CS480/680: Introduction to Machine Learning

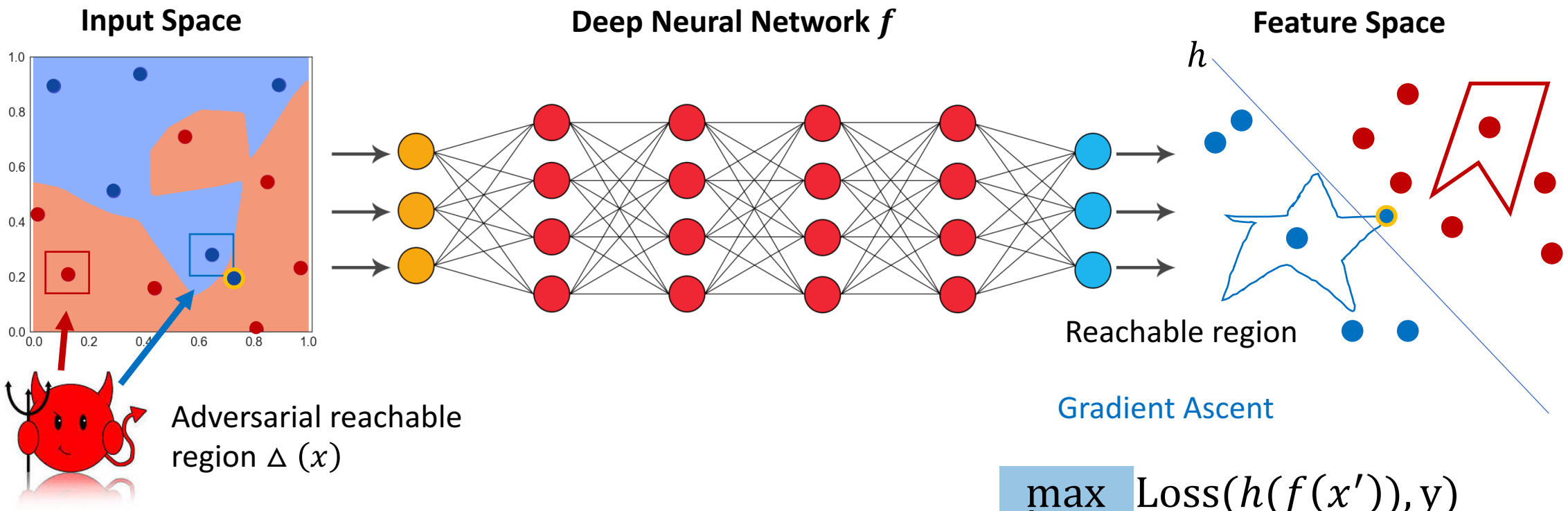
Lecture 15: Adversarial Robustness

Hongyang Zhang

UNIVERSITY OF WATERLOO

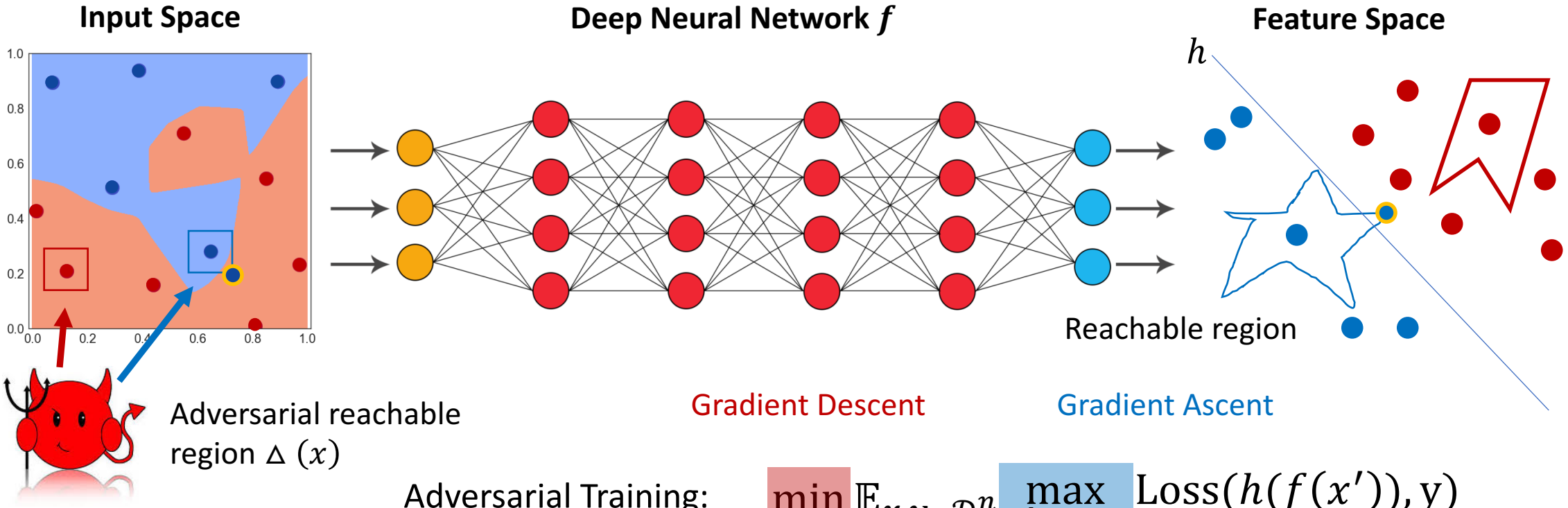
July 28, 2025

Categories of Adversarial Defense



- Categories of empirical defenses:
  - ***Gradient masking***
    - Design a method to hide the gradients in the classification model
    - This will make first-order attacks failed

Categories of Adversarial Defense



Adversarial Training: $\min_{h,f} \mathbb{E}_{x,y\sim\mathcal{D}^n} \max_{x'\in\Delta(x)} \text{Loss}(h(f(x')), y)$

- Categories of empirical defenses:
  - ***Gradient masking***
    - Design a method to hide the gradients in the classification model
    - This will make first-order attacks failed
  - ***Adversarial training***
    - Use the inner maximization to mimic the behavior of attacks
    - Use the outer minimization to update the weight of neural networks

Outline of the Lecture

- Gradient masking
  - Shattered gradient based method
  - Stochastic/randomized gradient based method
- Adversarial training
  - FGSM adversarial training
  - Ensemble adversarial training
  - PGD adversarial training
  - Trade-off between robustness and accuracy
  - TRADES

Outline of the Lecture

- Gradient masking
  - Shattered gradient based method
  - Stochastic/randomized gradient based method
