Towards Deep Learning Models Resistant to Adversarial Attacks

Aleksander Madry [1] **Aleksandar Makelov** [1] **Ludwig Schmidt** [1] **Dimitris Tsipras** [1] **Adrian Vladu** [1] *

Abstract

Recent work has demonstrated that neural networks are vulnerable to adversarial examples, i.e., inputs that are almost indistinguishable from natural data and yet classified incorrectly by the network. To address this problem, we study the adversarial robustness of neural networks through the lens of robust optimization. This approach provides a broad and unifying view on much of the prior work on this topic. Its principled nature also enables us to identify general methods for both training and attacking neural networks that are reliable and, in a certain sense, universal. These methods let us train networks with significantly improved resistance to a wide range of adversarial attacks. This suggests that adversarially resistant deep learning models might be within our reach after all.

1. Introduction

Recent breakthroughs in computer vision and speech recognition are bringing trained classifiers into the center of security-critical systems. Important examples include vision for autonomous cars, face recognition, and malware detection. These developments make security aspects of machine learning increasingly important. In particular, resistance to *adversarially chosen inputs* is becoming a crucial design goal. While trained models tend to be very effective in classifying benign inputs, recent work (Szegedy et al., 2013; Goodfellow et al., 2014; Nguyen et al., 2015; Sharif et al., 2016) shows that an adversary is often able to manipulate the input so that the model produces an incorrect output.

This phenomenon has received particular attention in the context of deep neural networks, and there is now a quickly growing body of work on this topic (Fawzi et al., 2015; Kurakin et al., 2016; Papernot & McDaniel, 2016; Rozsa et al., 2016; Torkamani, 2016; Sokolic et al., 2016; Tramèr et al., 2017b). Computer vision presents a particularly striking challenge: very small changes to the input image can fool state-of-the-art neural networks with high probability (Szegedy et al., 2013; Goodfellow et al., 2014; Nguyen et al., 2015; Sharif et al., 2016; Moosavi-Dezfooli et al., 2016). This holds even when the benign example was classified correctly, and the change is imperceptible to a human. Apart from the security implications, this phenomenon also demonstrates that our current models are not learning the underlying concepts in a robust manner. All these findings raise a fundamental question:

How can we learn models robust to adversarial inputs?

While existing attacks and defense mechanisms have had some successes, we currently do not have a good understanding of the *guarantees* they provide. We can never be certain that a given attack finds the "most adversarial" example, or that a particular defense mechanism prevents the existence of *all* adversarial examples. This makes it difficult to navigate the landscape of adversarial attacks or to fully evaluate the possible security implications.

In this paper, we study the adversarial robustness of neural networks through the lens of robust optimization. We use a natural saddle point (min-max) formulation to capture the notion of security against adversarial attacks in a principled manner. This formulation allows us to be precise about the type of security *guarantee* we would like to achieve, i.e., the broad *class* of attacks we want to be resistant to (in contrast to defending only against specific known attacks). The formulation also enables us to cast both *attacks* and *defenses* into a common theoretical framework. Most prior work on adversarial examples naturally fits into this framework. In particular, adversarial training directly corresponds to optimizing this saddle point problem. Similarly, prior methods for attacking neural networks correspond to specific algorithms for solving the underlying constrained optimization problem.

Equipped with this perspective, we make the following contributions.

1. We conduct a careful experimental study of the optimization landscape corresponding to this saddle point

*listed in alphabetical order [1]Massachusetts Institute of Technology, Cambridge, Massachusetts, USA. Correspondence to: Aleksander Madry <madry@mit.edu>.

Presented at the ICML 2017 Workshop on Principled Approaches to Deep Learning, Sydney, Australia, 2017.

formulation. Despite the non-convexity and non-concavity of its constituent parts, we find that the underlying optimization problem *is* tractable after all. In particular, we provide strong evidence that first-order methods can reliably solve this problem. We supplement these insights with ideas from real analysis to further motivate projected gradient descent (PGD) as a universal "first-order adversary", i.e., the strongest attack utilizing the local first order information about the network.

2. We explore the impact of network architecture on adversarial robustness and find that model capacity plays an important role here. To reliably withstand strong adversarial attacks, networks require a significantly larger capacity than for correctly classifying benign examples only. This shows that a robust decision boundary of the saddle point problem can be significantly more complicated than a decision boundary that simply separates the benign data points.

3. Building on the above insights, we train networks on MNIST and CIFAR10 that are robust to a wide range of adversarial attacks. Our approach is based on optimizing the aforementioned saddle point formulation and uses our optimal "first-order adversary". Our best MNIST model achieves an accuracy of more than 89% against the strongest adversaries in our test suite. In particular, our MNIST network is even robust against *white box* attacks of an *iterative* adversary. Our CIFAR10 model achieves an accuracy of 46% against the same adversary. Furthermore, in case of the weaker *black box/transfer* attacks, our MNIST and CIFAR10 networks achieve the accuracy of more than 95% and 64%, respectively. (More detailed overview can be found in Tables 1 and2.) To the best of our knowledge, we are the first to achieve these levels of robustness on MNIST and CIFAR10 against such a broad set of attacks.

