Bergische Universität Wuppertal

Fakultät für Mathematik und Naturwissenschaften

Institute of Mathematical Modelling, Analysis and Computational Mathematics (IMACM)

Preprint BUW-IMACM 20/41

M. Rottmann, M. Peyron, N. Krejic and H. Gottschalk

Detection of Iterative Adversarial Attacks via Counter Attack

September 28, 2020

http://www.imacm.uni-wuppertal.de

Detection of Iterative Adversarial Attacks via Counter Attack

Matthias Rottmann[*], Mathis Peyron[†], Natasa Krejic[‡] and Hanno Gottschalk[*]

Abstract

Deep neural networks (DNNs) have proven to be powerful tools for processing unstructured data. However for high-dimensional data, like images, they are inherently vulnerable to adversarial attacks. Small almost invisible perturbations added to the input can be used to fool DNNs. Various attacks, hardening methods and detection methods have been introduced in recent years. Notoriously, Carlini-Wagner (CW) type attacks computed by iterative minimization belong to those that are most difficult to detect. In this work, we demonstrate that such iterative minimization attacks can by used as detectors themselves. Thus, in some sense we show that one can fight fire with fire. This work also outlines a mathematical proof that under certain assumptions this detector provides asymptotically optimal separation of original and attacked images. In numerical experiments, we obtain AUROC values up to 99.73% for our detection method. This distinctly surpasses state of the art detection rates for CW attacks from the literature. We also give numerical evidence that our method is robust against the attacker's choice of the method of attack.

1 Introduction

For many applications, deep learning has shown to outperform concurring machine learning approaches by far [18, 36, 16]. Especially, when working with high-dimensional input data like images, deep neural networks (DNNs) are show impressive results. As discovered by Szegedy et al. [38], this remarkable performance comes with a downside. Very small (noise-like) perturbation added to an input image can result in incorrect predictions with high confidence [38, 12, 28, 3, 27]. Such adversarial attacks are usually crafted by performing a constrained gradient-descent procedure with respect to the input data in order to change the class predicted by the DNN and at the same time modify the image by the least possible amount measured. The new class can be either a class of choice (targeted attack) or an arbitrary but different one (untargeted attack). Many other types of attacks such as the fast signed gradient method [12] and DeepFool [28] have been introduced, but these methods do not fool DNNs reliably.

Carlini & Wagner (CW) extended [38] with a method that reliably attacks deep neural networks while controlling important features of the attack like sparsity and maximum size over all pixels, see [3]. This method aims at finding a targeted or an untargeted attack where the distance between the attacked image and the original one is minimal with respect to a chosen ℓ_p distance. Distances of choice within in CW-type frameworks are mostly ℓ_p with $p = 0, 2, \infty$. Mixtures of $p = 1, 2$ have also been proposed by some authors [5]. Except for the ℓ_0 distance, these attacks are perceptually extremely hard to detect, cf. figures 1 and 2. Minimization of the ℓ_0 distance minimizes the number of pixels changed, but also changes these pixels by the maximally and the resulting spikes are easy to detect. This changes for $p > 0$.

Typically, one distinguishes between three different attack scenarios:

- *white box* attack: the attacker has access to the DNN's parameters / the whole framework including defense strategies.
- *black box* attack: the attacker does not know the DNN's parameters / the whole framework including defense strategies.
- *gray box* attack: in-between white and black box, e.g. the attacker might know the framework but not the parameters used.

The CW attack currently is one of the most efficient white-box attacks.

Defense Methods. Several defense mechanisms have been proposed to either harden neural networks or to detect adversarial attacks. One such hardening method is the so-called *defensive distillation* [29]. With a single retraining step this method provides strong security, however not against CW attacks. Training for robustness via adversarial training is another popular defense approach, see e.g. [12, 25, 41, 19, 26]. See also [31] for an overview regarding defense methods.

Most of these methods are not able to deal with attacks based on iterative minimization like CW attacks. In a white box setting the hardened networks can still be subject to fooling.

Detection Methods. There are numerous detection methods for adversarial attacks. In many works, it has been observed and exploited that adversarial attacks are

[*] University of Wuppertal, School of Mathematics and Natural Sciences, {rottmann,hgottsch}@uni-wuppertal.de

[†] ENSEEIHT Toulouse, Department of HPC and Big Data, mathis.peyron@etu.enseeiht.fr

[‡] University of Novi Sad, Faculty of Sciences, natasa@dmi.uns.ac.rs

less robust to random noise than non-attacked clean samples, see e.g. [1, 42, 39, 32]. This robustness issue can be utilized by detection and defense methods. In [32], a statistical test is proposed for a white-box setup. Statistics are obtained under random corruptions of the inputs and observing the corresponding softmax probability distributions. In [39], noise injection is combined with a cryptography component by adding key-defined randomization to an ensemble of networks at both training and test time. The authors assume in a gray-box setup that the networks, attacks as well as the whole framework are known to the attacker, however the keys are not. Another popular approach to detecting adversarial examples is based on filtering certain frequencies or detecting them right in the input image. An adaptive noise reduction approach for detection of adversarial examples is presented in [21]. JPEG compression ([30, 23]), similar input transformations ([15]) and other filtering techniques ([20]) have demonstrated to filter out many types of adversarial attacks as well. In [10], adversarial images are detected based Bayesian uncertainty. It has also been demonstrated that image transformations such as rotations can be used [40] for filtering adversarial attacks. In [22], detection is performed from a steganalysis point of view by calculating filter statistics, the dependence between pixels is then modeled via higher order Markov chains. In [46] it was discovered that eigenvectors corresponding to large eigenvalues of the fisher information metric yield adversarial attacks and that adversarial attacks in turn can be detected by means of their spectral properties. Apart from that, also saliency maps can be used to detect adversarial examples, see [44].

Some approaches also work with several images or sequences of images to detect adversarial images, see e.g. [6, 14]. Approaches that not only take input or output layers into account, but also hidden layer statistics, are introduced in [4, 47]. Auxiliary models trained for robustness and equipped for detecting adversarial examples are presented in [45].

Recently, GANs have demonstrated to defend DNNs against adversarial attacks very well, see [33]. This approach called defense-GAN iteratively filters out the adversarial perturbation. The filtered result is then presented to the original classifier.

Semantic concepts in image classification tasks preselect image data that only cover a tiny fraction of the expressive power of rgb images. Therefore training data sets can be embedded in lower dimensional manifolds and DNNs are only trained with data within the manifolds and behave arbitrarily in perpendicular directions. These are also the directions used by adversarial perturbations. Thus, the manifold distance of adversarial examples can be used as criterion for detection in [17].

As opposed to many of the adversarial training and hardening approaches, most of the works mentioned in this paragraph are able to detect CW attacks with AUROC values up to 99% [24]. For an overview we refer to [43].

Our contribution. In this paper we present a novel approach, how to use Carlini's & Wagner's attack framework [3] not only to generate attacks, but also as a very efficient detector for CW white box attacks. This statement holds for any CW-type of attack generated by an iterative minimization process. In this sense we fight fire with fire.