- Adversarial training
  - FGSM adversarial training
  - Ensemble adversarial training
  - PGD adversarial training
  - Trade-off between robustness and accuracy
  - TRADES

Gradient Masking

- ***Gradient masking*** methods hide the gradient of the model from being used by an adversary
  - Because most attacks are based on the model's gradient information, creating adversarial examples with such attacks becomes less successful
- Gradient masking methods can be grouped into:
  - Shattered gradients methods
  - Stochastic/randomized gradients
- **Limitation:** they are designed to confuse the first-order attacks, but they cannot defend against other forms of attacks such as black-box attacks

Shattered Gradients

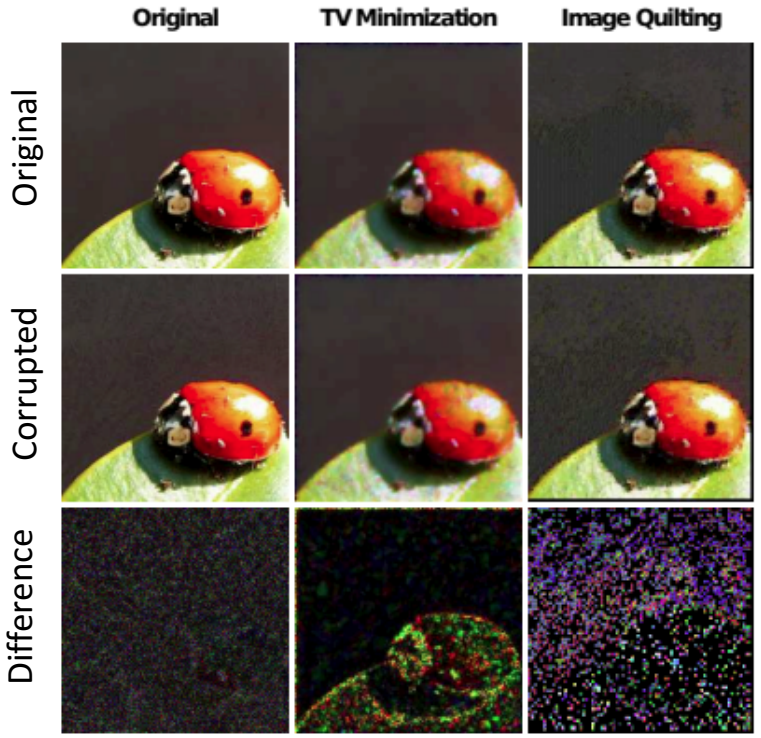
- ***Shattered gradients methods***
  - The goal is to prevent the flow of information from the inputs to the outputs in the model
  - By this, the adversaries are prevented from calculating the gradients
- A common approach towards this goal is to preprocess the input data
  - For example, by applying a non-smooth or non-differentiable preprocessor $g(.)$ to the inputs, and then training a DNN model f on the preprocessed inputs $g(x)$
  - The trained target classifier $f(g(x))$ is not differentiable w.r.t. the inputs x, causing the failure of adversarial attacks

