

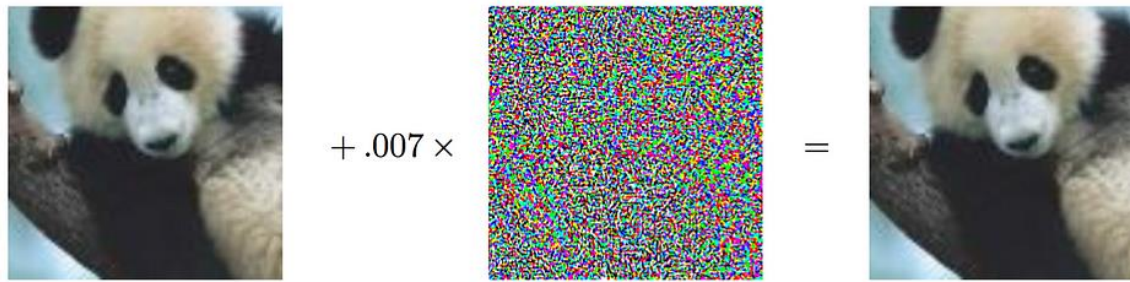
The background features a complex network of blue lines and arrows. Solid lines intersect at various angles, while dashed lines form loops and paths. Small circles, some solid and some hollow, are placed at various points along the lines, suggesting nodes or data points in a network or flow diagram.

TOWARDS DEEP LEARNING MODELS RESISTANT TO ADVERSARIAL ATTACKS

Lalitha Priya Garigapati
(50475268)

 **University at Buffalo**
School of Engineering and Applied Sciences

Adversarial Attack:



The diagram illustrates an adversarial attack. It shows a sequence of three images: a panda, a square of random noise, and the resulting panda image with the noise added. The noise is labeled as being multiplied by a factor of 0.007. Below each image is a label and a confidence percentage.

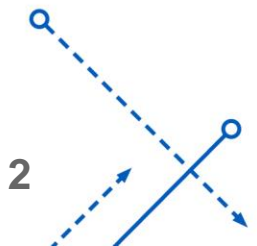
“panda”
57.7% confidence

+ .007 ×

noise

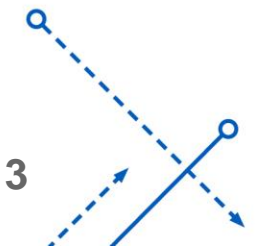
=

“gibbon”
99.3% confidence



What this paper adds...

- defining an adversary attack or defending the model
- guarantee on adversarial robustness

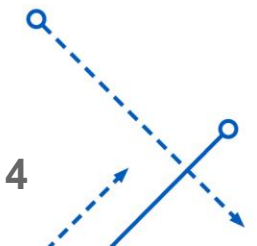


Ensuring Network Defense Against Worst-Case Adversarial Attacks:

Guarantee comes from the fact that network can defend against worst case adversarial attack.

How to ensure this

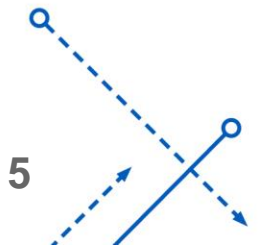
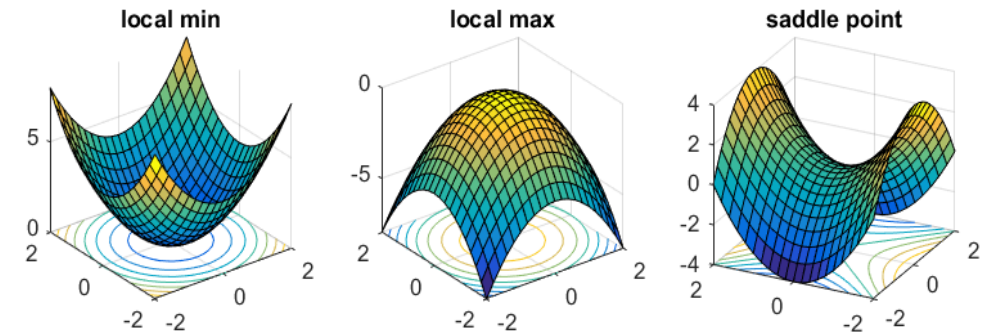
1. How can we produce strong adversarial examples, i.e., adversarial examples that fool a model with high confidence while requiring only a small perturbation?
2. How can we train a model so that there are no adversarial examples, or at least so that an adversary cannot find them easily?