A CW attack aims at finding the smallest perturbation in a given ℓ_p norm, such that a given image is pushed across the closest decision boundary. Applying another CW attack to the perturbed image moves this image back across the same decision boundary. Importantly, this second attack oftentimes generates much smaller perturbations. It is therefore properly the optimality of the CW attack that leaves a treacherous footprint in the attacked data. The perturbation is the easier to detect, the more optimal it is.

For this mechanism, we outline a mathematical proof (for the untargeted ℓ_2 case) that in the limit where the number of iterations of the original CW attack tends to infinity, by the counter attack method separation of attacked and non-attacked images is possible with an area under the receiver operator curve (AUROC) tending to one. We therefore for the first time provide a detection method for the CW ℓ_2 attack that is provably efficient. This result is based on the mathematical characterization of the stationary points and convergence for the CW attack that is new by itself. We also demonstrate that this (asymptotic) mathematical statement passes the numerical tests to a high extent. To this end, we threshold on the ℓ_p norms of the computed perturbations of first and second attack and find that statistical separation is possible with AUROC values well above those of many competing detection methods. This finding holds for both, targeted and untargeted attacks. We show results for CIFAR10 and ImageNet and obtain AUROC values of up to 99.64% for CIFAR10 and 99.73% for ImageNet when using the ℓ_2 norm. Our experiments show that the robustness under attacks with different ℓ_p norms can be detected very well, thus we can reliably detect iterative minimization attacks without knowing the actual parameter p of the original attack.

Related work. Unlike in [1, 42, 39, 32, 40], we do not apply any additional perturbation or transformation to the inputs of DNNs to detect adversarial examples. We also do not filter noisy signals in images like [21].

We detect adversarial attacks by measuring the norm of an adversarial perturbation obtained by a second attack and afterwards thresholding on it. Thus, the works closest to ours can be considered those that are based on distance measures, see [4, 47, 17]. However, [4, 47] work on the level of feature maps in hidden layers. All three works do not use a second adversarial attack to detect an attack. In spirit, [17] can be regarded to be closest to our approach. The authors measure the distance from a manifold containing the data, therefore a model of the manifold is learned. Our method can be interpreted as thresholding on the distance that is required to find the closest decision boundary outside the given manifold that

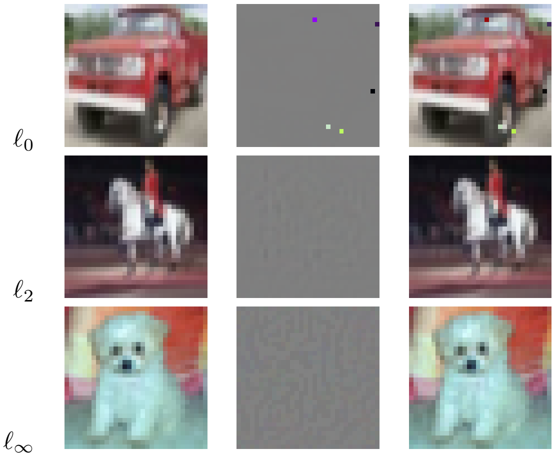

Figure 1: An illustration of attacked CIFAR10 images. (left): the input image, (center): the added noise and (right): the resulting adversarial image.

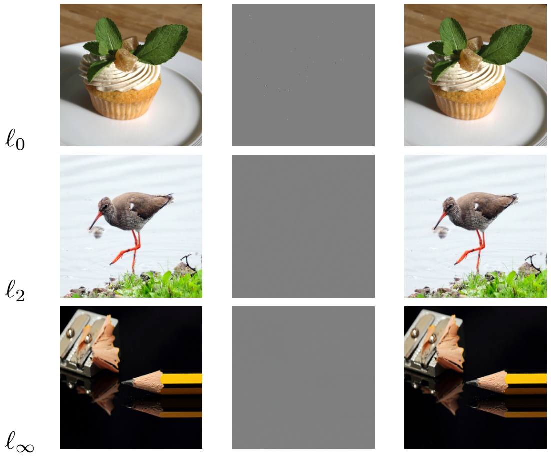

Figure 2: An illustration of attacked ImageNet images. (left): the input image, (center): the added noise and (right): the resulting adversarial image.

contains the image data. Unlike in [17], our approach does not require a model of the manifold and is rather based the intrinsic properties of CW attacks and other iterative minimization attacks. During the finalization of this paper, we found a work [48] that also pursues a counter attack idea. However in this work, distance is considered in terms of Kullback-Leibler divergence of softmax probabilities while we consider different norms on the input space. The authors also trained a model with adversarial examples whereas we only threshold on distance measures. Furthermore, we provide a general theoretical statement that our method is provably efficient. We conducted tests also with high definition ImageNet2012 data. The numerical results for CIFAR10 are in the same range.

Organization of this work. The remainder of this work is structured as follows: In section 2 we introduce our detection method an provide a theoretical statement that under certain assumptions a second identical attack performed for an already attacked image is an ideal detector for a given CW attack with norm ℓ_p. In section section 3 we demonstrate that this finding also applies to numerical experiments. We demonstrate, that once and twice attacked images can be well separated by measuring the distances between the attack's input and output in the ℓ_p-norm. For our evaluation we use the CIFAR10 test set as well as the ImageNet test set. We also show, that any CW attack with a norm ℓ_p can be detected reliably by performing another CW attack with a different norm $\ell_{p'}$. We also show that our approach works well for both targeted and untargeted attacks.

2 Detection by Counter Attack

Let $C = \{1, \dots, c\}$ denote the set of $2 \leq c \in \mathbb{N}$ distinct classes. and let $I = [0, 1]$ denote the closed unit interval. An image is an element $x \in I^n$. Let $\varphi : I^n \to I^c$ be a continuous function given by a DNN, almost everywhere differentiable, that maps x to a probability distribution $y = \varphi(x) \in I^c$ with $\sum_{i=1}^{c} y_i = 1$ and $y_i \geq 0$. Furthermore, $\kappa : I^n \to C$ denotes the map that yields the corresponding class index, i.e.,

$$\kappa(x) = \begin{cases} i & \text{if } y_i > y_j \ \forall j \in C \setminus \{i\} \\ 0 & \text{else.} \end{cases} \tag{1}$$

This is a slight modification of the arg max function as $K_i = \{x \in I^n : \kappa(x) = i\}$ for $i > 0$ gives the set of all $x \in I^n$ predicted by φ to be a member of class i and for $i = 0$ we obtain the set of all class boundaries with respect to φ.

Given $x \in \mathbb{R}^n$, the ℓ_p "norm" is defined as follows:

- for $p \in \mathbb{N}$: $\|x\|_p = (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}}$,
- for $p = 0$: $\|x\|_0 = |\{x_i > 0\}|$,
- for $p = \infty$: $\|x\|_\infty = \max_i |x_i|$.

The corresponding ℓ_p distance measure is given by $dist_p(x, x') = \|x - x'\|_p$ and the n-dimensional open ℓ_p-neighborhood with radius ε and center point $x_0 \in \mathbb{R}^n$ is defined as $B_p(x_0, \varepsilon) = \{x \in \mathbb{R}^n : dist_p(x, x_0) < \varepsilon\}$.

