	89.8
* MAHA	68.6	73	75.2	79.5	73.2	78.1	63.2	67.5	36.9	41.5	6.3	9.9

CIFAR10: FPR and **RErr** in % for τ@99%TPR

	L_∞ $\epsilon = 0.03$		L_∞ $\epsilon = 0.06$		L_2 $\epsilon = 2$		L_1 $\epsilon = 24$		L_0 $\epsilon = 10$		adv. frames	
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**	
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓
Normal	93.0	100.0	93.0	100.0	93.0	100.0	93.0	100.0	77.7	84.7	89.7	96.7
AT-50%	47.6	62.7	78.6	93.7	83.3	98.4	83.3	98.4	59.3	74.4	63.6	78.7
AT-100%	42.3	59.9	72.7	90.3	80.7	98.3	80.4	98.0	54.7	72.3	62.0	79.6
CCAT	59.9	68.4	83.9	92.4	43.7	52.2	50.3	58.8	14.4	23.0	57.4	66.1
* MSD	35.3	53.2	71.5	89.4	70.6	88.5	50.7	68.6	21.4	39.2	64.7	82.6
* TRADES	28.9	43.5	66.4	81.0	56.3	70.9	82.3	96.9	22.3	36.9	57.5	72.1
* AT-Madry	33.8	45.1	73.2	84.5	87.4	98.7	86.5	97.8	31.0	42.3	62.0	73.3
* LID	92.7	99	92.9	99.2	64	70.6	82.9	89.4	40.6	47	59.9	66.1
* MAHA	87.7	94.1	89	95.3	84.2	90.6	91.3	97.6	43.5	49.8	64.1	70

Table 3: Main Results: FPR for 99%**TPR.** For **99**%TPR, we report confidence-thresholded RErr *and* FPR for the results from the main paper. We emphasize that only PGD-CE and PGD-Conf were used against LID and MAHA. In general, the observations of the main paper can be confirmed considering FPR. Due to the poor Err of AT, MSD or TRADES on Cifar10, these methods benefit most from considering FPR instead of (confidence-thresholded) RErr. * Pre-trained models with different architectures.

In the following, we also highlight two special cases that are (correctly) taken into account by Eq. (8): (a) if a correctly classified test example x_n, i.e., $f(x_n) = y_n$, has confidence $c(x_n) < \tau$, i.e., is rejected, but the corresponding adversarial example $\tilde{x}_n$ with $f(\tilde{x}_n) \neq y$ has $c(\tilde{x}_n) \geq \tau$, i.e., is *not* rejected, this is counted both in the numerator and denominator; (b) if an incorrectly classified test example x_n, i.e., $f(x_n) \neq y$, with $c(x_n) < \tau$, i.e., rejected, has a corresponding adversarial example $\tilde{x}_n$, i.e., also $f(\tilde{x}_n) \neq y$, but with $x(x_n) \geq \tau$, i.e., *not* rejected, this is also counted in the numerator as well as denominator. Note that these cases are handled differently in our detection evaluation following related work (Ma et al., 2018; Lee et al., 2018): negatives are adversarial examples corresponding to correctly classified clean examples that are successful, i.e., change the label. For example, case (b) would not contribute towards the FPR since the original test example is already mis-classified. Thus, while RErr implicitly includes FPR as well as Err, it is even more conservative than just considering "FPR + Err".

	MNIST: FPR and confidence-thresholded **RErr** in % for τ@98%TPR													
	L_∞ $\epsilon = 0.3$		L_∞ $\epsilon = 0.4$		L_2 $\epsilon = 3$		L_1 $\epsilon = 18$		L_0 $\epsilon = 15$		adv. frames		distal	corr. MNIST-C
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**	**unseen**
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	Err ↓
Normal	99.3	100.0	99.3	100.0	99.3	100.0	99.3	100.0	87.3	88.1	79.8	80.5	100.0	31.0
AT-50%	0.5	0.8	99.3	100.0	66.7	67.9	16.3	17.5	16.1	17.2	61.3	62.5	100.0	12.3
AT-100%	0.6	1.3	99.2	100.0	77.3	78.3	16.9	18.0	9.2	10.2	52.6	53.7	100.0	15.4
CCAT	5.2	5.7	8.8	9.3	0.0	0.2	0.6	0.9	7.6	8.1	0.0	0.1	0.0	5.3
* MSD	28.8	31.0	96.6	98.8	53.9	56.2	51.3	53.5	61.5	63.7	4.4	6.6	100.0	5.6
* TRADES	1.2	1.9	99.1	99.7	31.6	32.6	4.3	5.1	28.0	29.7	0.0	0.1	100.0	5.7

	SVHN: FPR and confidence-thresholded **RErr** in % for τ@98%TPR												
	L_∞ $\epsilon = 0.03$		L_∞ $\epsilon = 0.06$		L_2 $\epsilon = 2$		L_1 $\epsilon = 24$		L_0 $\epsilon = 10$		adv. frames		distal
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓
Normal	95.7	99.8	95.9	100.0	95.9	100.0	95.9	100.0	72.7	76.8	72.4	76.5	87.1
AT-50%	50.0	53.7	83.4	87.1	95.5	99.2	95.7	99.4	54.0	57.9	26.8	30.6	86.3
AT-100%	42.1	48.3	80.9	87.1	93.3	99.5	93.6	99.8	82.2	88.8	19.3	25.1	81.0
CCAT	34.0	37.6	40.5	44.1	20.3	23.9	25.0	28.6	0.2	2.6	0.1	2.2	0.0

	CIFAR10: FPR and confidence-thresholded **RErr** in % for τ@98%TPR													
	L_∞ $\epsilon = 0.03$		L_∞ $\epsilon = 0.06$		L_2 $\epsilon = 2$		L_1 $\epsilon = 24$		L_0 $\epsilon = 10$		adv. frames		corr.	corr. CIFAR10-C
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**	**unseen**
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	Err ↓
Normal	93.0	100.0	93.0	100.0	93.0	100.0	93.0	100.0	70.1	77.1	89.6	96.6	83.3	11.4
AT-50%	47.6	62.7	78.6	93.7	83.3	98.4	83.3	98.4	57.2	72.4	63.6	78.7	75.0	15.1
AT-100%	42.1	59.7	72.7	90.3	80.7	98.3	80.4	98.0	52.1	70.0	62.0	79.6	72.5	17.8
CCAT	59.4	67.9	83.5	92.0	43.3	51.8	50.0	58.5	11.7	20.3	56.4	65.1	0.0	8.1
* MSD	35.1	53.0	71.5	89.4	69.9	87.8	50.6	68.5	17.8	35.8	64.7	82.6	76.7	17.1
* TRADES	28.9	43.5	66.4	81.0	56.2	70.8	82.3	96.9	21.9	36.4	57.4	72.0	76.2	14.1
* AT-Madry	33.6	44.9	73.2	84.5	87.4	98.7	86.5	97.8	30.8	42.0	61.9	73.2	78.5	11.4

Table 4: Main Results: Generalizable Robustness for 98%TPR. While reporting results for 99%TPR in the main paper, reducing the TPR requirement for confidence-thresholding to 98%TPR generally improves results, but only slightly. We report FPR and confidence-thresholded RErr for 98%TPR. For MNIST-C and Cifar10-C, we report mean Err across all corruptions. L_∞ attacks with ϵ=0.3 on MNIST and $\epsilon = 0.03$ on SVHN/Cifar10 were used for training (**seen**). All other attacks were not used during training (**unseen**). * Pre-trained models with different architectures.

C.5. Ablation Study

In the following, we include ablation studies for our attack PGD-Conf, in Tab. 1, and for CCAT, in Tab. 2.

Attack. Regarding the proposed attack PGD-Conf using momentum and backtracking, Tab. 1 shows that backtracking and sufficient iterations are essential to attack CCAT. On SVHN, for AT, the difference in RErr between $T = 200$ and $T = 1000$ iterations is only 3.7%, specifically, 46.2% and 49.9%. For CCAT, in contrast, using $T = 200$ iterations is not sufficient, with merely 5% RErr. However, $T = 1000$ iterations with zero initialization increases RErr to 22.8%. For more iterations, i.e., $T = 2000$, RErr stagnates with 23.3%. When using random initialization (one restart), RErr drops to 5.2%, even when using $T = 2000$ iterations. Similar significant drops are observed without backtracking. These observations generalize to MNIST and Cifar10.

Training: Tab. 2 reports results for CCAT with different values for ρ. We note that ρ controls the (speed of the) transition from (correct) one-hot distribution to uniform distribution depending on the distance of adversarial example to the corresponding original training example. Here, higher ρ results in a sharper (i.e., faster) transition from one-hot to uniform distribution. It is also important to note that the power transition does not preserve a bias towards the true label, i.e., for the maximum possible perturbation ($\|\delta\|_\infty = \epsilon$), the network is forced to predict a purely uniform distribution. As can be seen,

	MNIST: FPR and confidence-thresholded **RErr** in % for τ@**95**%TPR													
	L_∞ $\epsilon = 0.3$		L_∞ $\epsilon = 0.4$		L_2 $\epsilon = 3$		L_1 $\epsilon = 18$		L_0 $\epsilon = 15$		adv. frames		distal	corr. MNIST-C
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**	**unseen**
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	Err ↓
Normal	99.3	100.0	99.3	100.0	99.3	100.0	99.0	99.7	75.6	76.7	65.9	67.0	100.0	27.5
AT-50%	0.2	0.4	99.3	100.0	54.6	56.7	12.3	13.7	11.1	12.3	47.3	49.2	100.0	8.6
AT-100%	0.2	0.9	99.2	100.0	63.9	65.6	10.7	12.4	3.6	4.4	35.0	37.2	100.0	10.5
CCAT	3.1	3.7	5.7	6.3	0.0	0.1	0.0	0.1	2.0	2.2	0.0	0.1	0.0	5.5
* MSD	22.0	24.6	95.0	97.2	44.0	46.8	42.1	44.6	54.5	57.3	2.3	4.4	100.0	4.4
* TRADES	0.6	1.0	98.1	98.8	16.5	18.3	2.1	2.7	21.4	24.0	0.0	0.0	100.0	3.2

	SVHN: FPR and confidence-thresholded **RErr** in % for τ@**95**%TPR												
	L_∞ $\epsilon = 0.03$		L_∞ $\epsilon = 0.06$		L_2 $\epsilon = 2$		L_1 $\epsilon = 24$		L_0 $\epsilon = 10$		adv. frames		distal
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓
Normal	95.6	99.7	95.9	100.0	95.9	100.0	95.9	100.0	65.3	69.5	66.8	71.1	87.1
AT-50%	45.5	49.2	79.7	83.4	95.4	99.1	95.5	99.2	34.8	38.9	21.5	25.5	59.3
AT-100%	40.5	46.9	80.9	87.1	93.3	99.5	93.6	99.8	78.5	85.5	15.9	21.7	75.1
CCAT	32.8	36.5	38.6	42.2	16.5	20.3	21.1	24.9	0.0	1.2	0.0	1.2	0.0