Overall, these findings suggest that secure neural networks are within reach. In order to reliably test this claim, we invite the community to attempt attacks against our MNIST and CIFAR10 networks in the form of a challenge. The complete code, along with the description of the challenge, is available at https://github.com/MadryLab/mnist_challenge and https://github.com/MadryLab/cifar10_challenge.

2. An Optimization View on Adversarial Robustness

Much of our discussion will revolve around an optimization view of adversarial robustness. This perspective not only captures the phenomena we want to study in a precise manner, but will also inform our investigations. To this end, let us consider a standard classification task with an underlying data distribution $\mathcal{D}$ over pairs of examples $x \in \mathbb{R}^d$ and corresponding labels $y \in [k]$. We also assume that we are given a suitable loss function $J(\theta, x, y)$, for instance the cross-entropy loss for a neural network. As usual, $\theta \in \mathbb{R}^p$ is the set of model parameters. Our goal then is to find model parameters θ that minimize the risk $\mathbb{E}_{(x,y)\sim\mathcal{D}}[J(x, y, \theta)]$.

Empirical risk minimization (ERM) has been tremendously successful as a recipe for finding classifiers with small population risk. Unfortunately, ERM often does not yield models that are robust to adversarially crafted examples (Goodfellow et al., 2014; Kurakin et al., 2016; Moosavi-Dezfooli et al., 2016; Tramèr et al., 2017b). Formally, there are efficient algorithms ("adversaries") that take an example x belonging to class c_1 as input and find examples x^{adv} such that x^{adv} is very close to x but the model incorrectly classifies x^{adv} as belonging to class $c_2 \neq c_1$.

In order to *reliably* train models that are robust to adversarial attacks, it is necessary to augment the ERM paradigm appropriately. Instead of resorting to methods that directly focus on improving the robustness to specific attacks, our approach is to first propose a concrete *guarantee* that an adversarially robust model should satisfy. We then adapt our training methods towards achieving this guarantee.

The first step towards such a guarantee is to specify an *attack model*, i.e., a precise definition of the attacks our models should be resistant to. For each data point x, we introduce a set of allowed perturbations $\mathcal{S} \subseteq \mathbb{R}^d$ that formalizes the manipulative power of the adversary. In image classification, we choose $\mathcal{S}$ so that it captures perceptual similarity between images. For instance, the ℓ_∞-ball around x has recently been studied as a natural notion for adversarial perturbations (Goodfellow et al., 2014). While we focus on robustness against ℓ_∞-bounded attacks in this paper, we remark that more comprehensive notions of perceptual similarity are an important direction for future research.

Next, we modify the definition of population risk $\mathbb{E}_{\mathcal{D}}[J]$ by incorporating the above adversary. Instead of feeding samples from the distribution $\mathcal{D}$ directly into the loss J, we allow the adversary to perturb the input first. This gives rise to the following saddle point problem, which is our central object of study:

$$\min_{\theta} \rho(\theta) \tag{1}$$

$$\text{where } \rho(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta\in\mathcal{S}} J(\theta, x+\delta, y) \right] .$$

Formulations of this type (and their finite-sample counterparts) have a long history in robust optimization, going

back to Wald (Wald, 1939; 1945; 1992). It turns out that this formulation is also particularly useful in our context.

First, this formulation gives us a unifying perspective that encompasses much prior work on adversarial robustness. Our perspective stems from viewing the saddle point problem as the composition of an *inner maximization* problem and an *outer minimization* problem. Both of these problems have a natural interpretation in our context. The inner maximization problem aims to find an adversarial version of a given data point x that achieves a high loss. This is precisely the problem of attacking a given neural network. On the other hand, the goal of the outer minimization problem is to find model parameters so that the "adversarial loss" given by the inner attack problem is minimized. This is precisely the problem of training a robust classifier using adversarial training techniques.

Second, the saddle point problem specifies a clear goal that an ideal robust classifier should achieve, as well as a quantitative measure of its robustness. In particular, when the parameters θ yield a (nearly) vanishing risk, the corresponding model is perfectly robust to attacks specified by our attack model.

Our paper investigates the structure of this saddle point problem in the context of deep neural networks. These investigations then lead us to training techniques that produce models with high resistance to a wide range of adversarial attacks. Before turning to our contributions, we briefly review prior work on adversarial examples and describe in more detail how it fits into the above formulation.

2.1. A Unified View on Attacks and Defenses

Prior work on adversarial examples has focused on two main questions:

1. How can we produce strong adversarial examples, i.e., adversarial examples that fool a model with high confidence while requiring only a small perturbation?
2. How can we train a model so that there are no adversarial examples, or at least so that an adversary cannot find them easily?