The CW Attack. For any image $x_0 \in I^n$ the Carlini & Wagner (CW) attack introduced in [3] can be formulated as the following optimization problem:

$$\begin{aligned} x_k = \underset{x \in \mathbb{R}^n}{\arg\min}\ & dist_p(x_0, x) \\ \text{s.th.}\ \ & \kappa(x) \neq \kappa(x_0)\ x \in I^n . \end{aligned} \tag{2}$$

In order to obtain $\kappa(x') \neq \kappa(x)$, relaxations given by a choice of differentiable terms $f(x)$ are introduced. The box constraint $x \in I^n$ is enforced by a simple projection, i.e., clipping the gradient. Both loss functions are jointly minimized in a scalarized setting, i.e.,

$$F(x) := dist_p(x_0, x) + a\, f(x) \to \min . \tag{3}$$

In their experiments, Carlini & Wagner perform a binary search for the constant a such that $\kappa(x) \neq \kappa(x_0)$ is almost

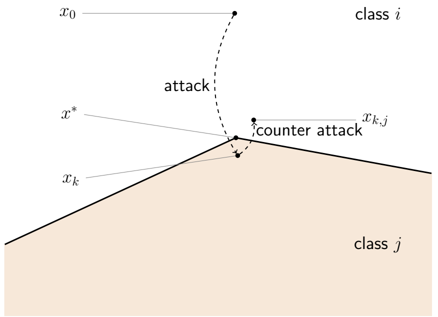

Figure 3: Sketch of the action of the counter attack, x^* is the stationary point of the first attack, x_k the final iterate of the first attack and $x_{k,j}$ the final iterate of the second attack.

always satisfied. For further algorithmic details we refer to [3]. Two illustrations of attacked images are given in figures 1 and 2. In particular for the ImageNet dataset in figure 2, even the perturbations themselves are imperceptible.

The CW Counter Attack. The goal of the CW attack is to estimate the distance of x_0 to the closest class boundary via $dist_p(x_0, x_k)$. The more eager the attacker minimizes the perturbation $x_k - x_0$ in a chosen p-norm, the more likely it is that x_k is close to a class boundary. Hence, when estimating the distance of x_k to the closest class boundary by performing another CW attack which yields an $x_{k,j}$, it is likely that

$$dist_p(x_k, x_{k,j}) \ll dist_p(x_0, x_k)\,, \tag{4}$$

cf. figure 3. This motivates our claim, that the CW attack itself is a good detector for CW attacks. An image $x_0 \in I^n$, that has not been attacked, is likely to have a greater distance to the closest class boundary than an $x_k \in I^n$ which has already been subject to a CW attack. In practice, it cannot be guaranteed that the CW attack finds a point x_k which is close to the closest boundary. However, an interesting statements can be made on which we elaborate in the following paragraph.

Theoretical Considerations. In this paragraph we outline the proof of the statement that asymptotically, i.e., for very eager attackers, the counter attack tends to detect all attacked samples with probability 1. However, a clear and formal proof of all aspects regarding this statement is beyond the scope of this paper and therefore left for future work.

We assume that $p = 2$ and fix a choice for f which is

$$f(x) = \max_{i \neq t}\{Z_t(x) - Z_i(x),\, 0\} \tag{5}$$

where Z denotes the neural network φ but without the final softmax activation function. Note that equation (5) is a construction for untargeted attack. Choosing a penalty term for a targeted attack does not affect the arguments provided in this section.

Furthermore, let

- $t := \kappa(x_{orig})$ the original class,
- $\mathcal{F} := \{x \in I^n \,:\, \kappa(x) \neq t\}$ the feasible region,
- $\mathcal{NF} := \{x \in I^n \,:\, \kappa(x) = t\}$ the infeasible region,
- $\partial\mathcal{F} = \partial\mathcal{NF}$ the class boundary between $\mathcal{F}$ and $\mathcal{NF}$.

It is common knowledge that I^n can be decomposed into a finite number of polytopes $\mathcal{Q} := \{Q_j \,:\, j = 1, \ldots, s\}$ such that Z is affine linear on each Q_j, see [8, 35]. In [8], each of the polytopes Q_j is expressed by a set of linear constraints, therefore being *convex*.

For a locally Lipschitz function g, the *Clarke generalized gradient* is defined by

$$\nabla_C g(x) := \operatorname{conv}\{\lim_{i\to\infty} \nabla g(x_i) : x_i \to x \text{ and } \nabla g(x_i) \text{ exists}\}\,. \tag{6}$$

Let S be the stationary set of problem (2),

$$S = \{x \in I^n : 0 \in \nabla_C F(y) + N_{I^n}(x)\} \tag{7}$$

where $\nabla_C F(x)$ is the set of all generalized gradients and $N_{I^n}(x)$ is the normal cone to the set I^n at point x. In general, computing the set of generalized gradients is not an easy task, but given the special structure of f – piece-wise linear – it can be done relatively easily in this case. Namely, the set $\nabla_C f(x)$ is a convex hull of the gradients of linear functions that are active at the point x, [34]. Therefore, by [7, Corollary2, p.39], the generalized gradient $G(x)$ of $F(x)$ has the form

$$G(x) = \{ag(x) + 2(x - x_{orig}),\ g(x) \in \nabla_C f(x)\}\,. \tag{8}$$

The projected generalized gradient method [37] is defined as follows. Let $P(x)$ denote the orthogonal projection of x onto I^n. Given the learning schedule $\{\alpha_k\}$ such that

$$\lim \alpha_k = 0 \text{ and } \sum_{k=1}^{\infty} \alpha_k = \infty, \tag{9}$$

the iterative sequence is generated as

$$x_{k+1} = P(x_k - \alpha_k G_k),\ G_k \in \nabla_C F(x^k),\ k = 0, 1, \ldots \tag{10}$$

In this paper we add the following condition for the learning rate

$$\sum_{k=1}^{\infty} \alpha_k^2 < \infty\,, \tag{11}$$

which strengthens the conditions (9) and does not alter the statements from [37].

It can be shown that every accumulation point of the sequence $\{x_k\}$ generated by (10) converges to a stationary point of F. The proof of this theorem can be found in [37] and only requires that the set S only contains isolated points from I^n, which can be shown with moderate effort exploiting the piece-wise linear structure of f.

Now, considering an iterate $x_k \in \mathcal{F}$, it is also clear that reducing $F(x_k)$ means decreasing the distance of x_k to x_0, provided α_{k+1} is sufficiently small. Therefore, $x_k \in \mathcal{F}$ cannot be a stationary point. Assume that the CW attack converges to x^* and is successful, i.e., $x^* \notin \mathcal{NF}$. This implies $x^* \in \partial\mathcal{F}$. Summarizing this, any

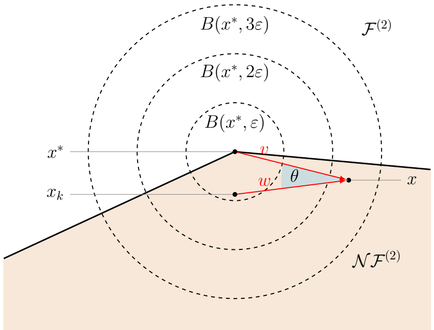

Figure 4: Situation present in the counter attack. By bounding $\cos(\theta)$ from below we show that x reduces its distance to x^* when performing a gradient descent step under the given assumptions. Therefore, x never leaves $B(x^*,3\varepsilon)$.

successful CW attack converges to a stationary point on the boundary $\partial\mathcal{F}$ of the feasible set.