	CIFAR10: FPR and confidence-thresholded **RErr** in % for τ@**95**%TPR													
	L_∞ $\epsilon = 0.03$		L_∞ $\epsilon = 0.06$		L_2 $\epsilon = 2$		L_1 $\epsilon = 24$		L_0 $\epsilon = 10$		adv. frames		corr.	corr. CIFAR10-C
	seen		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**		**unseen**	**unseen**
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	Err ↓
Normal	93.0	100.0	93.0	100.0	93.0	100.0	93.0	100.0	52.0	59.0	88.8	95.9	83.3	7.7
AT-50%	46.6	61.7	78.6	93.7	83.3	98.4	83.3	98.4	43.9	59.6	62.8	77.9	75.0	12.1
AT-100%	41.3	59.1	72.7	90.3	80.6	98.2	80.4	98.0	47.2	65.3	62.0	79.7	72.5	15.6
CCAT	57.4	66.0	80.8	89.3	39.7	48.2	48.9	57.4	3.6	11.1	50.8	59.7	0.0	6.0
* MSD	32.8	50.9	71.5	89.4	67.7	85.6	49.1	67.3	11.4	28.4	64.2	82.2	76.7	13.7
* TRADES	26.5	41.3	66.1	80.7	53.9	68.5	82.3	96.9	17.6	32.0	56.4	71.1	76.2	10.7
* AT-Madry	32.4	43.9	73.2	84.5	87.4	98.7	86.5	97.8	28.6	40.1	61.5	72.8	78.5	9.1

Table 5: Main Results: Generalizable Robustness for 95%TPR. We report FPR and RErr for **95**%TPR, in comparison with 98% in Tab. 4 and 99% in the main paper. For MNIST-C and Cifar10-C, we report mean Err across all corruptions. L_∞ attacks with ϵ=0.3 on MNIST and $\epsilon = 0.03$ on SVHN/Cifar10 **seen** during training; all other attacks **unseen** during training. Results improve slightly in comparison with 98%TPR. However, the improvements are rather small and do not justify the significantly increased fraction of "thrown away" (correctly classified) clean examples. * Pre-trained models with different architectures.

both on SVHN and Cifar10, higher ρ usually results in better robustness. Thus, for the main paper, we chose $\rho = 10$. Only on SVHN, $\rho = 6$ of $\rho = 12$ perform slightly better. However, we found that $\rho = 10$ generalizes better to previously unseen attacks.

C.6. Analysis

Confidence Histograms: For further analysis, Fig. 1 shows confidence histograms for AT and CCAT on MNIST and Cifar10. The confidence histograms for CCAT reflect the expected behavior: adversarial examples are mostly successful in changing the label, which is supported by high RErr values for confidence threshold $\tau = 0$, but their confidence is pushed towards uniform distributions. For AT, in contrast, successful adversarial examples – fewer in total – generally obtain high confidence. As a result, while confidence thresholding generally benefits AT, the improvement is not as significant as for CCAT.

Confidence Along Adversarial Directions: In Fig. 3, we plot the probabilities for all ten classes along an adversarial direction. We note that these directions do not necessarily correspond to successful or high-confidence adversarial examples. Instead, we chose the first 10 test examples on SVHN and Cifar10. The adversarial examples were obtained using our L_∞

	MNIST:		SVHN:		CIFAR10:		CIFAR10:	
	all **unseen**		**all** **unseen**		**all** **unseen**		**unseen** **except $\mathbf{L_\infty}$** with ϵ=**0.06**	
	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓	FPR↓	RErr ↓
Normal	99.3	100.0	95.9	100.0	93.0	100.0	93.0	100.0
AT-50%	99.3	100.0	96.2	99.9	84.1	99.2	84.1	99.2
AT-100%	99.2	100.0	93.7	99.9	81.0	98.6	81.1	98.7
CCAT	23.4	23.9	57.5	61.1	86.3	94.8	69.1	77.6
* MSD	97.0	99.2	–	–	76.2	94.1	75.6	93.5
* TRADES	99.3	99.9	–	–	82.8	97.4	82.7	97.3
* AT-Madry	–	–	–	–	87.6	98.9	87.6	98.9

Table 6: Worst-Case Results Across Unseen Attacks. We report the (per-example) worst-case, confidence-thresholded RErr and FPR across **all** unseen attacks on MNIST, SVHN and Cifar10. On Cifar10, we additionally present results for all attacks except L_∞ adversarial examples with larger $\epsilon = 0.06$ (indicated in blue). CCAT is able to outperform all baselines, including MSD and TRADES, significantly on MNIST and SVHN. On Cifar10, CCAT performs poorly on L_∞ adversarial examples with larger $\epsilon = 0.06$. However, excluding these adversarial examples, CCAT also outperforms all baselines on Cifar10. * Pre-trained models with different architectures.

PGD-Conf attack with $T = 1000$ iterations and zero initialization for $\epsilon = 0.03$. For AT, we usually observe a change in predictions along these directions; some occur within $\|\delta\|_\infty \leq \epsilon$, corresponding to successful adversarial examples (within ϵ), some occur for $\|\delta\|_\infty > \epsilon$, corresponding to unsuccessful adversarial examples (within ϵ). However, AT always assigns high confidence. Thus, when allowing larger adversarial perturbations at test time, robustness of AT reduces significantly. For CCAT, in contrast, there are only few such cases; more often, the model achieves a near uniform prediction for small $\|\delta\|_\infty$ and extrapolates this behavior beyond the ϵ-ball used for training. On SVHN, this behavior successfully allows to generalize the robustness to larger adversarial perturbations. Furthermore, these plots illustrate why using more iterations at test time, and using techniques such as momentum and backtracking, are necessary to find adversarial examples as the objective becomes more complex compared to AT.

Confidence Along Interpolation: In Fig. 4, on MNIST, we additionally illustrate the advantage of CCAT with respect to the toy example in Proposition 1. Here, we consider the case where the ϵ-balls of two training or test examples (in different classes) overlap. As we show in Proposition 1, adversarial training is not able to handle such cases, resulting in the trade-off between accuracy in robustness reported in the literature (Tsipras et al., 2018; Stutz et al., 2019; Raghunathan et al., 2019; Zhang et al., 2019). This is because adversarial training enforces high-confidence predictions on both ϵ-balls (corresponding to different classes), resulting in an obvious conflict. CCAT, in contrast, enforces uniform predictions throughout the largest parts of both ϵ-balls, resolving the conflict.

C.7. Results

Main Results for 98% and 95% TPR: Tab. 4 reports our main results requiring only 98%TPR; Tab. 5 shows results for 95%TPR. This implies, that compared to 99%TPR, up to 1% (or 4%) more correctly classified test examples can be rejected, increasing the confidence threshold and potentially improving robustness. For relatively simple tasks such as MNIST and SVHN, where Err is low, this is a significant "sacrifice". However, as can be seen, robustness in terms of RErr only improves slightly. We found that the same holds for 95%TPR, however, rejecting more than 2% of correctly classified examples seems prohibitive large for the considered datasets.

Worst-Case Across Unseen Attacks: Tab. 6 reports *per-example* worst-case RErr and FPR for 99%TPR considering **all** unseen attacks. On MNIST and SVHN, RErr increases to nearly 100% for AT, both AT-50% and AT-100%. CCAT, in contrast, is able to achieve considerably lower RErr: 23.9% on MNIST and 61.1% on SVHN. Only on Cifar10, CCAT does not result in a significant improvement; all methods, including related work such as MSD and TRADES yield RErr of 94% or higher. However, this is mainly due to the poor performance of CCAT against large L_∞ adversarial examples with $\epsilon = 0.06$. Excluding these adversarial examples (right most table, indicated in blue) shows that RErr improves to 77.6% for

CCAT, while RErr for the remaining methods remains nearly unchanged. Overall, these experiments emphasize that CCAT is able to generalize robustness to previously unseen attacks.

Per-Attack Results: In Tab. 7 to 17, we break down our main results regarding all used L_p attacks for $p \in \{\infty, 2, 1, 0\}$. For simplicity we focus on PGD-CE and PGD-Conf while reporting the used black-box attacks together, i.e., taking the per-example worst-case adversarial examples across all black-box attacks. For comparison, we also include the area under the ROC curve (ROC AUC), non-thresholded Err and non-thresholded RErr. On MNIST, where AT performs very well in practice, it is striking that for $\nicefrac{4}{3}\epsilon = 0.4$ even black-box attacks are able to reduce robustness completely, resulting in high RErr. This observation also transfers to SVHN and Cifar10. For CCAT, black-box attacks are only effective on Cifar10, where they result in roughly 87% RErr with τ@99%TPR. For the L_2, L_1 and L_0 attacks we can make similar observations. Across all L_p norms, it can also be seen that PGD-CE performs significantly worse against our CCAT compared to AT, which shows that it is essential to optimize the right objective to evaluate the robustness of defenses and adversarially trained models, i.e., maximize confidence against CCAT.

Results on Corrupted MNIST/Cifar10: We also conducted experiments on MNIST-C (Mu & Gilmer, 2019) and Cifar10-C (Hendrycks & Dietterich, 2019). These datasets are variants of MNIST and Cifar10 that contain common perturbations of the original images obtained from various types of noise, blur or transformations; examples include zoom or motion blue, Gaussian and shot noise, rotations, translations and shear. Tab. 18 to 21 presents the per-corruption results on MNIST-C and Cifar10-C, respectively. Here, `all` includes all corruptions and `mean` reports the average results across all corruptions. We note that, due to the thresholding, different numbers of corrupted examples are left after detection for different corruptions. Thus, the distinction between `all` and `mean` is meaningful. Striking is the performance of CCAT on noise corruptions such as `gaussian_noise` or `shot_noise`. Here, CCAT is able to reject 100% of the corrupted examples, resulting in a thresholded Err of 0%. This is in stark contrast to AT, exhibiting a Err of roughly 15% after rejection on Cifar10-C. On the remaining corruptions, CCAT is able to perform slightly better than AT, which is often due to higher detection rate, i.e., higher ROC AUC. On, Cifar10, the generally lower Err of CCAT also contributes to the results. Overall, this illustrates that CCAT is able to preserve the inductive bias of predicting near-uniform distribution on noise similar to L_∞ adversarial examples as seen during training.