Our perspective on the saddle point problem (1) gives answers to both these questions. On the attack side, prior work has proposed methods such as the Fast Gradient Sign Method (FGSM) and multiple variations of it (Goodfellow et al., 2014). FGSM is an attack for an ℓ_∞-bounded adversary and computes an adversarial example as

$$x + \varepsilon \operatorname{sgn}(\nabla_x J(\theta, x, y)).$$

One can interpret this attack as a simple one-step scheme for maximizing the inner part of the saddle point formulation. A more powerful adversary is the multi-step variant FGSM^k, which is essentially projected gradient descent (PGD) (Kurakin et al., 2016):

$$x^{t+1} = \Pi_{x+\mathcal{S}} \left(x^t + \alpha \operatorname{sgn}(\nabla_x J(\theta, x, y)) \right).$$

Other methods like FGSM with random perturbation have also been proposed (Tramèr et al., 2017a). Clearly, all of these approaches can be viewed as specific attempts to solve the inner maximization problem in (1).

On the defense side, the training dataset is often augmented with adversarial examples produced by FGSM. This approach also directly follows from (1) when linearizing the inner maximization problem. To solve the simplified robust optimization problem, we replace every training example with its FGSM-perturbed counterpart. More sophisticated defense mechanisms such as training against multiple adversaries can be seen as better, more exhaustive approximations of the inner maximization problem.

3. Towards Universally Robust Networks?

Current work on adversarial examples usually focuses on specific defensive mechanisms, or on attacks against such defenses. An important feature of formulation (1) is that attaining small adversarial loss gives a *guarantee* that no allowed attack will fool the network. By definition, no adversarial perturbations are possible because the loss is small for *all* perturbations allowed by our attack model. Hence, we now focus our attention on obtaining a good solution to (1).

Unfortunately, while the overall guarantee provided by the saddle point problem is evidently useful, it is not clear whether we can actually find a good solution in reasonable time. Solving the saddle point problem (1) involves tackling both a non-convex outer minimization problem *and* a non-concave inner maximization problem. One of our key contributions is demonstrating that, in practice, one *can* solve the saddle point problem after all. In particular, we now discuss an experimental exploration of the structure given by the non-concave inner problem. We argue that the loss landscape corresponding to this problem has a surprisingly tractable structure of local maxima. This structure also points towards projected gradient descent as the ultimate first order adversary. Sections 4 and 5 then show that the resulting trained networks are indeed robust against a wide range of attacks, provided the networks are sufficiently large.

3.1. The Landscape of Adversarial Examples

Recall that the inner problem corresponds to finding an adversarial example for a given network and data point (subject to our attack model). As this problem requires us to

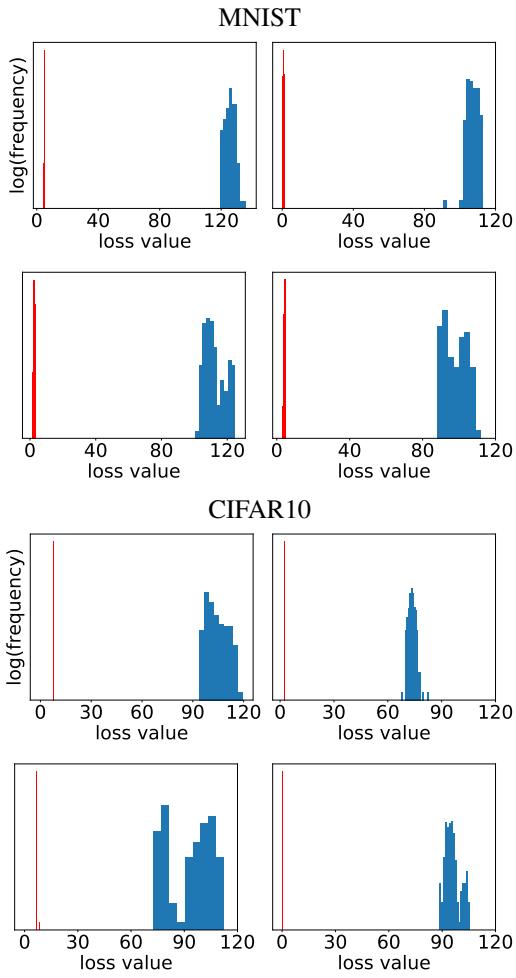


Figure 1. Histograms for the distribution of local maxima values over 200 random restarts for 4 random examples of MNIST and CIFAR10. The blue histogram corresponds to naturally training the network, while the red one corresponds to the adversarially trained version.

maximize a highly non-concave function, one would expect it to be intractable. Indeed, this is the conclusion reached by prior work which then resorted to linearizing the inner maximization problem (Huang et al., 2015a; Shaham et al., 2015). As pointed out above, this linearization approach yields well-known methods such as FGSM. While training against FGSM adversaries has shown some successes, recent work also highlights important shortcomings of this one-step approach (Tramèr et al., 2017a).