Saddle Point:

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

- Inner maximization: (measure the strength of adversary)
 - aims to find an adversarial version of a given data point x that achieves a high loss.
 - Non-concave (local maximum)
- Outer minimization: (measure the strength of defense)
 - goal is to find model parameters so that the “adversarial loss” given by the inner attack problem is minimized. (by adjusting θ .)
 - Non-convex (local minimum)
- If loss value is 0 i.e. $L(\theta, x + \delta, y) = 0$ then model is robust.



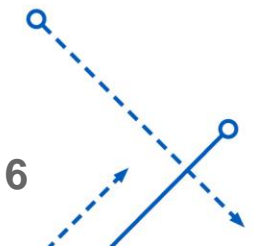
Adversarial Attacks:

Fast Gradient Sign Method (FGSM): $x + \varepsilon \operatorname{sgn}(\nabla_x L(\theta, x, y))$.

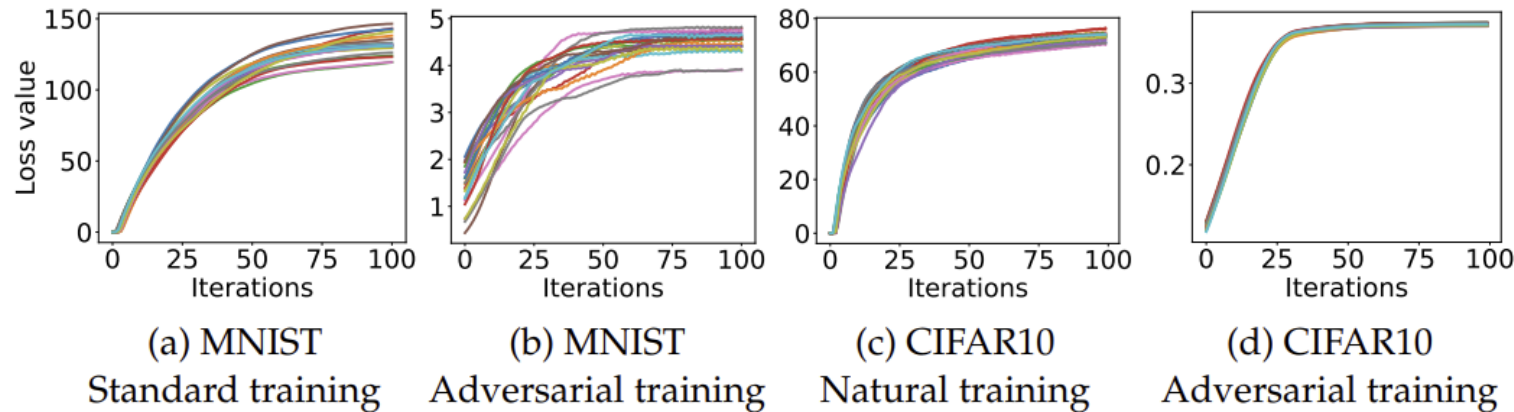
- It takes exactly one ε -sized step.
- Less training time and fails to increase robustness.
- Tune according loss gradient

Projected Gradient Method (PGD): $x^{t+1} = \Pi_{x+\mathcal{S}} (x^t + \alpha \operatorname{sgn}(\nabla_x L(\theta, x, y)))$.