MNIST: Supplementary Results for $\mathbf{L}_\infty$ Adversarial Examples								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case ($L_\infty, \epsilon = 0.30$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.97	1.0	0.0	1.0	1.00	0.5	7.2
	AT-100%	0.98	1.0	0.0	1.0	1.00	0.5	7.1
	CCAT	0.99	6.9	0.2	7.1	0.85	0.4	48.9
	MSD	0.86	32.1	0.9	33.9	0.51	1.8	38.0
	TRADES	0.96	3.4	0.1	3.5	0.92	0.5	9.5
PGD Conf ($L_\infty, \epsilon = 0.30$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.97	0.4	0.0	0.4	1.00	0.5	5.6
	AT-100%	0.98	0.6	0.0	0.6	1.00	0.5	5.4
	CCAT	0.98	6.9	0.2	7.1	0.85	0.4	45.9
	MSD	0.87	28.7	0.9	30.5	0.51	1.8	34.6
	TRADES	0.96	1.6	0.1	1.6	0.92	0.5	7.0
PGD CE ($L_\infty, \epsilon = 0.30$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.97	0.8	0.0	0.8	1.00	0.5	6.7
	AT-100%	0.98	0.7	0.0	0.7	1.00	0.5	6.3
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	100.0
	MSD	0.88	28.0	0.9	29.8	0.51	1.8	37.6
	TRADES	0.96	2.8	0.1	2.9	0.92	0.5	8.6
Black-Box ($L_\infty, \epsilon = 0.30$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.98	1.0	0.0	1.0	1.00	0.5	7.2
	AT-100%	0.98	1.0	0.0	1.0	1.00	0.5	6.9
	CCAT	1.00	0.1	0.2	0.3	0.85	0.4	82.4
	MSD	0.88	27.7	0.9	29.5	0.51	1.8	35.9
	TRADES	0.96	3.2	0.1	3.3	0.92	0.5	9.1
Worst-Case ($L_\infty, \epsilon = 0.40$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.20	99.3	0.0	100.0	1.00	0.5	100.0
	AT-100%	0.27	99.2	0.0	100.0	1.00	0.5	100.0
	CCAT	0.97	11.4	0.2	11.6	0.85	0.4	69.3
	MSD	0.66	96.7	0.9	98.9	0.51	1.8	99.8
	TRADES	0.56	99.3	0.1	99.9	0.92	0.5	100.0
PGD Conf ($L_\infty, \epsilon = 0.40$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.36	97.1	0.0	97.8	1.00	0.5	99.8
	AT-100%	0.81	46.5	0.0	46.9	1.00	0.5	77.0
	CCAT	0.97	11.4	0.2	11.6	0.85	0.4	58.9
	MSD	0.73	89.3	0.9	91.4	0.51	1.8	93.3
	TRADES	0.76	87.0	0.1	87.5	0.92	0.5	96.9
PGD CE ($L_\infty, \epsilon = 0.40$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.20	99.3	0.0	100.0	1.00	0.5	100.0
	AT-100%	0.27	99.2	0.0	100.0	1.00	0.5	100.0
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	100.0
	MSD	0.69	95.6	0.9	97.8	0.51	1.8	99.7
	TRADES	0.58	99.3	0.1	99.9	0.92	0.5	100.0
Black-Box ($L_\infty, \epsilon = 0.40$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.23	99.3	0.0	100.0	1.00	0.5	100.0
	AT-100%	0.27	99.2	0.0	100.0	1.00	0.5	100.0
	CCAT	1.00	0.2	0.2	0.4	0.85	0.4	91.2
	MSD	0.76	92.5	0.9	94.7	0.51	1.8	99.3
	TRADES	0.78	95.4	0.1	96.0	0.92	0.5	99.8

Table 7: Per-Attack Results on MNIST, Part I ($\mathbf{L}_\infty$). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for the L_∞ threat model. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr, we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

MNIST: Supplementary Results for $\mathbf{L_2}$ Adversarial Examples								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case (L_2, $\epsilon = 1.5$)	Normal	0.39	98.9	0.1	99.6	0.98	0.4	100.0
	AT-50%	0.96	2.3	0.0	2.4	1.00	0.5	13.6
	AT-100%	0.96	2.6	0.0	2.7	1.00	0.5	12.0
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	6.7
	MSD	0.94	7.9	0.9	9.6	0.51	1.8	15.2
	TRADES	0.96	2.2	0.1	2.3	0.92	0.5	8.5
PGD Conf (L_2, $\epsilon = 1.5$)	Normal	0.56	83.3	0.1	83.9	0.98	0.4	91.6
	AT-50%	0.99	0.2	0.0	0.2	1.00	0.5	3.2
	AT-100%	0.99	0.1	0.0	0.1	1.00	0.5	2.5
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	5.5
	MSD	0.96	2.0	0.9	3.7	0.51	1.8	6.5
	TRADES	0.99	0.2	0.1	0.2	0.92	0.5	3.6
PGD CE (L_2, $\epsilon = 1.5$)	Normal	0.39	98.8	0.1	99.5	0.98	0.4	99.9
	AT-50%	0.96	1.9	0.0	2.0	1.00	0.5	10.2
	AT-100%	0.96	1.9	0.0	2.0	1.00	0.5	10.0
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	100.0
	MSD	0.94	7.5	0.9	9.2	0.51	1.8	15.8
	TRADES	0.96	2.2	0.1	2.3	0.92	0.5	8.2
Black-Box (L_2, $\epsilon = 1.5$)	Normal	0.55	91.0	0.1	91.7	0.98	0.4	97.4
	AT-50%	0.97	0.8	0.0	0.8	1.00	0.5	8.1
	AT-100%	0.98	0.9	0.0	0.9	1.00	0.5	6.8
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	84.8
	MSD	0.95	3.6	0.9	5.3	0.51	1.8	8.6
	TRADES	0.99	0.3	0.1	0.3	0.92	0.5	4.4
Worst-Case (L_2, $\epsilon = 3$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.73	80.7	0.0	81.4	1.00	0.5	98.8
	AT-100%	0.66	83.9	0.0	84.7	1.00	0.5	98.5
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	15.5
	MSD	0.81	57.0	0.9	59.0	0.51	1.8	67.6
	TRADES	0.90	43.6	0.1	44.0	0.92	0.5	69.9
PGD Conf (L_2, $\epsilon = 3$)	Normal	0.41	93.2	0.1	93.9	0.98	0.4	96.9
	AT-50%	0.98	0.2	0.0	0.2	1.00	0.5	3.4
	AT-100%	0.99	0.1	0.0	0.1	1.00	0.5	2.7
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	4.9
	MSD	0.96	2.3	0.9	4.0	0.51	1.8	7.3
	TRADES	0.98	0.4	0.1	0.4	0.92	0.5	3.8
PGD CE (L_2, $\epsilon = 3$)	Normal	0.34	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.93	11.3	0.0	11.5	1.00	0.5	29.2
	AT-100%	0.92	9.3	0.0	9.5	1.00	0.5	24.5
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	100.0
	MSD	0.82	55.7	0.9	57.8	0.51	1.8	67.7
	TRADES	0.95	5.3	0.1	5.4	0.92	0.5	15.9
Black-Box (L_2, $\epsilon = 3$)	Normal	0.35	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.73	79.5	0.0	80.1	1.00	0.5	98.7
	AT-100%	0.67	83.0	0.0	83.8	1.00	0.5	98.1
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	88.4
	MSD	0.88	28.6	0.9	30.4	0.51	1.8	37.4
	TRADES	0.91	42.6	0.1	43.0	0.92	0.5	69.1

Table 8: Per-Attack Results on MNIST, Part II ($\mathbf{L_2}$). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for the L_2 threat model. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

MNIST: Supplementary Results for $\mathbf{L_1}$ Adversarial Examples								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case ($L_1, \epsilon = 12$)	Normal	0.44	98.9	0.1	99.6	0.98	0.4	100.0
	AT-50%	0.93	8.6	0.0	8.8	1.00	0.5	26.3
	AT-100%	0.91	8.2	0.0	8.4	1.00	0.5	23.6
	CCAT	1.00	0.8	0.2	1.0	0.85	0.4	16.5
	MSD	0.87	25.5	0.9	27.3	0.51	1.8	32.3
	TRADES	0.96	2.7	0.1	2.8	0.92	0.5	9.5
PGD Conf ($L_1, \epsilon = 12$)	Normal	0.45	98.9	0.1	99.6	0.98	0.4	100.0
	AT-50%	0.92	6.3	0.0	6.4	1.00	0.5	19.3
	AT-100%	0.90	5.8	0.0	6.0	1.00	0.5	16.7
	CCAT	1.00	0.7	0.2	0.9	0.85	0.4	14.2
	MSD	0.87	24.8	0.9	26.6	0.51	1.8	31.7
	TRADES	0.96	1.8	0.1	1.9	0.92	0.5	6.7
PGD CE ($L_1, \epsilon = 12$)	Normal	0.58	92.4	0.1	93.1	0.98	0.4	96.5
	AT-50%	0.94	6.4	0.0	6.6	1.00	0.5	20.2
	AT-100%	0.92	5.9	0.0	6.1	1.00	0.5	17.3
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	100.0
	MSD	0.92	12.5	0.9	14.3	0.51	1.8	21.8
	TRADES	0.96	2.4	0.1	2.5	0.92	0.5	8.8
Black-Box ($L_1, \epsilon = 12$)	Normal	0.98	3.3	0.1	3.5	0.98	0.4	10.8
	AT-50%	0.96	0.7	0.0	0.7	1.00	0.5	4.4
	AT-100%	0.97	0.8	0.0	0.8	1.00	0.5	5.3
	CCAT	1.00	0.1	0.2	0.3	0.85	0.4	8.8
	MSD	0.96	0.4	0.9	2.1	0.51	1.8	3.2
	TRADES	0.98	0.1	0.1	0.1	0.92	0.5	1.4
Worst-Case ($L_1, \epsilon = 18$)	Normal	0.36	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.88	23.8	0.0	24.1	1.00	0.5	50.4
	AT-100%	0.88	20.2	0.0	20.7	1.00	0.5	43.8
	CCAT	1.00	1.3	0.2	1.5	0.85	0.4	22.0
	MSD	0.81	53.7	0.9	55.6	0.51	1.8	60.8
	TRADES	0.95	8.2	0.1	8.4	0.92	0.5	21.4
PGD Conf ($L_1, \epsilon = 18$)	Normal	0.36	99.3	0.1	100.0	0.98	0.4	100.0
	AT-50%	0.88	15.5	0.0	15.7	1.00	0.5	33.8
	AT-100%	0.87	12.7	0.0	13.0	1.00	0.5	27.6
	CCAT	1.00	0.8	0.2	1.0	0.85	0.4	15.5
	MSD	0.81	53.1	0.9	55.0	0.51	1.8	60.3
	TRADES	0.95	2.9	0.1	3.0	0.92	0.5	8.9
PGD CE ($L_1, \epsilon = 18$)	Normal	0.45	97.7	0.1	98.4	0.98	0.4	98.8
	AT-50%	0.90	17.0	0.0	17.3	1.00	0.5	37.6
	AT-100%	0.90	11.7	0.0	12.0	1.00	0.5	29.8
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	100.0
	MSD	0.88	27.0	0.9	28.9	0.51	1.8	36.3
	TRADES	0.95	4.8	0.1	4.9	0.92	0.5	13.6
Black-Box ($L_1, \epsilon = 18$)	Normal	0.98	3.3	0.1	3.5	0.98	0.4	10.8
	AT-50%	0.96	0.7	0.0	0.7	1.00	0.5	4.4
	AT-100%	0.97	0.8	0.0	0.8	1.00	0.5	5.3
	CCAT	1.00	0.1	0.2	0.3	0.85	0.4	8.8
	MSD	0.96	0.4	0.9	2.1	0.51	1.8	3.2
	TRADES	0.98	0.1	0.1	0.1	0.92	0.5	1.4

Table 9: Per-Attack Results on MNIST, Part III ($\mathbf{L_1}$). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for the L_1 threat model. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