To understand the inner problem in more detail, we investigate the landscape of local maxima for multiple models on MNIST and CIFAR10. The main tool in our experiments is projected gradient descent (PGD), since it is the standard method for large-scale constrained optimization. In order to explore a large part of the loss landscape, we re-start

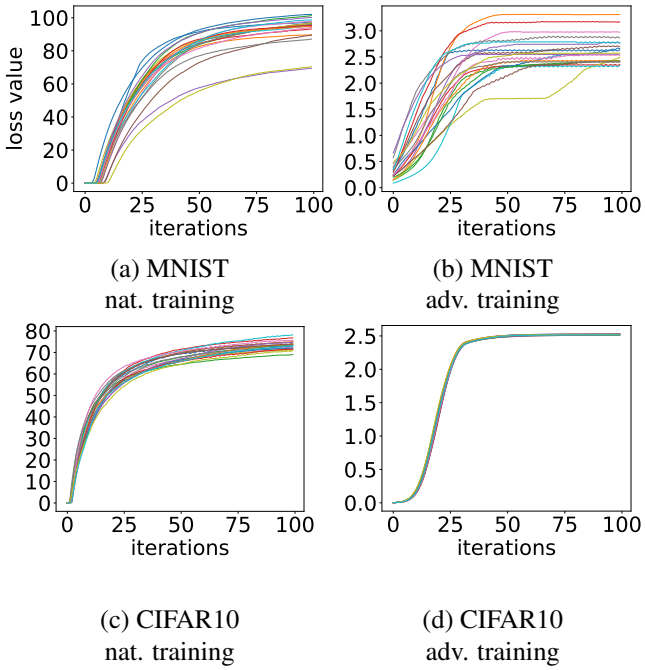


Figure 2. Value of loss function over PGD iterations for 20 random restarts using a random example.

PGD from many points in the ℓ_∞ balls around data points from the respective evaluation sets.

Surprisingly, our experiments show that the inner problem *is* tractable after all, at least from the perspective of first-order methods. While there are many local maxima spread widely apart within $x_i + \mathcal{S}$, they tend to have very *well-concentrated* loss *values*. This echoes the folklore belief that training neural networks is possible because the loss (as a function of model parameters) typically has many local minima with very similar values.

Specifically, in our experiments we found the following phenomena:

- We observe that the loss achieved by the adversary increases in a fairly consistent way and plateaus rapidly when performing projected ℓ_∞ gradient descent for randomly chosen starting points inside $x+\mathcal{S}$ (see Figure 2).
- Investigating the concentration of maxima further, we observe that over a large number of random restarts, the loss of the final iterate follows a well-concentrated distribution without extreme outliers (see Figure 1; we verified this concentration based on around 10^5 restarts).
- To demonstrate that maxima are noticeably distinct, we also measured the ℓ_2 distance and angles between all pairs of them and observed distances are distributed close to the expected distance between two random

points in the ℓ_∞ ball, and angles are close to 90°.

- Finally, we observe that the distribution of maxima suggests that the recently developed subspace view of adversarial examples is not fully capturing the richness of attacks (Tramèr et al., 2017b). In particular, we observe adversarial perturbations with negative inner product with the gradient of the example, and deteriorating overall correlation with the gradient direction as the scale of perturbation increases.

All of this evidence points towards PGD being a universal adversary among first-order approaches, as we will see next.

3.2. First-Order Bounded Adversaries

Our experiments show that the local maxima found by PGD all have similar loss values, both for normally trained networks and adversarially trained networks. This concentration phenomenon suggests an intriguing view on the problem in which robustness against the PGD adversary yields robustness against *all* first-order adversaries, i.e., attacks that rely only on first-order information. As long as the adversary only uses gradients of the loss function with respect to the input, we conjecture that it will not find significantly better local maxima than PGD. We give more experimental evidence for this hypothesis in Section 5: if we train a network to be robust against PGD adversaries, it becomes robust against a wide range of other attacks as well.

Of course, our exploration with PGD does not preclude the existence of some isolated maxima with much larger function value. However, our experiments suggest that such better local maxima are *hard to find* with first order methods: even a large number of random restarts did not find function values with significantly different loss values. Incorporating the computational power of the adversary into the attack model should be reminiscent of the notion of *polynomially bounded* adversary that is a cornerstone of modern cryptography. There, this classic attack model allows the adversary to only solve problems that require at most polynomial computation time. Here, we employ an *optimization-based* view on the power of the adversary as it is more suitable in the context of machine learning. After all, we have not yet developed a thorough understanding of the computational complexity of many recent machine learning problems. However, the vast majority of optimization problems in ML is solved with first-order methods, and variants of SGD are the most effective way of training deep learning models in particular. Hence we believe that the class of attacks relying on first-order information is, in some sense, universal for the current practice of deep learning.

Put together, these two ideas chart the way towards machine learning models with *guaranteed* robustness. If we train the network to be robust against PGD adversaries, it will be robust against a wide range of attacks that encompasses all current approaches.

In fact, this robustness guarantee would become even stronger in the context of *transfer attacks*, i.e., attacks in which the adversary does not have a direct access to the target network. Instead, the adversary only has less specific information such as the (rough) model architecture and the training data set. One can view this attack model as an example of "zero order" attacks, i.e., attacks in which the adversary has no direct access to the classifier and is only able to evaluate it on chosen examples without gradient feedback.

3.3. Descent Directions for Adversarial Training

The preceding discussion suggests that the inner optimization problem can be successfully solved by applying PGD. In order to train adversarially robust networks, we also need to solve the *outer* optimization problem of the saddle point formulation (1), that is find model parameters that minimize the "adversarial loss", the value of the inner maximization problem.