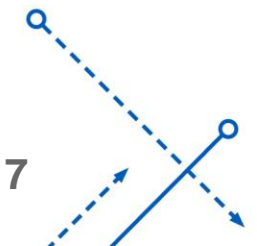
- One of the most effective attack.
- It requires too much training time as it takes multiple iterations(multi-step).
- Gives best possible adversary (within the constraint)



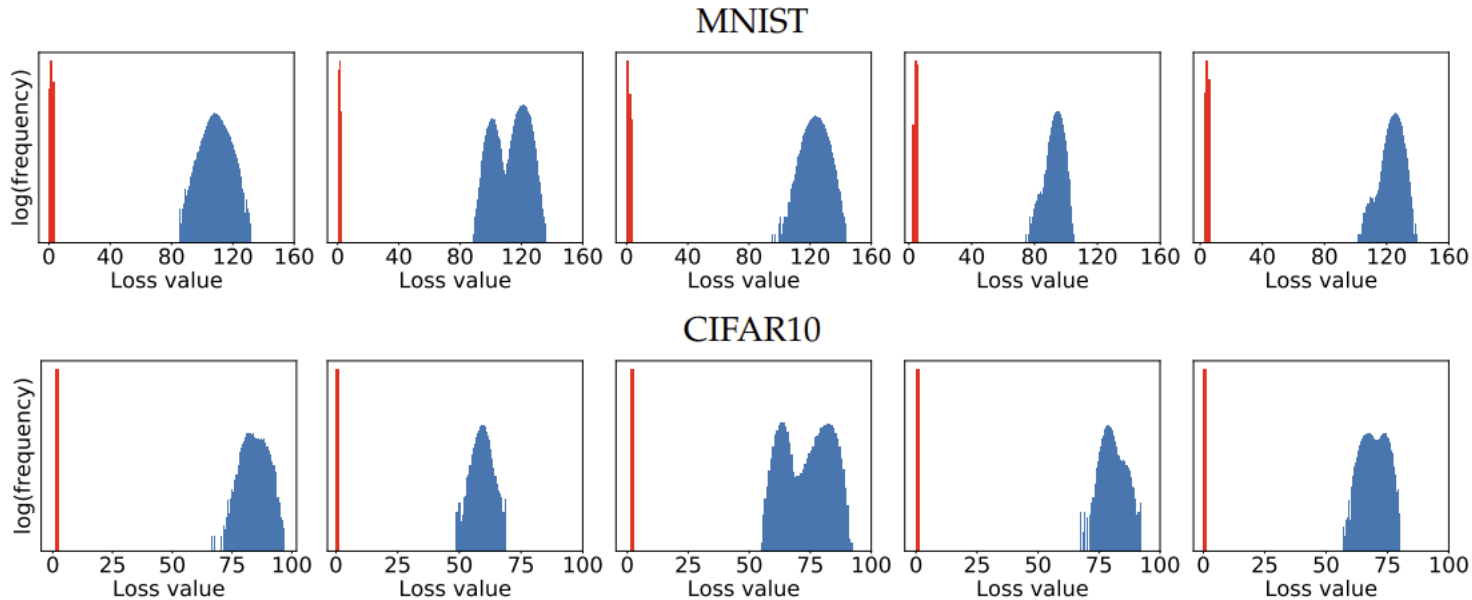
Exploring the local maxima of PGD



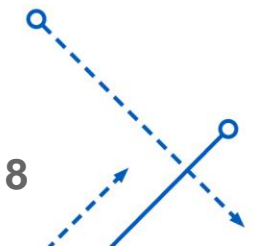
- PGD optimizes noise to the local maxima loss value.
- Randomly initialized value is within the set S of allowed perturbations bounded by L_∞
- For the given different starting points these are the graphs, and these all are consistent i.e. with similar loss value
- PGD iterated 100 times and adversarial models have much lower losses.