MNIST: Supplementary Results for $\mathbf{L_0}$ Adversarial Examples and **Adversarial Frames**								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case (L_0, $\epsilon = 15$)	Normal	0.63	91.6	0.1	92.3	0.98	0.4	99.1
	AT-50%	0.94	23.0	0.0	23.5	1.00	0.5	95.3
	AT-100%	0.98	12.9	0.0	13.3	1.00	0.5	94.3
	CCAT	0.99	14.2	0.2	14.5	0.85	0.4	83.5
	MSD	0.75	64.2	0.9	66.2	0.51	1.8	74.2
	TRADES	0.87	34.6	0.1	35.1	0.92	0.5	86.7
PGD Conf (L_0, $\epsilon = 15$)	Normal	0.69	84.4	0.1	85.0	0.98	0.4	92.7
	AT-50%	0.98	0.5	0.0	0.5	1.00	0.5	5.2
	AT-100%	0.98	0.3	0.0	0.3	1.00	0.5	3.6
	CCAT	0.99	3.1	0.2	3.3	0.85	0.4	13.7
	MSD	0.88	26.7	0.9	28.5	0.51	1.8	35.4
	TRADES	0.97	3.0	0.1	3.1	0.92	0.5	12.6
PGD CE (L_0, $\epsilon = 15$)	Normal	0.63	90.3	0.1	90.9	0.98	0.4	95.2
	AT-50%	0.98	2.3	0.0	2.4	1.00	0.5	17.7
	AT-100%	0.99	0.5	0.0	0.5	1.00	0.5	11.1
	CCAT	1.00	0.9	0.2	1.1	0.85	0.4	10.3
	MSD	0.88	30.5	0.9	32.3	0.51	1.8	41.5
	TRADES	0.95	10.7	0.1	11.0	0.92	0.5	27.9
Black-Box (L_0, $\epsilon = 15$)	Normal	0.97	18.3	0.1	18.5	0.98	0.4	98.6
	AT-50%	0.94	21.7	0.0	22.3	1.00	0.5	95.3
	AT-100%	0.98	12.9	0.0	13.3	1.00	0.5	94.3
	CCAT	0.99	12.5	0.2	12.8	0.85	0.4	84.4
	MSD	0.78	51.8	0.9	54.0	0.51	1.8	70.8
	TRADES	0.88	31.7	0.1	32.4	0.92	0.5	86.7
Adversarial Frames	Normal	0.69	87.0	0.1	87.6	0.98	0.4	94.6
	AT-50%	0.82	72.9	0.0	73.6	1.00	0.5	95.4
	AT-100%	0.86	61.3	0.0	62.0	1.00	0.5	93.9
	CCAT	1.00	0.0	0.2	0.2	0.85	0.4	95.4
	MSD	0.96	6.6	0.9	8.3	0.51	1.8	17.3
	TRADES	0.99	0.0	0.1	0.0	0.92	0.5	1.9

Table 10: Per-Attack Results on MNIST, Part IV ($\mathbf{L_0}$, Adversarial Frames). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for L_0 threat models and adversarial frames. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

SVHN: Supplementary Results for $\mathbf{L}_\infty$ Adversarial Examples								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case ($L_\infty, \epsilon = 0.03$)	Normal	0.17	95.8	2.6	99.9	0.78	3.6	100.0
	AT-50%	0.55	52.3	2.5	55.6	0.56	3.4	57.3
	AT-100%	0.73	42.1	4.6	47.5	0.27	5.9	48.4
	CCAT	0.70	35.5	2.1	38.5	0.60	2.9	97.8
PGD Conf ($L_\infty, \epsilon = 0.03$)	Normal	0.17	95.8	2.6	99.9	0.78	3.6	99.9
	AT-50%	0.55	51.6	2.5	54.9	0.56	3.4	56.9
	AT-100%	0.73	40.9	4.6	46.2	0.27	5.9	47.1
	CCAT	0.67	35.5	2.1	38.5	0.60	2.9	91.0
PGD CE ($L_\infty, \epsilon = 0.03$)	Normal	0.17	95.8	2.6	99.9	0.78	3.6	100.0
	AT-50%	0.68	40.0	2.5	43.2	0.56	3.4	50.7
	AT-100%	0.81	34.6	4.6	39.9	0.27	5.9	45.3
	CCAT	1.00	0.0	2.1	2.6	0.60	2.9	94.9
Black-Box ($L_\infty, \epsilon = 0.03$)	Normal	0.23	95.6	2.6	99.7	0.78	3.6	99.7
	AT-50%	0.95	27.7	2.5	30.8	0.56	3.4	46.2
	AT-100%	0.78	38.5	4.6	43.8	0.27	5.9	45.5
	CCAT	1.00	3.7	2.1	6.4	0.60	2.9	79.6
Worst-Case ($L_\infty, \epsilon = 0.06$)	Normal	0.16	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.32	84.7	2.5	88.3	0.56	3.4	89.0
	AT-100%	0.38	80.9	4.6	86.9	0.27	5.9	87.1
	CCAT	0.64	49.5	2.1	52.6	0.60	2.9	99.9
PGD Conf ($L_\infty, \epsilon = 0.06$)	Normal	0.16	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.31	81.1	2.5	84.7	0.56	3.4	86.1
	AT-100%	0.40	77.4	4.6	83.3	0.27	5.9	83.6
	CCAT	0.64	42.1	2.1	45.2	0.60	2.9	98.1
PGD CE ($L_\infty, \epsilon = 0.06$)	Normal	0.16	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.61	84.4	2.5	88.0	0.56	3.4	88.9
	AT-100%	0.43	80.9	4.6	86.9	0.27	5.9	87.1
	CCAT	0.99	14.3	2.1	17.2	0.60	2.9	100.0
Black-Box ($L_\infty, \epsilon = 0.06$)	Normal	0.17	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.78	78.9	2.5	82.4	0.56	3.4	84.0
	AT-100%	0.51	76.5	4.6	82.4	0.27	5.9	82.7
	CCAT	1.00	4.5	2.1	7.2	0.60	2.9	83.7

Table 11: Per-Attack Results on SVHN, Part I ($\mathbf{L}_\infty$). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for the L_∞ threat model. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

SVHN: Supplementary Results for $\mathbf{L_2}$ Adversarial Examples

		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case (L_2, $\epsilon = 0.5$)	Normal	0.27	93.3	2.6	97.4	0.78	3.6	97.7
	AT-50%	0.53	65.2	2.5	68.6	0.56	3.4	70.7
	AT-100%	0.67	60.1	4.6	65.8	0.27	5.9	66.3
	CCAT	0.87	17.2	2.1	20.0	0.60	2.9	63.3
PGD Conf (L_2, $\epsilon = 0.5$)	Normal	0.32	91.1	2.6	95.1	0.78	3.6	96.4
	AT-50%	0.56	57.5	2.5	60.8	0.56	3.4	62.9
	AT-100%	0.68	54.0	4.6	59.6	0.27	5.9	60.2
	CCAT	0.85	16.9	2.1	19.7	0.60	2.9	54.5
PGD CE (L_2, $\epsilon = 0.5$)	Normal	0.28	93.3	2.6	97.4	0.78	3.6	97.7
	AT-50%	0.57	63.1	2.5	66.5	0.56	3.4	70.1
	AT-100%	0.72	57.7	4.6	63.5	0.27	5.9	66.5
	CCAT	0.99	1.8	2.1	4.5	0.60	2.9	100.0
Black-Box (L_2, $\epsilon = 0.5$)	Normal	0.67	53.8	2.6	57.3	0.78	3.6	63.7
	AT-50%	0.99	3.0	2.5	5.9	0.56	3.4	13.8
	AT-100%	0.90	7.3	4.6	12.2	0.27	5.9	15.3
	CCAT	1.00	0.1	2.1	2.7	0.60	2.9	91.3
Worst-Case(L_2, $\epsilon = 1$)	Normal	0.17	95.8	2.6	99.9	0.78	3.6	99.9
	AT-50%	0.26	88.4	2.5	92.0	0.56	3.4	92.4
	AT-100%	0.38	87.3	4.6	93.4	0.27	5.9	93.5
	CCAT	0.88	18.9	2.1	21.7	0.60	2.9	80.9
PGD Conf (L_2, $\epsilon = 1$)	Normal	0.22	95.4	2.6	99.5	0.78	3.6	99.6
	AT-50%	0.49	73.6	2.5	77.1	0.56	3.4	78.7
	AT-100%	0.48	75.5	4.6	81.4	0.27	5.9	81.7
	CCAT	0.88	18.6	2.1	21.4	0.60	2.9	74.6
PGD CE (L_2, $\epsilon = 1$)	Normal	0.17	95.8	2.6	99.9	0.78	3.6	99.9
	AT-50%	0.27	88.4	2.5	92.0	0.56	3.4	92.4
	AT-100%	0.39	87.3	4.6	93.4	0.27	5.9	93.5
	CCAT	0.99	2.1	2.1	4.8	0.60	2.9	100.0
Black-Box (L_2, $\epsilon = 1$)	Normal	0.42	88.3	2.6	92.3	0.78	3.6	94.4
	AT-50%	0.98	10.8	2.5	13.8	0.56	3.4	29.8
	AT-100%	0.85	22.5	4.6	27.7	0.27	5.9	30.1
	CCAT	1.00	0.1	2.1	2.7	0.60	2.9	96.5
Worst-Case (L_2, $\epsilon = 2$)	Normal	0.16	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.11	95.7	2.5	99.4	0.56	3.4	99.4
	AT-100%	0.12	93.3	4.6	99.5	0.27	5.9	99.5
	CCAT	0.89	25.4	2.1	28.3	0.60	2.9	91.9
PGD Conf (L_2, $\epsilon = 2$)	Normal	0.21	95.4	2.6	99.5	0.78	3.6	99.8
	AT-50%	0.49	76.2	2.5	79.7	0.56	3.4	81.2
	AT-100%	0.45	78.1	4.6	84.0	0.27	5.9	84.3
	CCAT	0.89	19.1	2.1	21.9	0.60	2.9	88.0
PGD CE (L_2, $\epsilon = 2$)	Normal	0.16	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.12	95.7	2.5	99.4	0.56	3.4	99.4
	AT-100%	0.12	93.3	4.6	99.5	0.27	5.9	99.5
	CCAT	0.98	5.0	2.1	7.7	0.60	2.9	100.0

Table 12: Per-Attack Results on SVHN, Part II ($\mathbf{L_2}$). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for L_2 threat model. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