In the context of training neural networks, the main method for minimizing the loss function is Stochastic Gradient Descent (SGD). A natural way of computing the gradient of the outer problem, $\nabla_\theta \rho(\theta)$, is computing the gradient of the loss function at a maximizer of the inner problem. This corresponds to replacing the input points by their corresponding adversarial perturbations and normally training the network on the perturbed input. A priori, it is not clear that this is a valid descent direction for the saddle point problem. However, for the case of continuously differentiable functions, Danskin's theorem –a classic theorem in optimization– states this is indeed true and gradients at inner maximizers corresponds to descent directions for the saddle point problem.

Despite the fact that the exact assumptions of Danskin's theorem do not hold for our problem (the function is not continuously differentiable due to ReLU and max-pooling units, and we are only computing approximate maximizers of the inner problem), our experiments suggest that we can still use these gradients to optimize our problem. By applying SGD using the gradient of the loss at adversarial examples we can consistently reduce the loss of the saddle point problem during training, as can be seen in Figure 4. These observations suggest that we reliably optimize the saddle point formulation (1) and thus train robust classifiers.

4. Network Capacity and Adversarial Robustness

Solving the problem from Equation (1) successfully is not sufficient to guarantee robust and accurate classification. We need to also argue that the *value* of the problem (i.e. the final loss we achieve against adversarial examples) is small, thus providing guarantees for the performance of our classifier. In particular, achieving a value of zero corresponds to a perfect classifier, which is robust to adversarial inputs.

For a fixed set $\mathcal{S}$ of possible perturbations, the value of the problem is entirely dependent on the architecture of the classifier we are learning. Consequently, the architectural capacity of the model becomes a major factor affecting its overall performance. At a high level, classifying examples in a robust way requires a stronger classifier, since the presence of adversarial examples changes the decision boundary of the problem to a more complicated one.

Our experiments verify that capacity is crucial for robustness, as well as for the ability to successfully train against strong adversaries. For the MNIST dataset, we consider a simple convolutional network and study how its behavior changes against different adversaries as we keep doubling the number of convolutional filters and the size of the fully connected layer. The initial network has a convolutional layer with 2 filters, followed by another convolutional layer with 4 filters, and a fully connected hidden layer with 8 units. Convolutional layers are followed by 2×2 max-pooling layers and adversarial examples are constructed with $\varepsilon = 0.1$. The results are in Figure 3.

For the CIFAR10 dataset, we used the Resnet model (He et al., 2016; TFM). We performed data augmentation using random crops and flips, as well as per image standarization. To increase the capacity, we modified the network incorporating wider layers by a factor of 10. This results in a network with 5 residual units with (16, 160, 320, 640) filters each. This network can achieve an accuracy of 95.2% when trained with natural examples. Adversarial examples were constructed with $\varepsilon = 8$. Capacity results for $\varepsilon = 8$ are in Figure 3. We observe the following phenomena:

Capacity alone helps. We observe that increasing the capacity of the network when training using only natural examples (apart from increasing accuracy on these examples) increases the robustness against one-step perturbations significantly. Moreover we notice that training the model for a long time (much longer than required to achieve good accuracy on the eval set) significantly increases its performance on adversarial inputs.

Even weak adversaries benefit robustness. When training the network using adversarial examples generated with the FGSM, we observe robustness of the network against one-step attacks, and we additionally observe an increase in the robustness against iterative methods of attack when ε is small (we didn't observe this for $\varepsilon = 8$ on CIFAR10). This agrees with our intuition that one-step methods can find approximate solution to the inner maximization problem in the regime when the loss behaves linearly.

Weak models may fail to converge. In the case of small capacity networks, attempting to train against strong adversaries (PGD) fails to reach convergence and results in networks of poor performance, even when the network successfully converges with natural training. The small capacity of the network forces the optimization to sacrifice performance on natural examples in an attempt to fit the adversarial inputs. This behavior has a profound effect on cases where label leaking is observed, i.e. the network can overfit to the examples produced by a (weak) adversary and perform well on them while performing worse on the original images. Label leaking behavior is therefore suggestive of insufficient capacity.

The value of the saddle point problem decreases as we increase the capacity. Fixing an adversary model, and training against it, the value of (1) drops as capacity increases, indicating the the model can fit the adversarial examples increasingly well.