Let us now consider the counter attack and therefore let $\mathcal{F}^{(2)} \subset \mathcal{NF}^{(1)} := \mathcal{NF}$ denote the counter attack's feasible region and $\mathcal{NF}^{(2)} \subset \mathcal{F}^{(1)} := \mathcal{F}$ the infeasible region. We seek to minimize the functional

$$F^{(2)}(x) := \mathrm{dist}_p(x_k, x)^p + b f^{(2)}(x) \tag{12}$$

where x^* is the stationary point that we are approaching in the first iterative attack. For now, let $p = 2$. Let $\{Q_1,\dots,Q_s\}$ be the set of all polytopes that fulfill $x^* \in Q_i$, $i = 1,\dots,s$. We assume that the first attack is successful and that there exists an iterate x_k of the first attack such that $\mathrm{dist}(x^*, x_k) < \varepsilon$ and furthermore we assume that $B(x^*,3\varepsilon) \subset \bigcup_{i=1}^{s} Q_i$.

Theorem 1. *By the preceding assumptions and further assuming, that the counter attack starts at $x_k \in \mathcal{NF}^{(2)}$, stops when reaching $\mathcal{F}^{(2)}$ and uses a sufficiently small step size α, then the counter attack iteration minimizing $F^{(2)}$ never leaves $B(x^*,3\varepsilon)$.*

Proof. Let $x \in B(x^*,3\varepsilon)\backslash B(x^*,2\varepsilon)$ and $G^{(2)}(x) \in \nabla_C F^{(2)}(x)$ any gradient with corresponding $g^{(2)}(x) \in \nabla_C f^{(2)}(x)$ which is fixed for x being member of a chosen polytope Q_i. We can assume that $x \in \mathcal{NF}^{(2)\circ}$ since we are done otherwise. Furthermore, let $v := x_k - x$, $w := x^* - x$ and $\cos(\theta) = \frac{v^T w}{\|v\|\|w\|}$ where $\|\cdot\| = \|\cdot\|_2$. This implies $x^* - x_k = w - v$ and

$$\begin{aligned}\cos(\theta) &= \frac{\|v\|^2 + \|w\|^2 - \|w-v\|^2}{2\,\|v\|\,\|w\|} \\ &\geq \frac{4\varepsilon^2 + \varepsilon^2 - \varepsilon^2}{2\cdot 3\varepsilon \cdot 4\varepsilon} = \frac{1}{6}\end{aligned} \tag{13}$$

Since $G^{(2)}(x) = 2(x - x_k) + b\, g^{(2)}(x)$, we obtain

$$\begin{aligned}&G^{(2)}(x)^T(x - x^*) \\ &= 2(x - x_k)^T(x - x^*) + b\,\underbrace{g^{(2)}(x)^T(x - x^*)}_{>0} \\ &> 2(x - x_k)^T(x - x^*) \geq\ 2\varepsilon^2\cos(\theta) = \frac{\varepsilon^2}{3}\,.\end{aligned} \tag{14}$$

Note that, $g^{(2)}(x)^T(x - x^*) > 0$ follows from $x - x^*$ being an ascent direction. This holds due to $x \in Q_i \cap \mathcal{NF}^{(2)\circ}$ which implies $f^{(2)}(x) > 0$ and the fact that $x^* \in Q_i$ as well and $f^{(2)}(x^*) = 0$. Hence, the difference in distance to x^* when performing a gradient descent step is

$$\begin{aligned}&\|x - x^*\|^2 - \left\|x - x^* - \alpha G^{(2)}(x)\right\|^2 \\ &= 2\alpha G^{(2)}(x)^T(x - x^*) - \alpha^2 \underbrace{\left\|G^{(2)}(x)\right\|^2}_{\leq C(b)} \\ &\geq 2\alpha \cdot \frac{\varepsilon^2}{3} - \alpha^2 C(b)\,.\end{aligned} \tag{15}$$

The latter expression is greater than zero iff the step-size schedule α is small enough, i.e., $\alpha < \frac{2\varepsilon^2}{3C(b)}$. Hence, the distance to x^* decreases. □

We now take into account the stochastic effects that stem from choosing an arbitrary initial image $x \in I^n$ represented by a random variable X. Let k be the number of iterates of the original CW attack and let $X_k \in \mathcal{F}^{(2)}$ be the final iterate after $k \geq 1$ iterations, starting at $X_0 = X$. For any random variables $Y, Z \in I^n$ let

$$\begin{aligned}D(Y) &= \mathrm{dist}(Y,\mathcal{F}) = \inf_{y\in\mathcal{F}} \mathrm{dist}(Y,y) \quad \text{and} \\ D^{(0,k)} &= \mathrm{dist}(X_0, X_k) \geq D(X_0)\,,\end{aligned} \tag{16}$$

provided $X_k \in \mathcal{F}$, which means that the k-th iterate is a successful attack. For $\delta \in \mathbb{R}$, we define the distribution function corresponding to a random variable D (representing a random distance) by $F_D(\delta) = P(D \leq \delta) = D_*P((-\infty,\delta])$ where the push forward measure is $D_*P(B) = P(D^{-1}(B))$ for all B in the Borel σ-algebra. The area under receiver operator characteristic curve of $D(Y)$ and $D(Z)$ is defined by

$$\mathrm{AUROC}\,(D(Y), D(Z)) = \int_{\mathbb{R}} F_{D(Y)}(\delta)\, dD(Z)_*P(\delta). \tag{17}$$

Obviously, it holds that $D(Y), D^{(0,k)} \geq 0$. The following lemma formalizes under realistic assumptions that we obtain perfect separability of $D(X)$ and $D(X_k)$ as we keep iterating the initial CW attack, i.e., $k \to \infty$.