SVHN: Supplementary Results for $\mathbf{L_1}$, $\mathbf{L_0}$ Adversarial Examples and **Adversarial Frames**								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case (L_1, $\epsilon = 18$)	Normal	0.17	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.09	95.0	2.5	98.7	0.56	3.4	98.7
	AT-100%	0.14	93.1	4.6	99.3	0.27	5.9	99.3
	CCAT	0.82	24.3	2.1	27.2	0.60	2.9	72.8
PGD Conf (L_1, $\epsilon = 18$)	Normal	0.17	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.11	94.8	2.5	98.5	0.56	3.4	98.5
	AT-100%	0.15	92.9	4.6	99.1	0.27	5.9	99.1
	CCAT	0.82	23.4	2.1	26.3	0.60	2.9	69.3
PGD CE (L_1, $\epsilon = 18$)	Normal	0.21	95.7	2.6	99.8	0.78	3.6	99.8
	AT-50%	0.29	92.8	2.5	96.5	0.56	3.4	96.6
	AT-100%	0.27	92.1	4.6	98.3	0.27	5.9	98.3
	CCAT	0.99	2.7	2.1	5.4	0.60	2.9	100.0
Black-Box (L_1, $\epsilon = 18$)	Normal	0.89	6.2	2.6	9.2	0.78	3.6	13.0
	AT-50%	0.97	3.7	2.5	6.6	0.56	3.4	10.9
	AT-100%	0.90	7.4	4.6	12.3	0.27	5.9	14.9
	CCAT	1.00	0.5	2.1	3.2	0.60	2.9	10.0
Worst-Case (L_1, $\epsilon = 24$)	Normal	0.16	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.05	95.8	2.5	99.5	0.56	3.4	99.5
	AT-100%	0.09	93.6	4.6	99.8	0.27	5.9	99.8
	CCAT	0.80	28.1	2.1	31.0	0.60	2.9	77.0
PGD Conf (L_1, $\epsilon = 24$)	Normal	0.16	95.9	2.6	100.0	0.78	3.6	100.0
	AT-50%	0.06	95.7	2.5	99.4	0.56	3.4	99.4
	AT-100%	0.09	93.6	4.6	99.8	0.27	5.9	99.8
	CCAT	0.79	27.1	2.1	30.0	0.60	2.9	73.3
PGD CE (L_1, $\epsilon = 24$)	Normal	0.19	95.8	2.6	99.9	0.78	3.6	99.9
	AT-50%	0.25	94.0	2.5	97.7	0.56	3.4	98.0
	AT-100%	0.22	92.5	4.6	98.7	0.27	5.9	98.7
	CCAT	0.98	3.4	2.1	6.1	0.60	2.9	100.0
Black-Box (L_1, $\epsilon = 24$)	Normal	0.86	18.7	2.6	21.8	0.78	3.6	27.1
	AT-50%	0.96	11.2	2.5	14.2	0.56	3.4	22.2
	AT-100%	0.86	16.3	4.6	21.3	0.27	5.9	24.6
	CCAT	1.00	1.3	2.1	4.0	0.60	2.9	18.3
Worst-Case (L_0, $\epsilon = 10$)	Normal	0.60	79.6	2.6	83.5	0.78	3.6	98.0
	AT-50%	0.90	70.0	2.5	73.5	0.56	3.4	89.2
	AT-100%	0.65	83.1	4.6	89.2	0.27	5.9	89.9
	CCAT	1.00	0.4	2.1	3.0	0.60	2.9	75.3
PGD Conf (L_0, $\epsilon = 10$)	Normal	0.55	63.4	2.6	67.1	0.78	3.6	69.0
	AT-50%	0.90	48.0	2.5	51.3	0.56	3.4	59.7
	AT-100%	0.64	61.3	4.6	67.2	0.27	5.9	68.0
	CCAT	1.00	0.1	2.1	2.7	0.60	2.9	63.9
PGD CE (L_0, $\epsilon = 10$)	Normal	0.52	71.4	2.6	75.2	0.78	3.6	78.1
	AT-50%	0.89	57.4	2.5	60.8	0.56	3.4	68.9
	AT-100%	0.65	68.2	4.6	74.2	0.27	5.9	75.5
	CCAT	1.00	0.0	2.1	2.6	0.60	2.9	96.9
Black-Box (L_0, $\epsilon = 10$)	Normal	0.96	44.5	2.6	47.9	0.78	3.6	97.7
	AT-50%	0.98	34.4	2.5	37.7	0.56	3.4	86.6
	AT-100%	0.86	74.3	4.6	80.6	0.27	5.9	85.5
	CCAT	1.00	0.3	2.1	3.0	0.60	2.9	93.1
Adversarial Frames	Normal	0.47	74.6	2.6	78.4	0.78	3.6	80.1
	AT-50%	0.77	30.0	2.5	33.1	0.56	3.4	36.0
	AT-100%	0.78	19.9	4.6	25.1	0.27	5.9	27.4
	CCAT	1.00	1.0	2.1	3.7	0.60	2.9	97.3

Table 13: Per-Attack Results on SVHN, Part III ($\mathbf{L_1}$, $\mathbf{L_0}$, Adversarial Frames). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for L_1, L_0 threat models and adversarial frames. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

CIFAR10: Supplementary Results for $\mathbf{L}_\infty$ Adversarial Examples								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case ($L_\infty, \epsilon = 0.03$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.64	47.6	15.1	62.3	0.35	16.6	62.7
	AT-100%	0.64	42.3	18.3	59.5	0.29	19.4	59.9
	CCAT	0.60	59.9	8.7	67.9	0.40	10.1	96.7
	MSD	0.66	35.3	17.6	53.0	0.24	18.4	53.2
	TRADES	0.73	28.9	13.2	41.9	0.30	15.2	43.5
	AT-Madry	0.73	33.8	11.7	44.4	0.29	13.0	45.1
PGD Conf ($L_\infty, \epsilon = 0.03$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.65	46.2	15.1	60.9	0.35	16.6	61.3
	AT-100%	0.65	40.0	18.3	57.2	0.29	19.4	57.6
	CCAT	0.60	55.1	8.7	63.0	0.40	10.1	95.0
	MSD	0.66	33.0	17.6	50.7	0.24	18.4	50.9
	TRADES	0.73	28.2	13.2	41.2	0.30	15.2	42.8
	AT-Madry	0.73	31.9	11.7	42.5	0.29	13.0	43.2
PGD CE ($L_\infty, \epsilon = 0.03$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.67	46.7	15.1	61.4	0.35	16.6	62.3
	AT-100%	0.69	40.5	18.3	57.7	0.29	19.4	59.3
	CCAT	0.99	2.5	8.7	9.8	0.40	10.1	100.0
	MSD	0.75	33.2	17.6	50.9	0.24	18.4	52.6
	TRADES	0.80	25.3	13.2	38.4	0.30	15.2	42.9
	AT-Madry	0.78	30.3	11.7	40.9	0.29	13.0	44.2
Black-Box ($L_\infty, \epsilon = 0.03$)	Normal	0.21	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.70	42.2	15.1	56.9	0.35	16.6	57.3
	AT-100%	0.70	36.9	18.3	54.1	0.29	19.4	54.7
	CCAT	0.90	41.6	8.7	49.3	0.40	10.1	96.4
	MSD	0.73	30.2	17.6	47.9	0.24	18.4	48.4
	TRADES	0.78	25.1	13.2	38.0	0.30	15.2	40.0
	AT-Madry	0.78	29.2	11.7	39.7	0.29	13.0	41.2
Worst-Case ($L_\infty, \epsilon = 0.06$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.35	78.6	15.1	93.6	0.35	16.6	93.7
	AT-100%	0.39	72.7	18.3	90.2	0.29	19.4	90.3
	CCAT	0.40	83.9	8.7	92.3	0.40	10.1	99.4
	MSD	0.43	71.5	17.6	89.4	0.24	18.4	89.4
	TRADES	0.53	66.4	13.2	80.5	0.30	15.2	81.0
	AT-Madry	0.46	73.2	11.7	84.3	0.29	13.0	84.5
PGD Conf ($L_\infty, \epsilon = 0.06$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.37	77.1	15.1	92.1	0.35	16.6	92.2
	AT-100%	0.43	70.7	18.3	88.2	0.29	19.4	88.3
	CCAT	0.45	64.5	8.7	72.6	0.40	10.1	97.4
	MSD	0.46	68.7	17.6	86.5	0.24	18.4	86.6
	TRADES	0.54	63.9	13.2	77.9	0.30	15.2	78.5
	AT-Madry	0.51	69.4	11.7	80.4	0.29	13.0	80.7
PGD CE ($L_\infty, \epsilon = 0.06$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.40	78.6	15.1	93.6	0.35	16.6	93.7
	AT-100%	0.45	72.7	18.3	90.2	0.29	19.4	90.3
	CCAT	0.98	3.7	8.7	11.0	0.40	10.1	100.0
	MSD	0.53	71.6	17.6	89.5	0.24	18.4	89.5
	TRADES	0.64	64.8	13.2	78.8	0.30	15.2	80.4
	AT-Madry	0.50	73.1	11.7	84.2	0.29	13.0	84.4
Black-Box ($L_\infty, \epsilon = 0.06$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.50	72.1	15.1	87.1	0.35	16.6	87.2
	AT-100%	0.53	65.9	18.3	83.4	0.29	19.4	83.5
	CCAT	0.77	79.1	8.7	87.4	0.40	10.1	99.7
	MSD	0.58	60.5	17.6	78.3	0.24	18.4	78.5
	TRADES	0.65	57.6	13.2	71.4	0.30	15.2	72.2
	AT-Madry	0.62	63.5	11.7	74.5	0.29	13.0	75.2

Table 14: Per-Attack Results on Cifar10, Part I ($\mathbf{L}_\infty$). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for L_∞ and L_2 threat models. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

CIFAR10: Supplementary Results for $\mathbf{L_2}$ Adversarial Examples								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case (L_2, $\epsilon = 1$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.59	59.1	15.1	73.9	0.35	16.6	74.4
	AT-100%	0.57	55.8	18.3	73.2	0.29	19.4	73.4
	CCAT	0.75	42.6	8.7	50.4	0.40	10.1	83.2
	MSD	0.68	34.3	17.6	52.0	0.24	18.4	52.2
	TRADES	0.64	56.3	13.2	70.1	0.30	15.2	71.3
	AT-Madry	0.59	62.4	11.7	73.4	0.29	13.0	73.8
PGD Conf (L_2, $\epsilon = 1$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.63	50.2	15.1	64.9	0.35	16.6	65.3
	AT-100%	0.62	46.4	18.3	63.7	0.29	19.4	64.0
	CCAT	0.76	41.9	8.7	49.6	0.40	10.1	82.6
	MSD	0.68	30.4	17.6	48.1	0.24	18.4	48.3
	TRADES	0.67	43.1	13.2	56.5	0.30	15.2	57.7
	AT-Madry	0.68	44.8	11.7	55.5	0.29	13.0	56.1
PGD CE (L_2, $\epsilon = 1$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.61	59.1	15.1	73.9	0.35	16.6	74.6
	AT-100%	0.60	55.1	18.3	72.5	0.29	19.4	73.7
	CCAT	0.93	13.8	8.7	21.2	0.40	10.1	100.0
	MSD	0.74	33.6	17.6	51.3	0.24	18.4	52.1
	TRADES	0.68	56.1	13.2	70.0	0.30	15.2	72.6
	AT-Madry	0.61	61.9	11.7	72.9	0.29	13.0	74.1
Black-Box (L_2, $\epsilon = 1$)	Normal	0.38	90.1	7.4	97.1	0.59	8.3	97.4
	AT-50%	0.81	21.3	15.1	35.8	0.35	16.6	36.9
	AT-100%	0.78	19.8	18.3	36.8	0.29	19.4	37.8
	CCAT	0.99	5.0	8.7	12.3	0.40	10.1	78.8
	MSD	0.80	16.5	17.6	34.2	0.24	18.4	34.6
	TRADES	0.84	15.2	13.2	27.9	0.30	15.2	31.0
	AT-Madry	0.83	16.3	11.7	26.7	0.29	13.0	28.8
Worst-Case (L_2, $\epsilon = 2$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.28	83.3	15.1	98.4	0.35	16.6	98.4
	AT-100%	0.34	80.7	18.3	98.3	0.29	19.4	98.4
	CCAT	0.74	43.7	8.7	51.5	0.40	10.1	85.8
	MSD	0.53	70.6	17.6	88.5	0.24	18.4	88.8
	TRADES	0.64	56.3	13.2	70.1	0.30	15.2	71.3
	AT-Madry	0.26	87.4	11.7	98.7	0.29	13.0	98.7
PGD Conf (L_2, $\epsilon = 2$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.62	52.4	15.1	67.2	0.35	16.6	67.5
	AT-100%	0.60	48.2	18.3	65.5	0.29	19.4	65.8
	CCAT	0.76	42.4	8.7	50.2	0.40	10.1	84.6
	MSD	0.67	31.9	17.6	49.6	0.24	18.4	49.8
	TRADES	0.67	43.1	13.2	56.5	0.30	15.2	57.7
	AT-Madry	0.67	48.6	11.7	59.4	0.29	13.0	59.9
PGD CE (L_2, $\epsilon = 2$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.28	83.3	15.1	98.4	0.35	16.6	98.4
	AT-100%	0.34	80.7	18.3	98.3	0.29	19.4	98.4
	CCAT	0.92	15.9	8.7	23.3	0.40	10.1	100.0
	MSD	0.54	71.2	17.6	89.1	0.24	18.4	89.5
	TRADES	0.68	56.1	13.2	70.0	0.30	15.2	72.6
	AT-Madry	0.26	87.4	11.7	98.7	0.29	13.0	98.7
Black-Box (L_2, $\epsilon = 2$)	Normal	0.21	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.67	47.5	15.1	62.2	0.35	16.6	62.6
	AT-100%	0.66	43.4	18.3	60.6	0.29	19.4	61.3
	CCAT	0.99	7.7	8.7	15.1	0.40	10.1	81.0
	MSD	0.71	33.7	17.6	51.5	0.24	18.4	51.8
	TRADES	0.84	15.2	13.2	27.9	0.30	15.2	31.0
	AT-Madry	0.73	42.2	11.7	52.9	0.29	13.0	53.9