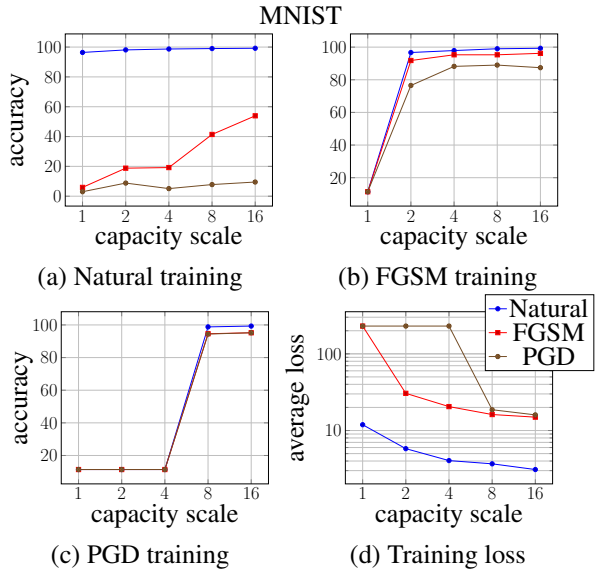
More capacity and stronger adversaries decrease transferability. Either increasing the capacity of the network, or using a stronger method for the inner optimization problem reduces the effectiveness of transferred adversarial inputs. We validate this experimentally by observing that the correlation between gradients from the source and the transfer network, becomes less significant as capacity increases. We describe our experiments in the full version of the paper.

5. Experiments: Adversarially Robust Deep Learning Models?

Following the understanding of the problem we developed in previous sections, we can now apply our proposed approach to train robust classifiers. As our experiments so far demonstrated, we need to focus on two key elements: a) train a sufficiently high capacity network, b) use the strongest possible adversary.

For both MNIST and CIFAR10, the adversary of choice will be projected gradient ascent starting from a random perturbation around the natural example. This corresponds to our notion of a "complete" first-order adversary , an algorithm that can efficiently maximize the loss of an example using only first order information.

When training against the adversary, we observe a steady



(a) Natural training (b) FGSM training

(c) PGD training (d) Training loss

CIFAR10

	simple	wide
Natural	92.7%	95.2%
FGSM	27.5%	32.7%
PGD	0.8%	3.5%

(a) Natural training

	simple	wide
Natural	87.4%	90.3%
FGSM	90.9%	95.1%
PGD	0.0%	0.0%

(b) FGSM training

	simple	wide
Natural	79.4%	87.3%
FGSM	51.7%	56.1%
PGD	43.7%	45.8%

(c) PGD training

	simple	wide
Natural	0.00357	0.00371
FGSM	0.0115	0.00557
PGD	1.11	0.0218

(d) Training Loss

Figure 3. The effect of network capacity on the performance of the network. The first 3 plots/tables of each data set show the accuracy against different adversaries depending on the adversary used during training. The last plot/table shows the final training loss (on adversarial examples) as a function of capacity (the loss is computed against the adversary used during training).

decrease in the training loss of adversarial examples, illustrated in Figure 4. This behavior indicates that we are indeed successfully solving our original optimization problem during training.

We evaluate the trained models against a range of adversaries. We illustrate our results in Table 1 for MNIST and Table 2 for CIFAR10. The adversaries we consider are:

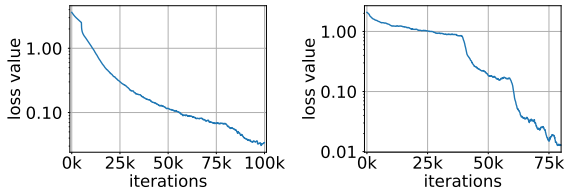
- White-box attacks with PGD for a different number of of iterations and restarts, denoted by source A.
- White-box attacks from (Carlini & Wagner, 2016b). We use their suggested loss function and minimize it using PGD. This is denoted as CW, where the corresponding attack with a high confidence parameter ($\kappa = 50$) is denoted as CW+.
- Black-box attacks from an independently trained copy of the network, denoted A'.
- Black-box attacks from a version of the same network trained only on natural examples, denoted A_{nat}.
- Black-box attacks from a different convolution architecture, denoted B, described in (Tramèr et al., 2017a).

MNIST. Guided by our observations on Figure 2, we run 40 iterations of projected gradient ascent as our adversary, with a step size of 0.01 (we choose to take gradient steps in the ℓ_∞ norm, i.e. adding the sign of the gradient, since this makes the choice of the step size simpler). We train and evaluate against perturbations of size $\varepsilon = 0.3$. We use a network consisting of two convolutional layers with 32 and 64 filters respectively, each followed by 2×2 max-pooling, and a fully connected layer of size 1024. When trained with natural examples, this network reaches 99.2% accuracy on the evaluation set. However, when evaluating on examples perturbed with FGSM the accuracy drops to 6.4%.

method	#steps	restarts	source	accuracy
natural	-	-	-	98.8%
FGSM	-	-	A	95.6%
PGD	40	1	A	93.2%
PGD	100	1	A	91.8%
PGD	40	20	A	90.4%
PGD	100	20	A	**89.3%**
targeted	40	1	A	92.7%
CW	40	1	A	94.0%
CW+	40	1	A	93.9%
FGSM	-	-	A'	96.8%
PGD	40	1	A'	96.0%
PGD	100	20	A'	**95.7%**
CW	40	1	A'	97.0%
CW+	40	1	A'	96.4%
FGSM	-	-	B	**95.4%**
PGD	40	1	B	96.4%
CW	-	-	B	95.7%

Table 1. MNIST: Performance of the adversarially trained network against different adversaries for $\varepsilon = 0.3$. For each model of attack (white-box, black-box, black-box from different architecture) we show the most successful attack with bold.