Lemma 2. *Let $D(X) \geq 0$ with $P(D(X) = 0) = 0$ and $D(X_k) \to 0$ for $k \to \infty$ weakly by law. Then,*

$$\mathrm{AUROC}\,(D(X_k), D(X)) \overset{k\to\infty}{\longrightarrow} 1\,. \tag{18}$$

Proof. Let δ_0 be the Dirac measure in 0 with distribution function

$$F_{\delta_0}(z) = \begin{cases} 1 & z \geq 0 \\ 0 & \text{else.} \end{cases} \tag{19}$$

By the characterization of weak convergence in law by the Helly-Bray lemma [11, Theorem 3], $F_{D(X_k)}(\delta) \to F_{\delta_0}(\delta)$ for all δ where $F_{\delta_0}(\delta)$ is continuous. This is the fact is for all $\delta > 0$. As $P(D(X) = 0) = 0$, this implies $F_{D(X_k)} \to F_{\delta_0}$ $D(X)_*P$-almost surely. Furthermore, it holds that

$|F_{D(X_k)}(\delta)| \leq 1$. Consequently, by Lebesgue's theorem of dominated convergence

$$\begin{aligned} \text{AUROC}\,(D(X_k), D(X)) &= \int_{\mathbb{R}} F_{D(X_k)}(\delta)\, dD(X)_* P(\delta) \\ &\overset{k\to\infty}{\longrightarrow} \int_{\mathbb{R}} F_{\delta_0}(\delta)\, dD(X)_* P(\delta) \\ &= \int_{\mathbb{R}_+} dD(X)_* P(\delta) = 1 \end{aligned} \tag{20}$$

as $D(X) \geq 0$ by assumption which concludes the proof. □

Le now X be a random input image. As $X \in \mathcal{NF}^{(1)}$, it holds that $D(X) > 0$ almost surely, such that indeed $P(D(X) = 0) = 0$. Let furthermore X_k be the k-th iterate inside $\mathcal{F}^{(1)}$of the CW attack starting with X. Given that the attack is successful, we obtain that $X_k \overset{k\to\infty}{\longrightarrow} X^* \in \partial\mathcal{F}$, thus

$$\text{dist}(X_k, X^*) \overset{k\to\infty}{\longrightarrow} 0 \tag{21}$$

almost surely. Now there exists a learning rate schedule $\alpha_{k,j}$ and a multiplier a such that for all steps $X_{k,j}$ of the counter attack originating at X_k, the distance

$$\bar{D}^{(k,j)} = \text{dist}(X_{k,j}, X_k) \leq 8\,\text{dist}(X_k, X^*) \overset{k\to\infty}{\longrightarrow} 0 \tag{22}$$

almost surely. Note that the latter inequality holds since we can choose ε in theorem 1, such that $\text{dist}(x_k, x^*) > \varepsilon/2$ and then obtain this constant 8 by using the triangle inequality. Therefore, let X_{k,j^*} be the first iterate of the CW counter attack in $\mathcal{F}^{(2)}$. We consider $\bar{D}^{(k,j^*)} = \text{dist}(X_k, X_{k,j^*}) \leq 8\,\text{dist}(X_k, X^*) \overset{k\to\infty}{\longrightarrow} 0$ almost surely. As this trivially implies $\bar{D}^{(k,j^*)} \overset{k\to\infty}{\longrightarrow} 0$, we obtain the following theorem by application of Lemma 2:

Theorem 3. *Under the assumptions outlined above, we obtain the perfect separation of the distribution of the distance metric $\bar{D}^{(k,j^*)}$ of the CW counter attack from the distribution on the distance metric $D^{(0,k)}$ of the original CW attack, i.e.,*

$$\text{AUROC}\left(D^{(0,k)}, \bar{D}^{(k,j^*)}\right) \overset{k\to\infty}{\longrightarrow} 1\,. \tag{23}$$

3 Numerical Experiments

We now demonstrate how our theoretical considerations apply to numerical experiments. Therefore we aim at separating the distributions $D^{(k)}$ and $\bar{D}^{(k,j)}$. For our tests we used the framework [2] (provided with [3]). For the CIFAR10 dataset consisting of tiny 32×32 RGB images with concepts from 10 classes, containing 50k training and 10k test images, we used a network with 4 convolutional layers and two dense ones as included and used per default in [2]. For the ImageNet2012 high resolution RGB images we used a pre-trained Inception network [13] (trained on all 1000 classes). For all tests we used default parameters.

Numbers of samples in $\mathcal{X}$ and $\mathcal{X}$		
	CIFAR10	ImageNet2012
ℓ_0	5000	500
ℓ_2	5000	500
ℓ_∞	1000	500

Table 1: Numbers of images in $\mathcal{X}$ and $\bar{\mathcal{X}}$, respectively.

Success rates				
	CIFAR10		ImageNet2012	
	1st attack	2nd attack	1st attack	2nd attack
ℓ_0	100%	100%	100%	100%
ℓ_2	100%	100%	100%	100%
ℓ_∞	100%	100%	100%	100%

Table 2: Success rates of first and second attacks. An attack is considered successful if the predicted class of the attack's output is different from the predicted class of the attack's input.

For each of the two datasets, we randomly split the test set into two equally sized portions $\mathcal{X}$ and $\bar{\mathcal{X}}$. For one portion we compute the values $D^{(k)}$ and for the other one also $\bar{D}^{(k,j)}$ such that they refer to two distinct sets of original images. Since CW attacks can be computationally demanding, we chose the sample sizes for $\mathcal{X}$ and $\bar{\mathcal{X}}$ as stated in table 1.

Since we discuss the distance measures $D^{(k)}$ and $\bar{D}^{(k,j)}$ for different ℓ_p norms, we now alter our notation. Let $\gamma_p : I^n \to I^n$ be the function that maps $x \in \mathcal{X}$ to its ℓ_p attacked counterpart for $p = 0, 2, \infty$. In order to demonstrate the separability of $\mathcal{X}$ and $\gamma_p(\bar{\mathcal{X}})$ under a second ℓ_q attack, we compute the two scalar sets

$$\begin{aligned} D_q =& \{ dist_q(x, \gamma_q(x)) \,:\, x \in \mathcal{X} \} \quad \text{and} \\ \bar{D}_{p,q} =& \{ dist_q(\gamma_p(\bar{x}), \gamma_q(\gamma_p(\bar{x}))) \,:\, \bar{x} \in \bar{\mathcal{X}} \} \end{aligned} \tag{24}$$

for $q = 0, 2, \infty$.

ℓ_p Counter Attacks. For now we focus on the case $p = q$ where the first and the second attack are performed by means of the same ℓ_p norm as this matches with our theoretical findings from section 2. First the state success rates of $\gamma_p(\bar{\mathcal{X}})$, i.e., the percentages where $\kappa(\bar{x}) \neq \kappa(\gamma_p(\bar{x}))$ for $\bar{x} \in \bar{\mathcal{X}}$, see table 2. For all attacks γ_p, $p = 0, 2, \infty$, we obtain ideal success rates.