Table 15: Per-Attack Results on Cifar10, Part II ($\mathbf{L_2}$). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for the L_2 threat model. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

CIFAR10: Supplementary Results for $\mathbf{L_1}$ Adversarial Examples

		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case (L_1, $\epsilon = 18$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.39	80.1	15.1	95.2	0.35	16.6	95.2
	AT-100%	0.39	76.8	18.3	94.3	0.29	19.4	94.4
	CCAT	0.67	47.6	8.7	55.4	0.40	10.1	85.4
	MSD	0.63	40.0	17.6	57.7	0.24	18.4	57.9
	TRADES	0.45	78.6	13.2	93.0	0.30	15.2	93.2
	AT-Madry	0.39	83.0	11.7	94.2	0.29	13.0	94.3
PGD Conf (L_1, $\epsilon = 18$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.39	79.9	15.1	94.9	0.35	16.6	95.0
	AT-100%	0.40	76.5	18.3	94.0	0.29	19.4	94.1
	CCAT	0.67	47.3	8.7	55.1	0.40	10.1	85.4
	MSD	0.63	38.6	17.6	56.3	0.24	18.4	56.5
	TRADES	0.45	78.4	13.2	92.8	0.30	15.2	93.0
	AT-Madry	0.41	82.5	11.7	93.7	0.29	13.0	93.8
PGD CE (L_1, $\epsilon = 18$)	Normal	0.20	92.9	7.4	99.9	0.59	8.3	99.9
	AT-50%	0.57	69.0	15.1	83.9	0.35	16.6	85.0
	AT-100%	0.55	66.2	18.3	83.7	0.29	19.4	84.6
	CCAT	0.94	13.8	8.7	21.1	0.40	10.1	100.0
	MSD	0.75	32.2	17.6	49.9	0.24	18.4	51.1
	TRADES	0.63	69.4	13.2	83.6	0.30	15.2	85.5
	AT-Madry	0.55	74.6	11.7	85.8	0.29	13.0	87.0
Black-Box (L_1, $\epsilon = 18$)	Normal	0.94	3.6	7.4	9.9	0.59	8.3	11.6
	AT-50%	0.93	1.9	15.1	16.3	0.35	16.6	17.6
	AT-100%	0.93	1.0	18.3	17.9	0.29	19.4	19.1
	CCAT	0.99	0.8	8.7	8.1	0.40	10.1	18.7
	MSD	0.84	0.8	17.6	18.4	0.24	18.4	18.8
	TRADES	0.92	0.9	13.2	13.3	0.30	15.2	15.9
	AT-Madry	0.90	1.1	11.7	11.3	0.29	13.0	12.8
Worst-Case (L_1, $\epsilon = 24$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.28	83.3	15.1	98.4	0.35	16.6	98.4
	AT-100%	0.31	80.4	18.3	98.0	0.29	19.4	98.0
	CCAT	0.62	50.3	8.7	58.2	0.40	10.1	87.0
	MSD	0.57	50.7	17.6	68.5	0.24	18.4	68.6
	TRADES	0.37	82.3	13.2	96.8	0.30	15.2	97.0
	AT-Madry	0.29	86.5	11.7	97.8	0.29	13.0	97.8
PGD Conf (L_1, $\epsilon = 24$)	Normal	0.20	93.0	7.4	100.0	0.59	8.3	100.0
	AT-50%	0.29	83.3	15.1	98.4	0.35	16.6	98.4
	AT-100%	0.32	80.1	18.3	97.7	0.29	19.4	97.7
	CCAT	0.63	50.2	8.7	58.1	0.40	10.1	86.9
	MSD	0.57	49.3	17.6	67.1	0.24	18.4	67.2
	TRADES	0.37	82.3	13.2	96.8	0.30	15.2	96.9
	AT-Madry	0.31	86.3	11.7	97.6	0.29	13.0	97.6
PGD CE (L_1, $\epsilon = 24$)	Normal	0.20	92.9	7.4	99.9	0.59	8.3	99.9
	AT-50%	0.52	76.3	15.1	91.3	0.35	16.6	92.5
	AT-100%	0.51	72.7	18.3	90.2	0.29	19.4	91.2
	CCAT	0.94	15.0	8.7	22.4	0.40	10.1	100.0
	MSD	0.72	38.7	17.6	56.4	0.24	18.4	57.9
	TRADES	0.59	75.6	13.2	89.9	0.30	15.2	91.2
	AT-Madry	0.49	80.5	11.7	91.7	0.29	13.0	92.4
Black-Box (L_1, $\epsilon = 24$)	Normal	0.93	8.6	7.4	14.9	0.59	8.3	17.4
	AT-50%	0.93	3.5	15.1	17.9	0.35	16.6	19.7
	AT-100%	0.90	3.8	18.3	20.7	0.29	19.4	21.8
	CCAT	0.99	1.6	8.7	8.9	0.40	10.1	26.8
	MSD	0.87	2.6	17.6	20.2	0.24	18.4	20.6
	TRADES	0.89	1.7	13.2	14.0	0.30	15.2	16.7
	AT-Madry	0.89	2.9	11.7	13.2	0.29	13.0	14.9

Table 16: Per-Attack Results on Cifar10, Part III ($\mathbf{L_1}$). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for the L_1 threat model. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

CIFAR10: Supplementary Results for $\mathbf{L_0}$ Adversarial Examples and **Adversarial Frames**								
		Detection Setting τ@99%TPR					**Standard Setting** τ=0	
Attack	Training	ROC AUC	FPR in %	Err in %	RErr in %	τ	Err in %	RErr in %
Worst-Case (L_0, $\epsilon = 10$)	Normal	0.80	77.7	7.4	84.6	0.59	8.3	96.2
	AT-50%	0.80	59.3	15.1	74.1	0.35	16.6	75.9
	AT-100%	0.75	54.7	18.3	72.0	0.29	19.4	73.2
	CCAT	0.97	14.4	8.7	21.9	0.40	10.1	54.7
	MSD	0.84	21.4	17.6	39.1	0.24	18.4	39.8
	TRADES	0.78	22.3	13.2	35.2	0.30	15.2	37.8
	AT-Madry	0.73	31.0	11.7	41.5	0.29	13.0	42.8
PGD Conf (L_0, $\epsilon = 10$)	Normal	0.74	48.9	7.4	55.5	0.59	8.3	57.4
	AT-50%	0.74	29.7	15.1	44.3	0.35	16.6	45.1
	AT-100%	0.72	25.2	18.3	42.3	0.29	19.4	43.3
	CCAT	0.98	5.2	8.7	12.5	0.40	10.1	34.8
	MSD	0.81	5.2	17.6	22.8	0.24	18.4	23.1
	TRADES	0.80	18.0	13.2	30.8	0.30	15.2	33.2
	AT-Madry	0.77	21.8	11.7	32.3	0.29	13.0	33.5
PGD CE (L_0, $\epsilon = 10$)	Normal	0.73	56.3	7.4	63.0	0.59	8.3	65.4
	AT-50%	0.76	34.8	15.1	49.4	0.35	16.6	51.4
	AT-100%	0.72	30.3	18.3	47.5	0.29	19.4	48.9
	CCAT	1.00	0.3	8.7	7.6	0.40	10.1	79.3
	MSD	0.82	5.9	17.6	23.5	0.24	18.4	23.9
	TRADES	0.81	21.0	13.2	33.9	0.30	15.2	37.5
	AT-Madry	0.77	26.7	11.7	37.2	0.29	13.0	39.2
Black-Box (L_0, $\epsilon = 10$)	Normal	0.94	66.1	7.4	72.9	0.59	8.3	96.2
	AT-50%	0.91	54.2	15.1	69.6	0.35	16.6	75.7
	AT-100%	0.83	52.8	18.3	70.3	0.29	19.4	72.4
	CCAT	0.99	12.0	8.7	19.6	0.40	10.1	96.6
	MSD	0.86	22.1	17.6	39.8	0.24	18.4	41.0
	TRADES	0.84	8.4	13.2	20.9	0.30	15.2	24.3
	AT-Madry	0.79	15.7	11.7	26.1	0.29	13.0	27.6
Adversarial Frames	Normal	0.28	89.7	7.4	96.7	0.59	8.3	96.9
	AT-50%	0.40	63.6	15.1	78.5	0.35	16.6	78.7
	AT-100%	0.39	62.0	18.3	79.4	0.29	19.4	79.6
	CCAT	0.75	57.4	8.7	65.6	0.40	10.1	86.7
	MSD	0.38	64.7	17.6	82.5	0.24	18.4	82.6
	TRADES	0.45	57.5	13.2	71.3	0.30	15.2	72.1
	AT-Madry	0.48	62.0	11.7	72.9	0.29	13.0	73.3

Table 17: Per-Attack Results on Cifar10, Part IV ($\mathbf{L_0}$, Adversarial Frames). Per-attack results considering PGD-CE, as in (Madry et al., 2018), our PGD-Conf and the remaining black-box attacks for the L_0 threat model and adversarial frames. The used ϵ values are reported in the left-most column. For the black-box attacks, we report the per-example worst-case across all black-box attacks. In addition to FPR and RErr we include ROC AUC, Err as well as Err and RErr in the standard, non-thresholded setting, as reference.