CIFAR10. For the CIFAR10 dataset, we use the two architectures described in 4 (the original Resnet and its $10x$ wider variant). We trained the network against a PGD adversary with ℓ_∞ projected gradient descent again, this time using 7 steps of size 2, and a total $\varepsilon = 8$. For our hardest adversary we chose 20 steps with the same settings, since other hyperparameter choices didn't offer a significant decrease in accuracy. The results of our experiments appear in Table 2.



(a) MNIST (b) CIFAR10

Figure 4. Training loss of adversarial examples during the training of the networks. The sharp drops in the CIFAR10 plot correspond to decreases in step size.

method	ε	#steps	source	accuracy
natural	-	-	-	87.3%
FGSM	8	-	A	56.1%
PGD	8	7	A	50.0%
PGD	8	20	A	**45.8%**
CW	8	30	A	**46.8%**
FGSM	8	-	A'	67.0%
PGD	8	7	A'	**64.2%**
CW	8	30	A'	78.7%
FGSM	8	-	A_{nat}	85.6%
PGD	8	7	A_{nat}	86.0%

Table 2. CIFAR10: Performance of the adversarially trained network against different adversaries. For each type of attack (black-box, white-box) we show the most effective attack in bold.

The adversarial robustness of our network is significant, given the power of iterative adversaries, but still far from satisfactory. We believe that these results can be improved by further pushing along these directions, and training networks of larger capacity.

6. Related Work

Due to the growing body of work on adversarial examples (Gu & Rigazio, 2014; Fawzi et al., 2015; Torkamani, 2016; Papernot et al., 2016; Carlini & Wagner, 2016a; Tramèr et al., 2017b; Goodfellow et al., 2014; Kurakin et al., 2016), we focus only on the most related papers here. Before we compare our contributions, we remark that robust optimization has been studied outside deep learning for multiple decades. We refer the reader to (Ben-Tal et al., 2009) for an overview of this field.

Recent work on adversarial training on ImageNet also observed that the model capacity is important for adversarial training (Kurakin et al., 2016). In contrast to this paper, we find that training against multi-step methods (PGD) *does* lead to resistance against such adversaries. Moreover, we study the loss landscape of the saddle point problem in more detail.

In (Huang et al., 2015b) and (Shaham et al., 2015) a version of the min-max optimization problem is also considered for adversarial training. There are, however, three important differences between the formerly mentioned result and the present paper. Firstly, the authors claim that the inner maximization problem can be difficult to solve, whereas we explore the loss surface in more detail and find that randomly re-started projected gradient descent often converges to solutions with comparable quality. This shows that it is possible to obtain sufficiently good solutions to the inner maximization problem, which offers good evidence that deep neural network can be immunized against adversarial examples. Secondly, they consider only one-step adversaries, while we work with multi-step methods. Additionally, while the experiments in (Shaham et al., 2015) produce promising results, they are only evaluated against FGSM. However, FGSM-only evaluations are not fully reliable. One evidence for that is that (Shaham et al., 2015) reports 70% accuracy for $\varepsilon = 0.7$, but any adversary that is allowed to perturb each pixel by more than 0.5 can construct a uniformly gray image, thus fooling any classifier.

A more recent paper (Tramèr et al., 2017b) also explores the transferability phenomenon. The authors propose a linear algebraic notion of adversarial subspaces. In our experiments, we find that larger model capacity and adversarial training reduces the transferability of adversarial examples. Moreover, we explore how the adversarial loss behaves along random directions. Overall, our experiments show that the structure of adversarial examples cannot be described fully by the linear subspace view.

7. Conclusion

Our findings provide evidence that deep neural networks can be made resistant to adversarial attacks. As our theory and experiments indicate, we can design reliable adversarial training methods. One of the key insights behind this is the unexpectedly regular structure of the underlying optimization task: even though the relevant problem corresponds to the maximization of a highly non-concave function with many distinct local maxima, their *values* are highly concentrated. Overall, our findings give us hope that adversarially robust deep learning models may be within current reach.

For the MNIST dataset, our networks are very robust, achieving high accuracy for a wide range of powerful adversaries and large perturbations. Our experiments on CIFAR10 have not reached the same level of performance yet. However, our results already show that our techniques lead to significant increase in the robustness of the network. We believe that further exploring this direction will lead to adversarially robust networks for this dataset.

References

Tensor flow models repository. https://github.com/tensorflow/models/tree/master/resnet.

Ben-Tal, Aharon, El Ghaoui, Laurent, and Nemirovski, Arkadi. *Robust optimization*. Princeton University Press, 2009.

Carlini, Nicholas and Wagner, David. Defensive distillation is not robust to adversarial examples. *arXiv preprint arXiv:1607.04311*, 2016a. URL http://arxiv.org/abs/1607.04311.

Carlini, Nicholas and Wagner, David. Towards evaluating the robustness of neural networks. *arXiv preprint arXiv:1608.04644*, 2016b. URL http://arxiv.org/abs/1608.04644.