The separability of original data $\mathcal{X}$ and attacked data $\gamma_p(\bar{\mathcal{X}})$ is visualized by violin plots for D_p and $\bar{D}_{p,p}$ in section 3 for CIFAR10 and in section 3 for ImageNet2012. For both datasets we observe a similar behavior. In the first three panels of each figure, i.e., for $p = 0, 2$, we observe well-separable distributions. For $p = \infty$ this separability seems rather moderate. When looking at absolute ℓ_∞ distances, we observe that both distributions peak for an ℓ_∞ distance of roughly 0.1. For ℓ_2, the absolute distances are much higher. This is clear, since the ℓ_∞ norm is independent of the image dimensions whereas the other norms suffer from the curse of dimensionality. This can be omitted by re-normalizing the shrinkage coefficient for ℓ_p distance minimization. However it also underlines the punchline of our method, the more efficient the iterative minimization process, the better our detection method – the efficiency becomes treacherous. Less efficient attacks

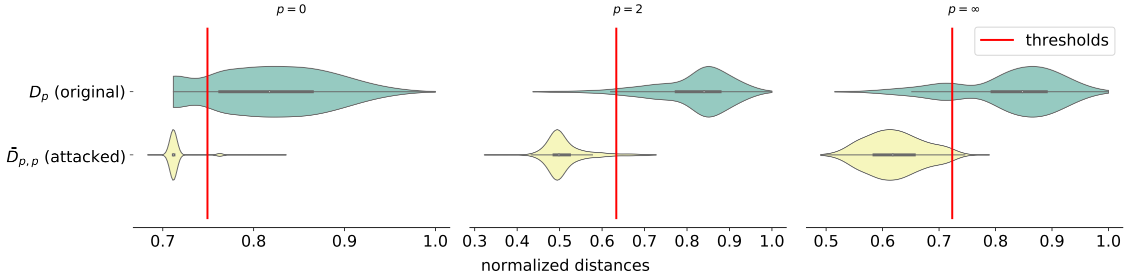

Figure 5: Violin plots displaying the distributions of distances in D_p (top) and $\bar{D}_{p,p}$ (bottom) from equation (24) for the CIFAR10 dataset, $p = 0, 2, \infty$ in ascending order from left to right.

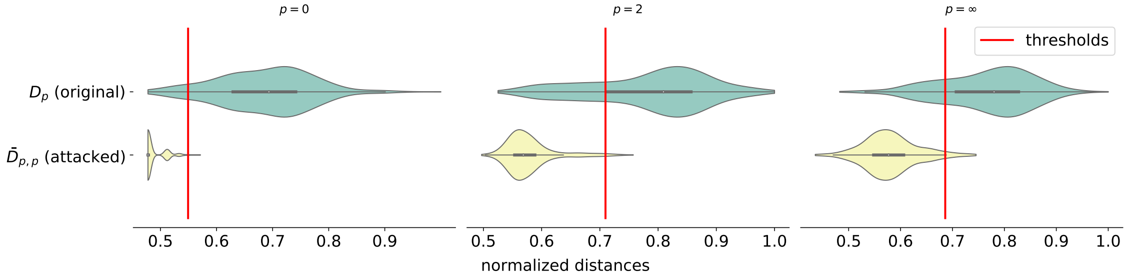

Figure 6: Violin plots displaying the distributions of distances in D_p (top) and $\bar{D}_{p,p}$ (bottom) from equation (24) for the ImageNet2012 dataset, $p = 0, 2, \infty$ in ascending order from left to right.

are not suitable detectors, however such attacks can potentially be detected by other detection methods as they tend to be more perceptible.

Next, we study the performance in terms of detection accuracy, i.e., the relative frequency of correctly classified as attacked / original for two different thresholds, as well as in terms of the area under receiver operator characteristic curve (AUROC, [9]) which is a threshold independent performance measure.

In all further discussions, true positives are correctly classified attacked images and true negatives are correctly classified original (non-attacked) images. More precisely, we fix the following notations:

- threshold: the value t used for discriminating whether $d \in D_q$ or $d \in \bar{D}_{p,q}$,
- true positive: $TP = |\{d \in \bar{D}_{p,q} \, : \, d < t\}|$,
- false negative: $FN = |\{d \in \bar{D}_{p,q} \, : \, d \geq t\}|$,
- true negative: $TN = |\{d \in D_q \, : \, d \geq t\}|$,
- false positive: $FP = |\{d \in D_q \, : \, d < t\}|$,
- accuracy: $\frac{TP+TN}{TP+FN+TN+FP}$,
- precision: $\frac{TP}{TP+FP}$,
- recall: $\frac{TP}{TP+FN}$.

For this paragraph, let $p = q$.

In table 3 we report several metrics for two choices of threshold computation. Choice no. 1 is obtained by $0.5 \times \mathit{recall} + 0.25 \times \mathit{accuracy} + 0.25 \times \mathit{precision}$, choice no. 2 is obtained by maximizing accuracy. While choice no. 1 rather aims at detecting many adversarial attacks, therefore accepting to produce additional false positive detections, choice no. 2 aims attributes the same importance to the reduction of false positives and false negatives.

In accordance to our findings in section 3 we observe high accuracy, recall and precision values. In particular the ℓ_2 attack shows very strong results and can be detected best by a repeated attack. Please recall that we reduced the number of images for our tests with the ℓ_∞ attack to 1,000 original and 1,000 attacked images as these tests were computationally intensive. Noteworthily, out of 10,000 images we only obtain 16 false positive detections, i.e., images that are falsely accused to be attacked ones, and 56 false negatives, i.e., attacked images that are not detected, when using threshold choice no. 1. For the ℓ_∞ attack we observe an over production of false positives. However, most of the 1,000 attacked images were still detected.

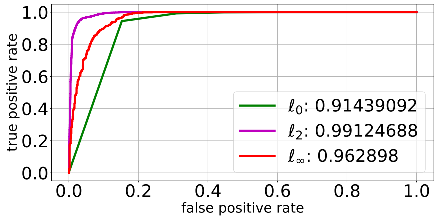

Figure 7: ROC curves and AUROC values for the different ℓ_p norms, $p = 0, 2, \infty$.

Returning to the Original Class. As motivated in section 2, the second attack theoretically is supposed to make $\gamma_p(\bar{x})$ for $\bar{x} \in \bar{\mathcal{X}}$ return across the decision boundary that it passed via the first attack, i.e.,

$$\kappa(\gamma_p(\gamma_p(\bar{x})) = \kappa(\bar{x}) \, . \tag{25}$$

We present results for the number of returnees, i.e., $R_p = |\{\bar{x} \in \bar{\mathcal{X}} \, : \, \kappa(\gamma_p(\gamma_p(\bar{x})) = \kappa(\bar{x})\}|$, and the corresponding

Sensitivity & Specificity values						
	threshold choice no. 1			threshold choice no. 2		
	ℓ_0	ℓ_2	ℓ_∞	ℓ_0	ℓ_2	ℓ_∞
True Positive	4,723	4,817	984	4,723	4,803	982
False Negative	277	183	16	277	197	18
True Negative	4,239	4,792	837	4,239	4,809	840
False Positive	761	208	163	761	191	160
Accuracy	**0.8962**	**0.9609**	**0.9105**	**0.8962**	**0.9612**	**0.9110**
Recall	**0.9446**	**0.9634**	**0.9840**	**0.9446**	**0.9606**	**0.9820**
Precision	**0.8612**	**0.9586**	**0.8579**	**0.8612**	**0.9617**	**0.8599**
Threshold	5.0126	0.00891	0.0324	5.0126	0.008225	0.0321

Table 3: Sensitivity & Specificity values for the CIFAR10. Choice no. 1 of thresholds is obtained by maximizing $0.5 \times \textit{recall} + 0.25 \times \textit{accuracy} + 0.25 \times \textit{precision}$, choice no. 2 is obtained by simply maximizing accuracy. True positives is the number of correctly classified attacked images, true negatives is the number of correctly classified original images.

percentages (return rates) given by $R_p/|\bar{\mathcal{X}}|$ in table 4. For both datasets we observe strong return rates, in particular for ℓ_2 only 3 out of 5000 samples $\bar{x} \in \bar{\mathcal{X}}$ do not fulfill $\kappa(\gamma_p(\gamma_p(\bar{x})) = \kappa(\bar{x})$.

	dataset	ℓ_0	ℓ_2	ℓ_∞
Returnees	CIFAR10	4994	4997	995
Percentage	CIFAR10	99.88%	99.94%	99.5%
Returnees	ImageNet2012	500	486	488
Percentage	ImageNet2012	100%	97.20%	97.6%

Table 4: Number of returnees as well as return rates.