MNIST-C: Supplementary Results for Corruption

		Detection Setting τ@99%TPR					**Standard Setting** τ=0
Corruption	Training	ROC AUC	FPR in %	TNR in %	Err in %	τ	Err in %
all	Normal	0.75	82.8	17.2	31.9	0.98	36.4
	AT-50%	0.80	59.0	41.0	4.0	1.00	26.9
	AT-100%	0.78	63.5	36.5	8.6	1.00	27.4
	CCAT	0.96	29.0	71.0	6.7	0.85	41.5
	MSD	0.75	75.0	25.0	5.4	0.51	14.0
	TRADES	0.79	74.1	25.9	5.5	0.92	16.1
mean	Normal	0.75	82.8	17.2	32.8	1.0	36.3
	AT-50%	0.80	59.0	41.0	12.6	1.0	26.9
	AT-100%	0.78	63.5	36.5	17.6	1.0	27.4
	CCAT	0.96	29.0	71.0	5.7	0.9	41.5
	MSD	0.75	75.0	25.0	6.0	0.5	14.0
	TRADES	0.79	74.1	25.9	7.9	0.9	16.1
brightness	Normal	0.36	100.0	0.0	90.2	0.98	90.2
	AT-50%	0.99	0.9	99.1	36.4	1.00	84.3
	AT-100%	0.97	35.4	64.6	82.2	1.00	90.9
	CCAT	1.00	0.0	100.0	0.0	0.85	90.5
	MSD	0.95	38.7	61.3	0.1	0.51	23.1
	TRADES	0.95	43.7	56.3	67.3	0.92	77.2
canny_edges	Normal	0.91	62.4	37.6	34.8	0.98	45.6
	AT-50%	0.96	28.8	71.2	33.4	1.00	53.2
	AT-100%	0.94	35.3	64.7	38.2	1.00	53.4
	CCAT	0.99	34.3	65.7	70.5	0.85	61.9
	MSD	0.87	63.4	36.6	5.5	0.51	17.8
	TRADES	0.92	53.2	46.8	21.5	0.92	36.3
dotted_line	Normal	0.76	85.4	14.6	2.4	0.98	7.6
	AT-50%	0.77	72.8	27.2	0.9	1.00	7.9
	AT-100%	0.75	74.5	25.5	0.9	1.00	6.4
	CCAT	0.98	53.4	46.6	2.5	0.85	10.1
	MSD	0.60	94.2	5.8	1.1	0.51	2.9
	TRADES	0.73	84.5	15.5	0.5	0.92	4.2
fog	Normal	0.38	99.9	0.1	90.2	0.98	90.2
	AT-50%	0.90	29.7	70.3	10.4	1.00	59.0
	AT-100%	0.88	46.3	53.7	37.7	1.00	62.0
	CCAT	1.00	0.0	100.0	0.0	0.85	90.3
	MSD	0.92	46.8	53.2	4.3	0.51	26.1
	TRADES	0.87	60.9	39.1	14.3	0.92	35.8
glass_blur	Normal	0.87	71.6	28.4	57.2	0.98	56.2
	AT-50%	0.82	67.4	32.6	1.2	1.00	11.0
	AT-100%	0.77	70.5	29.5	1.1	1.00	8.5
	CCAT	1.00	0.0	100.0	0.0	0.85	86.8
	MSD	0.69	89.1	10.9	1.7	0.51	4.9
	TRADES	0.81	80.9	19.1	0.6	0.92	6.0
impulse_noise	Normal	0.87	72.4	27.6	79.2	0.98	81.3
	AT-50%	0.98	13.9	86.1	18.8	1.00	61.4
	AT-100%	0.97	18.8	81.2	12.8	1.00	56.2
	CCAT	1.00	0.0	100.0	0.0	0.85	69.1
	MSD	0.83	69.4	30.6	0.7	0.51	8.2
	TRADES	0.91	57.3	42.7	2.2	0.92	17.2
motion_blur	Normal	0.86	73.9	26.1	29.5	0.98	37.2
	AT-50%	0.62	90.6	9.4	0.3	1.00	2.7
	AT-100%	0.58	91.3	8.7	0.3	1.00	2.4
	CCAT	1.00	0.0	100.0	0.0	0.85	75.6
	MSD	0.74	83.1	16.9	3.6	0.51	9.1
	TRADES	0.71	88.3	11.7	0.6	0.92	4.0
rotate	Normal	0.70	92.4	7.6	1.7	0.98	4.6
	AT-50%	0.64	87.9	12.1	0.8	1.00	4.1
	AT-100%	0.63	87.2	12.8	0.8	1.00	4.5
	CCAT	0.98	41.5	58.5	0.3	0.85	4.4
	MSD	0.64	90.7	9.3	5.7	0.51	9.7
	TRADES	0.66	87.7	12.3	1.4	0.92	4.8

Table 18: Per-Corruptions Results on MNIST-C, PART I. Results on MNIST-C, broken down by individual corruptions (first column); `mean` are the averaged results over all corruptions. We report ROC AUC, FPR and the true negative rate (TNR) in addition to the thresholded and unthresholded Err on the corrupted examples. The table is continued in Tab. 19.

MNIST-C: Supplementary Results for Corruption							
		Detection Setting τ@99%TPR					**Standard Setting** τ=0
Corruption	Training	ROC AUC	FPR in %	TNR in %	Err in %	τ	Err in %
scale	Normal	0.84	89.5	10.5	0.7	0.98	3.1
	AT-50%	0.86	78.5	21.5	0.1	1.00	3.0
	AT-100%	0.80	82.1	17.9	0.4	1.00	3.0
	CCAT	0.75	96.7	3.3	0.8	0.85	2.0
	MSD	0.84	64.4	35.6	8.1	0.51	18.1
	TRADES	0.82	84.4	15.6	0.3	0.92	2.9
shear	Normal	0.60	97.6	2.4	0.2	0.98	0.8
	AT-50%	0.56	95.1	4.9	0.1	1.00	0.9
	AT-100%	0.55	94.8	5.2	0.1	1.00	0.9
	CCAT	0.98	27.4	72.6	0.0	0.85	1.1
	MSD	0.56	95.9	4.1	1.9	0.51	3.2
	TRADES	0.61	94.8	5.2	0.3	0.92	1.4
shot_noise	Normal	0.74	93.0	7.0	1.4	0.98	3.6
	AT-50%	0.62	91.3	8.7	0.2	1.00	1.9
	AT-100%	0.57	92.6	7.4	0.2	1.00	1.8
	CCAT	0.96	61.2	38.8	0.1	0.85	2.3
	MSD	0.59	95.2	4.8	0.9	0.51	2.4
	TRADES	0.65	94.9	5.1	0.2	0.92	1.4
spatter	Normal	0.85	72.8	27.2	18.9	0.98	28.1
	AT-50%	0.65	88.3	11.7	0.6	1.00	3.6
	AT-100%	0.61	90.3	9.7	0.4	1.00	2.7
	CCAT	1.00	0.0	100.0	0.0	0.85	29.7
	MSD	0.56	95.6	4.4	1.2	0.51	2.6
	TRADES	0.63	90.4	9.6	0.7	0.92	3.3
stripe	Normal	0.92	61.1	38.9	69.6	0.98	70.2
	AT-50%	0.98	12.2	87.8	83.8	1.00	81.3
	AT-100%	0.99	12.4	87.6	88.2	1.00	87.4
	CCAT	1.00	0.0	100.0	0.0	0.85	78.0
	MSD	0.90	55.8	44.2	1.1	0.51	10.4
	TRADES	0.88	72.9	27.1	0.6	0.92	5.9
translate	Normal	0.72	95.2	4.8	0.3	0.98	1.6
	AT-50%	0.80	74.6	25.4	0.2	1.00	4.0
	AT-100%	0.79	72.4	27.6	0.2	1.00	4.2
	CCAT	0.84	95.3	4.7	0.3	0.85	1.4
	MSD	0.88	59.3	40.7	44.7	0.51	55.2
	TRADES	0.94	42.3	57.7	6.3	0.92	28.1
zigzag	Normal	0.85	74.5	25.5	16.3	0.98	24.8
	AT-50%	0.87	53.6	46.4	10.9	1.00	24.9
	AT-100%	0.87	48.3	51.7	10.1	1.00	25.8
	CCAT	0.99	25.6	74.4	9.8	0.85	18.6
	MSD	0.70	83.5	16.5	9.7	0.51	15.8
	TRADES	0.76	75.2	24.8	4.9	0.92	13.8

Table 19: Per-Corruptions Results on MNIST-C, PART II. Continued results of Tab. 18 including results on MNIST-C focusing on individual corruptions. textttmean are the averaged results over all corruptions. We report ROC AUC, FPR and the true negative rate (TNR) in addition to the thresholded and unthresholded Err on the corrupted examples.