Exploring the local maxima of PGD



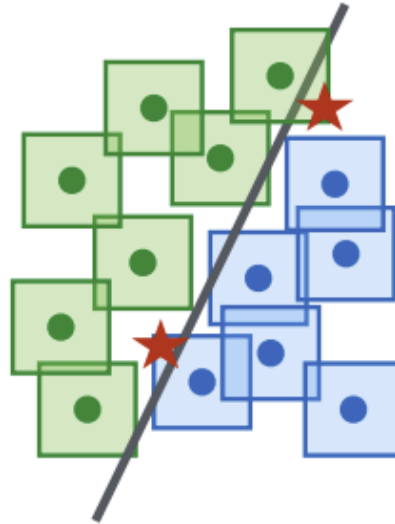
- Local maxima for random 5 images from each dataset
- **Blue** → loss on a standard network
- **Red** → loss on an adversarial trained part
- Loss significantly smaller for adversarial trained networks and they tend to be concentrated as well as with very few outliers.
- Even with many different start points, there is no global maxima, which is higher than other maximas.



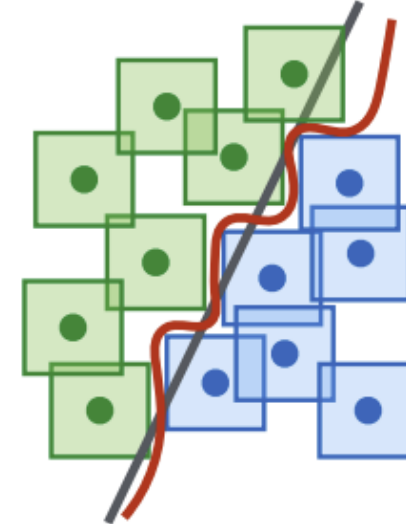
Normal vs Adversarial Decision Boundaries



Simple decision boundary
for standard input



Simple decision boundary
for Adversarial input



Adversarial decision boundary
for Adversarial input

- To reliably withstand strong adversarial attacks, networks require a larger capacity than for correctly classifying

Model Capacity's Impact - MNIST:

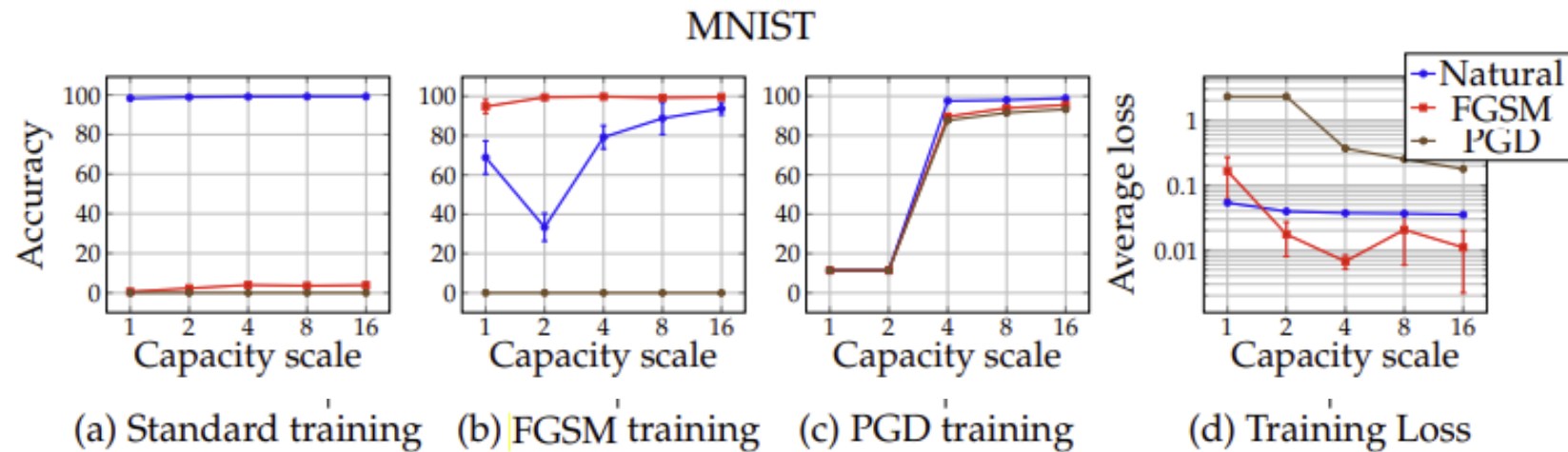


Figure 1: Cross-entropy loss values while creating an adversarial example from the MNIST and CIFAR10 evaluation datasets. The plots show how the loss evolves during 20 runs of projected gradient descent (PGD). Each run starts at a uniformly random point in the ℓ_∞ -ball around the same natural example (additional plots for different examples appear in Figure 11). The adversarial loss plateaus after a small number of iterations. The optimization trajectories and final loss values are also fairly clustered, especially on CIFAR10. Moreover, the final loss values on adversarially trained networks are significantly smaller than on their standard counterparts.

Model Capacity's Impact - CIFAR10:

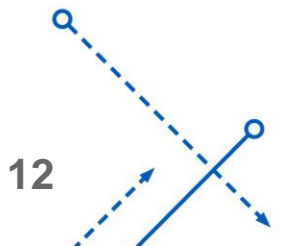
	CIFAR10							
	Simple	Wide	Simple	Wide	Simple	Wide	Simple	Wide
Natural	92.7%	95.2%	87.4%	90.3%	79.4%	87.3%	0.00357	0.00371
FGSM	27.5%	32.7%	90.9%	95.1%	51.7%	56.1%	0.0115	0.00557
PGD	0.8%	3.5%	0.0%	0.0%	43.7%	45.8%	1.11	0.0218
(a) Standard training			(b) FGSM training			(c) PGD training	(d) Training Loss	

Figure 1: Cross-entropy loss values while creating an adversarial example from the MNIST and CIFAR10 evaluation datasets. The plots show how the loss evolves during 20 runs of projected gradient descent (PGD). Each run starts at a uniformly random point in the ℓ_∞ -ball around the same natural example (additional plots for different examples appear in Figure 11). The adversarial loss plateaus after a small number of iterations. The optimization trajectories and final loss values are also fairly clustered, especially on CIFAR10. Moreover, the final loss values on adversarially trained networks are significantly smaller than on their standard counterparts.



Outcomes of Expanded Capacity:

- Improved robustness to one-step adversary through capacity expansion, especially for low values of ϵ .
- For large ϵ , Fast Gradient Sign Method (FGSM) adversaries result in overfitting.
- The small models are too small to derive any significant learning from PGD.
- Training on stronger adversaries and along with having a larger capacity results in lower transferability of these adversarial examples, which is desirable.



Training on PGD Input:

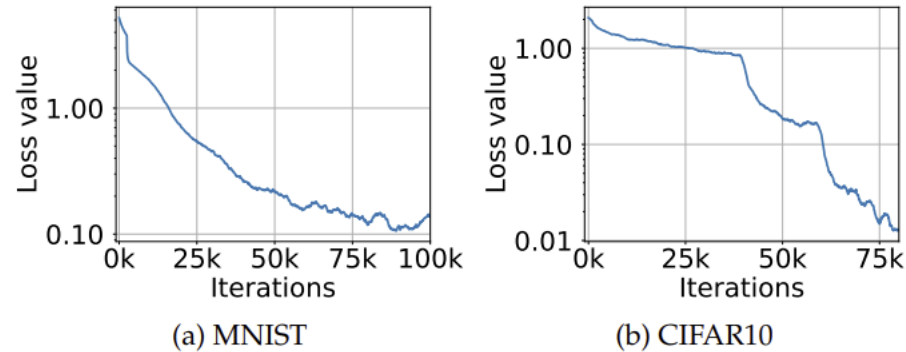


Figure 5: Cross-entropy loss on adversarial examples during training. The plots show how the adversarial loss on training examples evolves during training the MNIST and CIFAR10 networks against a PGD adversary. The sharp drops in the CIFAR10 plot correspond to decreases in training step size. These plots illustrate that we can consistently reduce the value of the inner problem of the saddle point formulation (2.1), thus producing an increasingly robust classifier.