Cross attacks. As the ℓ_p norms used in our tests are equivalent except for ℓ_0, we expect that cross attacks, i.e., the case $p \neq q$ is supposed to yield a good separation of D_q and $\bar{D}_{p,q}$ (cf. equation (24)). Figures 8 and 9 show results for cross attacks. Each column shows the detection performance of a norm ℓ_q.

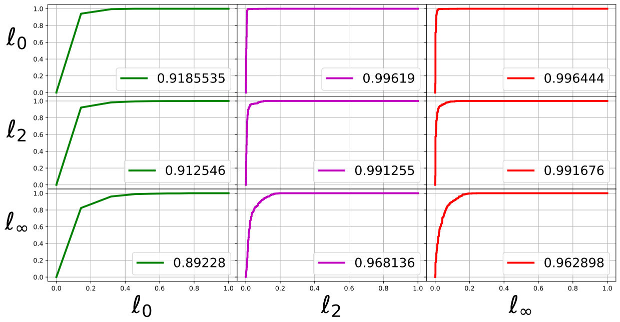

Figure 8: ROC curves and AUROC values for the different cross attacks performed on the CIFAR10 dataset. The task is to separate D_q and $\bar{D}_{p,q}$, the ℓ_q norm is the one used for detection, $q = 0, 2, \infty$ is the row index and $p = 0, 2, \infty$ the column index.

In both cases, for CIFAR10 and ImageNet2012, when comparing the different columns of both plots we observe a superiority of the ℓ_2 norm. In our tests we observe that the ℓ_2 norm also requires lowest computational effort, thus the ℓ_2 norm might be favorable from both perspectives. Noteworthily there is also only a minor performance degradation when going from CIFAR10 to ImageNet2012 even though the perturbations introduced by the ℓ_p attacks, in particular for $p = 2$, are almost imperceptible, cf. also figure 2. For a better overview, the presented results are summarized in table 5.

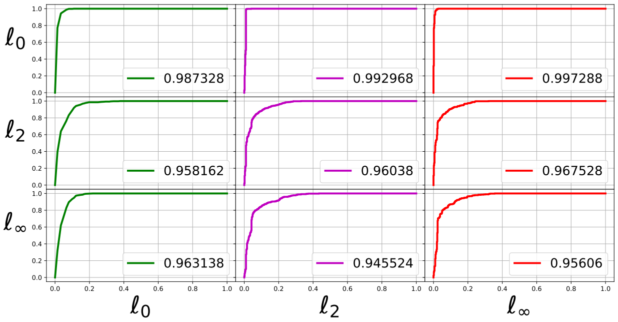

Figure 9: ROC curves and AUROC values for the different cross attacks performed on the ImageNet2012 dataset. The task is to separate D_q and $\bar{D}_{p,q}$, the ℓ_q norm is the one used for detection, $q = 0, 2, \infty$ is the row index and $p = 0, 2, \infty$ the column index.

Targeted Attacks. So far, all results presented have been computed for untargetted attacks. In principle a targeted attack γ_p only increases the distances $dist_q(\gamma_p(\bar{x}), \bar{x})$ while the distance measures $dist_q(\gamma_q(\gamma_p(\bar{x})), \gamma_q(\bar{x}))$ corresponding to another untargeted attack γ_q in principle are supposed to remain unaffected. Thus, targeted attacks should be even easier to detect. However, it might happen that we sometimes lose the property that after a second attack the predicted class returns to the one of the original image, i.e., $\kappa(\gamma_q(\gamma_p(\bar{x}))) = \kappa(\bar{x})$ does not need to hold. For instance, it might happen that the shortest straight ℓ_p path crosses another class before reaching its desired destination. In figure 10 we indeed observe that a targeted ℓ_2 attack yields a higher AUROC value than an untargeted one.

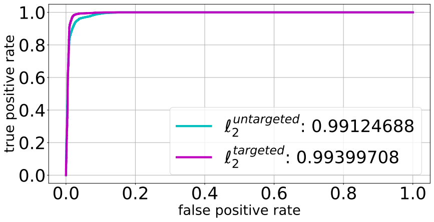

Figure 10: ROC curves and AUROC values for ℓ_2 detections on CIFAR10 data where the first attack was once targeted and once untargeted.

Metrics	CIFAR10			ImageNet2012		
L_0	91.86%	99.62%	99.64%	98.73	99.30	99.73
L_2	91.25%	99.13%	99.17%	95.82	96.04	96.75
L_∞	89.23%	96.81%	96.29%	96.31	94.55	95.61
Metrics	L_0	L_2	L_∞	L_0	L_2	L_∞

Table 5: Summarizing table for crossed-attacked. First column represents the first attack used to craft data and last row the one performed for detection. We remind that CIFAR10 data sets shuffle 1000 regular images and 1000 adversarial images while ImageNet2012 data sets hold 500 clean images and 500 perturbed images.

4 Conclusion and Outlook

We outlined a mathematical proof for asymptotically optimal detection of CW attacks via counter attacks for the ℓ_2 norm and demonstrated in numerical experiments that our findings hold to a high extent in practice for different ℓ_p norms. We obtained AUROC values of up to 99.73% on the CIFAR10 dataset and demonstrated that also cross attacks based on different norms ℓ_p and ℓ_q yield high detection performance.

Our results are in range of state-of-the-art results. DBA [48] shows a superior detection accuracy for CIFAR10, however the underlying neural network is also much stronger which makes a clear comparison difficult. AUROC values are not reported. Our results for the ℓ_2-attack with CIFAR10 outperform those of defense-GAN [33] for fashion MNIST by far (which is also difficult to compare). The same statement holds for the CIFAR10 results presented for I-defender [47] when detecting an iterative ℓ_2 attack. I-defender was successful for roughly 90% of the test data.

We expect that our results can be further improved by proceeding similarly to [14] and investigating statistics obtained from sets of attacked and original images instead of single samples (the latter is how we proceed so far). All networks used in our tests have not been hardened / trained for robustness. We anticipate that our method might benefit from using hardened networks as the ℓ_p-distance of the first attack might even increase. Furthermore, works that study the nature of adversarial attacks found that most attacks are located close to decision boundaries, see e.g. [43]. In accordance with the findings in [48] this suggests, that our counter attack framework might transfer also to other attacks than the CW attack. Studies on these aspects as well as an extension of our mathematical analysis remain future work.

Acknowledgement. H. G., M. P. and M. R. acknowledge interesting discussions with T. Hantschmann.

References

[1] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok. Synthesizing robust adversarial examples. *CoRR*, abs/1707.07397, 2017.