CIFAR10-C: Supplementary Results for Corruption

		Detection Setting τ@99%TPR					**Standard Setting** τ=0
Corruption	Training	ROC AUC	FPR in %	TNR in %	Err in %	τ	Err in %
all	Normal	0.57	97.1	2.9	12.2	0.59	13.7
	AT-50%	0.53	96.2	3.8	16.2	0.35	18.1
	AT-100%	0.53	97.2	2.8	19.6	0.29	20.9
	CCAT	0.66	72.1	27.9	10.4	0.40	27.2
	MSD	0.53	98.2	1.8	19.2	0.24	20.2
	TRADES	0.53	95.8	4.2	15.0	0.30	17.3
	AT-Madry	0.53	96.3	3.7	12.8	0.29	14.7
mean	Normal	0.57	97.1	2.9	12.3	0.6	13.7
	AT-50%	0.53	96.2	3.8	16.2	0.3	18.1
	AT-100%	0.53	97.2	2.8	19.6	0.3	21.0
	CCAT	0.66	72.1	27.9	8.5	0.4	27.2
	MSD	0.53	98.2	1.8	19.3	0.2	20.2
	TRADES	0.53	95.8	4.2	15.0	0.3	17.3
	AT-Madry	0.53	96.3	3.7	12.9	0.3	14.7
brightness	Normal	0.50	98.1	1.9	7.5	0.59	8.4
	AT-50%	0.50	97.0	3.0	14.9	0.35	16.5
	AT-100%	0.50	97.9	2.1	18.2	0.29	19.2
	CCAT	0.54	94.8	5.2	8.2	0.40	10.4
	MSD	0.50	98.6	1.4	17.8	0.24	18.5
	TRADES	0.49	96.8	3.2	13.1	0.30	14.9
	AT-Madry	0.49	97.5	2.5	11.2	0.29	12.5
contrast	Normal	0.52	98.1	1.9	8.4	0.59	9.4
	AT-50%	0.60	94.1	5.9	17.1	0.35	20.0
	AT-100%	0.60	95.2	4.8	21.1	0.29	23.5
	CCAT	0.55	96.6	3.4	10.3	0.40	11.9
	MSD	0.60	97.3	2.7	22.2	0.24	23.5
	TRADES	0.58	94.4	5.6	17.2	0.30	20.2
	AT-Madry	0.58	94.6	5.4	13.5	0.29	16.3
defocus_blur	Normal	0.50	98.1	1.9	7.4	0.59	8.4
	AT-50%	0.51	96.8	3.2	15.5	0.35	17.2
	AT-100%	0.51	97.7	2.3	18.6	0.29	19.7
	CCAT	0.49	97.5	2.5	9.2	0.40	10.5
	MSD	0.51	98.5	1.5	18.2	0.24	19.0
	TRADES	0.51	96.3	3.7	13.7	0.30	15.7
	AT-Madry	0.51	96.9	3.1	12.0	0.29	13.6
elastic_transform	Normal	0.60	97.3	2.7	12.1	0.59	13.5
	AT-50%	0.57	95.8	4.2	19.0	0.35	21.1
	AT-100%	0.57	96.7	3.3	22.0	0.29	23.6
	CCAT	0.54	96.5	3.5	13.2	0.40	14.9
	MSD	0.57	98.0	2.0	21.0	0.24	22.2
	TRADES	0.56	95.0	5.0	17.5	0.30	20.0
	AT-Madry	0.58	96.0	4.0	15.5	0.29	17.6
fog	Normal	0.51	98.0	2.0	7.9	0.59	8.8
	AT-50%	0.57	94.7	5.3	15.7	0.35	18.3
	AT-100%	0.57	96.1	3.9	19.5	0.29	21.5
	CCAT	0.55	96.0	4.0	9.0	0.40	11.1
	MSD	0.57	97.5	2.5	19.7	0.24	21.0
	TRADES	0.55	95.2	4.8	15.0	0.30	17.5
	AT-Madry	0.55	95.3	4.7	12.2	0.29	14.7
frost	Normal	0.57	97.2	2.8	11.5	0.59	12.8
	AT-50%	0.53	95.9	4.1	16.0	0.35	18.0
	AT-100%	0.53	96.9	3.1	20.1	0.29	21.6
	CCAT	0.65	88.1	11.9	9.0	0.40	12.5
	MSD	0.52	98.2	1.8	20.6	0.24	21.4
	TRADES	0.51	96.0	4.0	14.7	0.30	16.8
	AT-Madry	0.51	96.7	3.3	12.1	0.29	13.6
gaussian_blur	Normal	0.50	98.0	2.0	7.4	0.59	8.4
	AT-50%	0.51	96.8	3.2	15.5	0.35	17.1
	AT-100%	0.51	97.7	2.3	18.6	0.29	19.7
	CCAT	0.49	97.4	2.6	9.2	0.40	10.4
	MSD	0.51	98.5	1.5	18.1	0.24	19.0
	TRADES	0.51	96.3	3.7	13.7	0.30	15.7
	AT-Madry	0.51	96.9	3.1	12.0	0.29	13.5
gaussian_noise	Normal	0.62	96.2	3.8	15.7	0.59	17.7
	AT-50%	0.51	96.8	3.2	15.5	0.35	17.0
	AT-100%	0.51	97.7	2.3	18.6	0.29	19.8
	CCAT	1.00	0.0	100.0	0.0	0.40	84.9
	MSD	0.51	98.5	1.5	18.6	0.24	19.3
	TRADES	0.52	96.0	4.0	14.3	0.30	16.5
	AT-Madry	0.52	96.7	3.3	12.2	0.29	13.9
glass_blur	Normal	0.76	92.5	7.5	40.8	0.59	43.1
	AT-50%	0.58	95.4	4.6	18.0	0.35	20.2
	AT-100%	0.56	97.2	2.8	21.1	0.29	22.4
	CCAT	0.93	31.1	68.9	14.7	0.40	31.8
	MSD	0.56	98.1	1.9	21.0	0.24	22.0
	TRADES	0.56	95.1	4.9	17.6	0.30	20.1
	AT-Madry	0.58	95.5	4.5	15.4	0.29	17.6

Table 20: Per-Corruptions Results on Cifar10-C, PART I. Results on Cifar10-C focusing on individual corruptions (first column); textttmean are the averaged results over all corruptions. We report ROC AUC, FPR and the true negative rate (TNR) in addition to the thresholded and unthresholded Err on the corrupted examples. The table is continued in Tab. 21.

CIFAR10-C: Supplementary Results for Corruption							
		Detection Setting τ@99%TPR					**Standard Setting** τ=0
Corruption	Training	ROC AUC	FPR in %	TNR in %	Err in %	τ	Err in %
impulse_noise	Normal	0.59	97.0	3.0	13.1	0.59	14.5
	AT-50%	0.53	96.4	3.6	16.4	0.35	17.9
	AT-100%	0.52	97.6	2.4	19.8	0.29	21.1
	CCAT	1.00	0.1	99.9	0.0	0.40	61.0
	MSD	0.51	98.4	1.6	18.3	0.24	19.2
	TRADES	0.54	94.9	5.1	15.7	0.30	18.4
	AT-Madry	0.53	96.1	3.9	13.3	0.29	15.3
jpeg_compression	Normal	0.59	96.9	3.1	12.3	0.59	13.7
	AT-50%	0.51	96.8	3.2	15.5	0.35	17.0
	AT-100%	0.51	97.6	2.4	18.9	0.29	20.2
	CCAT	0.59	93.3	6.7	9.3	0.40	12.1
	MSD	0.51	98.3	1.7	18.3	0.24	19.3
	TRADES	0.51	96.3	3.7	14.4	0.30	16.6
	AT-Madry	0.52	96.5	3.5	12.6	0.29	14.4
motion_blur	Normal	0.58	97.3	2.7	10.9	0.59	12.2
	AT-50%	0.55	95.9	4.1	16.8	0.35	18.9
	AT-100%	0.54	97.1	2.9	19.9	0.29	21.3
	CCAT	0.52	96.5	3.5	11.9	0.40	13.6
	MSD	0.54	98.1	1.9	19.7	0.24	20.7
	TRADES	0.54	95.8	4.2	15.7	0.30	17.9
	AT-Madry	0.55	95.9	4.1	13.3	0.29	15.6
pixelate	Normal	0.54	97.6	2.4	9.7	0.59	10.9
	AT-50%	0.51	96.7	3.3	15.3	0.35	17.0
	AT-100%	0.51	97.8	2.2	18.6	0.29	19.7
	CCAT	0.52	97.0	3.0	9.2	0.40	10.8
	MSD	0.51	98.4	1.6	18.2	0.24	19.1
	TRADES	0.51	96.4	3.6	14.0	0.30	16.0
	AT-Madry	0.51	97.1	2.9	12.1	0.29	13.6
saturate	Normal	0.55	97.5	2.5	10.0	0.59	11.3
	AT-50%	0.55	95.3	4.7	18.2	0.35	20.5
	AT-100%	0.54	96.0	4.0	21.3	0.29	23.3
	CCAT	0.48	97.6	2.4	11.8	0.40	13.0
	MSD	0.55	97.3	2.7	20.9	0.24	22.1
	TRADES	0.53	95.0	5.0	15.6	0.30	18.2
	AT-Madry	0.54	95.2	4.8	14.2	0.29	16.4
shot_noise	Normal	0.58	97.0	3.0	12.1	0.59	13.7
	AT-50%	0.51	97.1	2.9	15.4	0.35	16.7
	AT-100%	0.50	97.7	2.3	18.5	0.29	19.6
	CCAT	1.00	0.0	100.0	0.0	0.40	84.5
	MSD	0.51	98.5	1.5	18.1	0.24	18.9
	TRADES	0.51	96.2	3.8	13.7	0.30	15.8
	AT-Madry	0.51	97.1	2.9	11.8	0.29	13.4
snow	Normal	0.57	97.3	2.7	12.1	0.59	13.5
	AT-50%	0.51	96.7	3.3	15.3	0.35	17.0
	AT-100%	0.51	97.5	2.5	18.8	0.29	20.1
	CCAT	0.58	95.6	4.4	11.2	0.40	13.3
	MSD	0.51	98.4	1.6	18.7	0.24	19.6
	TRADES	0.50	96.3	3.7	14.1	0.30	16.1
	AT-Madry	0.51	97.0	3.0	12.3	0.29	13.8
spatter	Normal	0.54	97.5	2.5	9.4	0.59	10.6
	AT-50%	0.51	96.8	3.2	15.5	0.35	17.2
	AT-100%	0.51	97.6	2.4	18.5	0.29	19.8
	CCAT	0.58	95.2	4.8	9.3	0.40	11.6
	MSD	0.51	98.3	1.7	18.0	0.24	18.8
	TRADES	0.51	96.0	4.0	14.2	0.30	16.4
	AT-Madry	0.51	96.7	3.3	12.3	0.29	13.9
speckle_noise	Normal	0.57	97.0	3.0	12.4	0.59	13.8
	AT-50%	0.51	97.1	2.9	15.2	0.35	16.7
	AT-100%	0.51	97.7	2.3	18.4	0.29	19.5
	CCAT	1.00	0.0	100.0	0.0	0.40	83.0
	MSD	0.51	98.4	1.6	17.9	0.24	18.8
	TRADES	0.51	96.4	3.6	13.8	0.30	15.9
	AT-Madry	0.51	96.9	3.1	12.0	0.29	13.4
zoom_blur	Normal	0.61	96.7	3.3	12.9	0.59	14.6
	AT-50%	0.56	95.7	4.3	17.2	0.35	19.2
	AT-100%	0.56	97.0	3.0	21.1	0.29	22.4
	CCAT	0.52	97.1	2.9	13.9	0.40	15.6
	MSD	0.56	98.0	2.0	20.5	0.24	21.5
	TRADES	0.55	95.4	4.6	17.0	0.30	19.2
	AT-Madry	0.56	95.7	4.3	14.3	0.29	16.4

Table 21: Per-Corruptions Results on Cifar10-C, PART II Continued results of Tab. 20 including results on Cifar10-C focusing on individual corruptions. textttmean are the averaged results over all corruptions. We report ROC AUC, FPR and the true negative rate (TNR) in addition to the thresholded and unthresholded Err on the corrupted examples.

References

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

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *SP*, 2017.

Croce, F. and Hein, M. Sparse and imperceivable adversarial attacks. In *ICCV*, 2019.

Duchi, J. C., Shalev-Shwartz, S., Singer, Y., and Chandra, T. Efficient projections onto the l_1-ball for learning in high dimensions. In *ICML*, 2008.

Goodfellow, I., Qin, Y., and Berthelot, D. Evaluation methodology for attacks against confidence thresholding models, 2019. URL https://openreview.net/forum?id=H1g0piA9tQ.

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

Hendrycks, D. and Dietterich, T. G. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv.org*, abs/1903.12261, 2019.

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

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

Khoury, M. and Hadfield-Menell, D. On the geometry of adversarial examples. *arXiv.org*, abs/1811.00525, 2018.

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

Lee, K., Lee, K., Lee, H., and Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*, 2018.

Ma, X., Li, B., Wang, Y., Erfani, S. M., Wijewickrema, S. N. R., Schoenebeck, G., Song, D., Houle, M. E., and Bailey, J. Characterizing adversarial subspaces using local intrinsic dimensionality. In *ICLR*, 2018.

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. *ICML*, 2020.

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.

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.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *JMLR*, 12:2825–2830, 2011.

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

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

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. *arXiv.org*, abs/1805.12152, 2018.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In *BMVC*, 2016.

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.