Fawzi, Alhussein, Fawzi, Omar, and Frossard, Pascal. Analysis of classifiers' robustness to adversarial perturbations. *arXiv preprint arXiv:1502.02590*, 2015. URL http://arxiv.org/abs/1502.02590.

Goodfellow, Ian J., Shlens, Jonathon, and Szegedy, Christian. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. URL http://arxiv.org/abs/1412.6572.

Gu, Shixiang and Rigazio, Luca. Towards deep neural network architectures robust to adversarial examples. *arXiv preprint arXiv:1412.5068*, 2014. URL http://arxiv.org/abs/1412.5068.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Huang, Ruitong, Xu, Bing, Schuurmans, Dale, and Szepesvári, Csaba. Learning with a strong adversary. *arXiv preprint arXiv:1511.03034*, 2015a. URL http://arxiv.org/abs/1511.03034.

Huang, Ruitong, Xu, Bing, Schuurmans, Dale, and Szepesvári, Csaba. Learning with a strong adversary. *arXiv preprint arXiv:1511.03034*, 2015b. URL http://arxiv.org/abs/1511.03034.

Kurakin, Alexey, Goodfellow, Ian J., and Bengio, Samy. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016. URL http://arxiv.org/abs/1611.01236.

Moosavi-Dezfooli, Seyed-Mohsen, Fawzi, Alhussein, and Frossard, Pascal. Deepfool: A simple and accurate method to fool deep neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 2574–2582. IEEE Computer Society, 2016. ISBN 978-1-4673-8851-1. doi: 10.1109/CVPR.2016.282. URL http://dx.doi.org/10.1109/CVPR.2016.282.

Nguyen, Anh Mai, Yosinski, Jason, and Clune, Jeff. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pp. 427–436. IEEE Computer Society, 2015. ISBN 978-1-4673-6964-0. doi: 10.1109/CVPR.2015.7298640. URL http://dx.doi.org/10.1109/CVPR.2015.7298640.

Papernot, Nicolas and McDaniel, Patrick D. On the effectiveness of defensive distillation. *arXiv preprint arXiv:1607.05113*, 2016. URL http://arxiv.org/abs/1607.05113.

Papernot, Nicolas, McDaniel, Patrick D., Jha, Somesh, Fredrikson, Matt, Celik, Z. Berkay, and Swami, Ananthram. The limitations of deep learning in adversarial settings. In *IEEE European Symposium on Security and Privacy, EuroS&P 2016, Saarbrücken, Germany, March 21-24, 2016*, pp. 372–387. IEEE, 2016. ISBN 978-1-5090-1751-5. doi: 10.1109/EuroSP.2016.36. URL http://dx.doi.org/10.1109/EuroSP.2016.36.

Rozsa, Andras, Günther, Manuel, and Boult, Terrance E. Towards robust deep neural networks with BANG. *arXiv preprint arXiv:1612.00138*, 2016. URL http://arxiv.org/abs/1612.00138.

Shaham, Uri, Yamada, Yutaro, and Negahban, Sahand. Understanding adversarial training: Increasing local stability of neural nets through robust optimization. *arXiv preprint arXiv:1511.05432*, 2015. URL http://arxiv.org/abs/1511.05432.

Sharif, Mahmood, Bhagavatula, Sruti, Bauer, Lujo, and Reiter, Michael K. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In Weippl, Edgar R., Katzenbeisser, Stefan, Kruegel, Christopher, Myers, Andrew C., and Halevi, Shai (eds.), *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pp. 1528–1540. ACM, 2016. ISBN 978-1-4503-4139-4. doi: 10.1145/2976749.2978392. URL http://doi.acm.org/10.1145/2976749.2978392.

Sokolic, Jure, Giryes, Raja, Sapiro, Guillermo, and Rodrigues, Miguel RD. Robust large margin deep neural networks. *arXiv preprint arXiv:1605.08254*, 2016.

Szegedy, Christian, Zaremba, Wojciech, Sutskever, Ilya, Bruna, Joan, Erhan, Dumitru, Goodfellow, Ian J., and Fergus, Rob. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. URL http://arxiv.org/abs/1312.6199.

Torkamani, MohamadAli. *Robust Large Margin Approaches for Machine Learning in Adversarial Settings*. PhD thesis, University of Oregon, 2016.

Tramèr, Florian, Kurakin, Alexey, Papernot, Nicolas, Boneh, Dan, and McDaniel, Patrick D. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017a. URL http://arxiv.org/abs/1705.07204.

Tramèr, Florian, Papernot, Nicolas, Goodfellow, Ian J., Boneh, Dan, and McDaniel, Patrick D. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017b. URL http://arxiv.org/abs/1704.03453.

Wald, Abraham. Contributions to the theory of statistical estimation and testing hypotheses. *The Annals of Mathematical Statistics*, 10(4):299–326, 1939.

Wald, Abraham. Statistical decision functions which minimize the maximum risk. *Annals of Mathematics*, pp. 265–280, 1945.

Wald, Abraham. Statistical decision functions. In *Breakthroughs in Statistics*, pp. 342–357. Springer, 1992.