[2] N. Carlini. Robust evasion attacks against neural network to find adversarial examples. https://github.com/carlini/nn_robust_attacks.

[3] N. Carlini and D. A. Wagner. Towards evaluating the robustness of neural networks. *CoRR*, abs/1608.04644, 2016.

[4] F. Carrara, R. Becarelli, R. Caldelli, F. Falchi, and G. Amato. Adversarial examples detection in features distance spaces. In *The European Conference on Computer Vision (ECCV) Workshops*, September 2018.

[5] P. Chen, Y. Sharma, H. Zhang, J. Yi, and C.-J. Hsieh. Ead: Elastic-net attacks to deep neural networks via adversarial examples. *ArXiv*, abs/1709.04114, 2017.

[6] S. Chen, N. Carlini, and D. A. Wagner. Stateful detection of black-box adversarial attacks. *CoRR*, abs/1907.05587, 2019.

[7] F. Clarke. *Optimization and Nonsmooth Analysis.* SIAM, 1990.

[8] F. Croce, M. Andriushchenko, and M. Hein. Provable robustness of relu networks via maximization of linear regions. *CoRR*, abs/1810.07481, 2018.

[9] J. Davis and M. Goadrich. The relationship between precision-recall and ROC curves. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, pages 233–240, 2006.

[10] R. Feinman, R. R. Curtin, S. Shintre, and A. B. Gardner. Detecting adversarial samples from artifacts. *ArXiv*, abs/1703.00410, 2017.

[11] T. Ferguson. *A Course in Large Sample Theory.* Springer US, 1996.

[12] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *CoRR*, abs/1412.6572, 2014.

[13] Google. Tensorflow inception network. http://download.tensorflow.org/models/image/imagenet/inception-2015-12-05.tgz.

[14] K. Grosse, P. Manoharan, N. Papernot, M. Backes, and P. D. McDaniel. On the (statistical) detection of adversarial examples. *CoRR*, abs/1702.06280, 2017.

[15] C. Guo, M. Rana, M. Cissé, and L. van der Maaten. Countering adversarial images using input transformations. *CoRR*, abs/1711.00117, 2017.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[17] S. Jha, U. Jang, S. Jha, and B. Jalaian. Detecting adversarial examples using data manifolds. In *MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM)*, pages 547–552, Oct 2018.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[19] A. Kurakin, I. J. Goodfellow, and S. Bengio. Adversarial machine learning at scale. *CoRR*, abs/1611.01236, 2016.

[20] S. Lee, S. Park, and J. Lee. Defensive denoising methods against adversarial attack. In *24th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2018.

[21] B. Liang, H. Li, M. Su, X. Li, W. Shi, and X. Wang. Detecting adversarial examples in deep networks with adaptive noise reduction. *CoRR*, abs/1705.08378, 2017.

[22] J. Liu, W. Zhang, Y. Zhang, D. Hou, Y. Liu, and N. Yu. Detecting adversarial examples based on steganalysis. *CoRR*, abs/1806.09186, 2018.

[23] Z. Liu, Q. Liu, T. Liu, Y. Wang, and W. Wen. Feature distillation: Dnn-oriented JPEG compression against adversarial examples. *CoRR*, abs/1803.05787, 2018.

[24] J. Lu, T. Issaranon, and D. A. Forsyth. Safetynet: Detecting and rejecting adversarial examples robustly. *CoRR*, abs/1704.00103, 2017.

[25] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *ArXiv*, abs/1706.06083, 2017.

[26] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff. On detecting adversarial perturbations. *ArXiv*, abs/1702.04267, 2017.

[27] S. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard. Universal adversarial perturbations. *CoRR*, abs/1610.08401, 2016.

[28] S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: a simple and accurate method to fool deep neural networks. *CoRR*, abs/1511.04599, 2015.

[29] N. Papernot, P. D. McDaniel, X. Wu, S. Jha, and A. Swami. Distillation as a defense to adversarial perturbations against deep neural networks. *CoRR*, abs/1511.04508, 2015.

[30] A. Prakash, N. Moran, S. Garber, A. DiLillo, and J. A. Storer. Protecting JPEG images against adversarial attacks. *CoRR*, abs/1803.00940, 2018.

[31] K. Ren, T. Zheng, Z. Qin, and X. Liu. Adversarial attacks and defenses in deep learning. *Engineering*, 6(3):346 – 360, 2020.

[32] K. Roth, Y. Kilcher, and T. Hofmann. The odds are odd: A statistical test for detecting adversarial examples. *CoRR*, abs/1902.04818, 2019.

[33] P. Samangouei, M. Kabkab, and R. Chellappa. Defensegan: Protecting classifiers against adversarial attacks using generative models. *CoRR*, abs/1805.06605, 2018.

[34] S. Scholtes. *Introduction to Piecewise Differentiable Equations*. SpringerBriefs in Optimization. Springer New York, 2012.

[35] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014.

[36] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[37] M. V. Sodolov and S. K. Zavriev. Error stability properties of generalized gradient-type algorithms. *JOTA*, 1998.

[38] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *CoRR*, abs/1312.6199, 2013.

[39] O. Taran, S. Rezaeifar, T. Holotyak, and S. Voloshynovskiy. Defending against adversarial attacks by randomized diversification. *CoRR*, abs/1904.00689, 2019.

[40] S. Tian, G. Yang, and Y. Cai. Detecting adversarial examples through image transformation. *AAAI Conference on Artificial Intelligence*, 2018.

[41] Y. Wang, X. Ma, J. Bailey, J. Yi, B. Zhou, and Q. Gu. On the convergence and robustness of adversarial training. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6586–6595, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

[42] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. L. Yuille. Mitigating adversarial effects through randomization. *CoRR*, abs/1711.01991, 2017.

[43] H. Xu, Y. Ma, H. Liu, D. Deb, H. Liu, J. Tang, and A. K. Jain. Adversarial attacks and defenses in images, graphs and text: A review. *Int. J. Autom. Comput.*, 17(2):151–178, 2020.

[44] D. A. Yap, J. Xu, and V. U. Prabhu. On detecting adversarial inputs with entropy of saliency maps. In *CV-COPS, IEEE CVPR*, 2019.

[45] X. Yin, S. Kolouri, and G. K. Rohde. Divide-and-conquer adversarial detection. *CoRR*, abs/1905.11475, 2019.

[46] C. Zhao, P. T. Fletcher, M. Yu, Y. Peng, G. Zhang, and C. Shen. The adversarial attack and detection under the fisher information metric. *CoRR*, abs/1810.03806, 2018.

[47] Z. Zheng and P. Hong. Robust detection of adversarial attacks by modeling the intrinsic properties of deep neural networks. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 7924–7933, USA, 2018. Curran Associates Inc.

[48] Q. Zhou, R. Zhang, B. Wu, W. Li, and T. Mo. Detection by attack: Detecting adversarial samples by undercover attack. In L. Chen, N. Li, K. Liang, and S. Schneider, editors, *Computer Security – ESORICS 2020*, pages 146–164, Cham, 2020. Springer International Publishing.