Evaluation of Trained Models Against Various Adversarial Attacks:

The adversaries we consider are:

- White-box attacks with PGD for a different number of iterations and restarts, denoted by source A.
- White-box attacks with PGD using the Carlini-Wagner (CW) loss function. 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 Anat.



Results on Different Adversaries:

A → white box attack

A` → Black box attack using independently initialized and trained network.

B → Black box attack using similar network.

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%

Results on Different Adversaries – CIFAR 10:

$A \rightarrow$ white box attack

$A' \rightarrow$ Black box attack using independently initialized and trained network.

$A_{nat} \rightarrow$ Black box attack using copy of network trained on natural examples.

Method	Steps	Source	Accuracy
Natural	-	-	87.3%
FGSM	-	A	56.1%
PGD	7	A	50.0%
PGD	20	A	45.8%
CW	30	A	46.8%
FGSM	-	A'	67.0%
PGD	7	A'	64.2%
CW	30	A'	78.7%
FGSM	-	A_{nat}	85.6%
PGD	7	A_{nat}	86.0%

Results on adversaries of Different Strength:

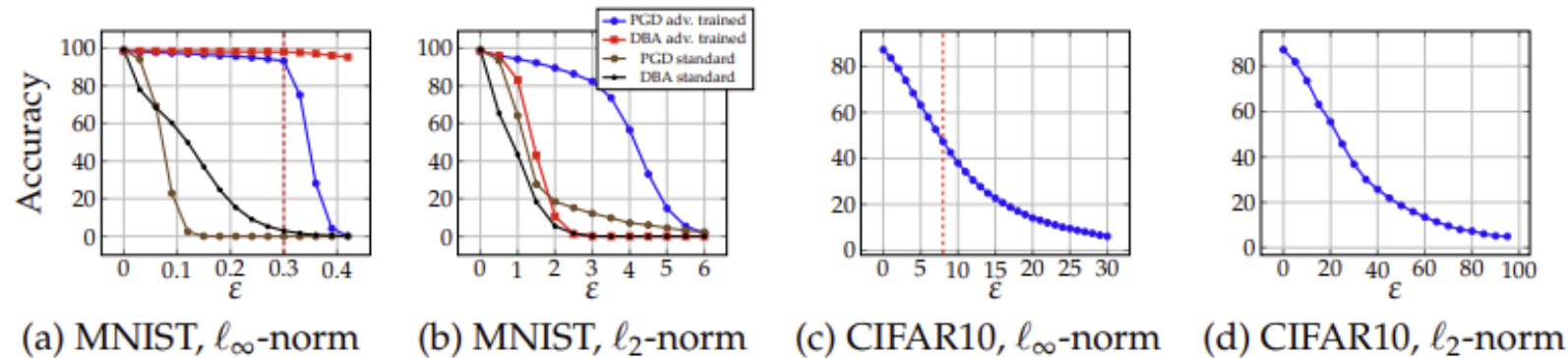


Figure 6: Performance of our adversarially trained networks against PGD adversaries of different strength. The MNIST and CIFAR10 networks were trained against $\epsilon = 0.3$ and $\epsilon = 8$ PGD ℓ_∞ adversaries respectively (the training ϵ is denoted with a red dashed lines in the ℓ_∞ plots). In the case of the MNIST adversarially trained networks, we also evaluate the performance of the Decision Boundary Attack (DBA) [4] with 2000 steps and PGD on standard and adversarially trained models. We observe that for ϵ less or equal to the value used during training, the performance is equal or better. For MNIST there is a sharp drop shortly after. Moreover, we observe that the performance of PGD on the MNIST ℓ_2 -trained networks is poor and significantly overestimates the robustness of the model. This is potentially due to the threshold filters learned by the model masking the loss gradients (the decision-based attack does not utilize gradients).

Conclusion:

- Training larger capacity networks on PGD adversaries leads to resistance against various attacks
- As the capacity increases, the value of the saddle point problem decreases for the given PGD adversarial model
- MNIST models are very robust against a range of powerful adversaries
- CIFAR10 significant increase in performance but not as robust
 - Further exploration will likely lead to increased robustness