Shattered Gradients

- Buckman (2018) Thermometer Encoding: One Hot Way to Resist Adversarial Examples
- ***Thermometer Encoding*** defense applies discretization of the intensity levels of each pixel into an l-dimensional vector
  - They propose a mechanism for the mapping table
  - The value of the pixel with intensity 0.13 is replaced by a 10-dimensional vector $[0111111111]$, but one can use other mappings as well
- The target classifier is trained using discrete vectors for all pixels, which breaks the calculation of the gradients
  - Experimental evaluation indicates increased robustness by the DNN models to adversarial examples

Shattered Gradients

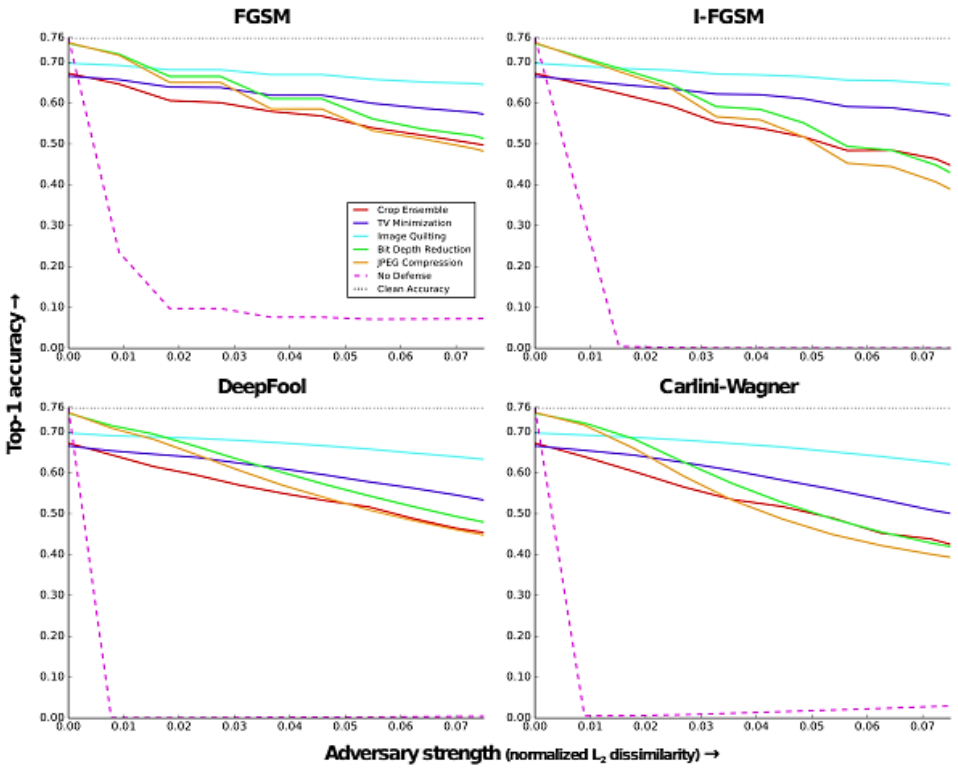
- Guo (2017) Countering Adversarial Images using Input Transformations
- This work employs several image transformation $g(.)$ to break the calculation of the gradients
  - These include: image cropping and rescaling, bit depth reduction, JPEG compression, total variance minimization, and image quilting



Shattered Gradients

- Guo (2017) Countering Adversarial Images using Input Transformations
- This work employs several image transformation $g(.)$ to break the calculation of the gradients
  - These include: image cropping and rescaling, bit depth reduction, JPEG compression, total variance minimization, and image quilting

- Evaluation of different attacks against ResNet on ImageNet
  - X-axis: perturbation size
  - Y-axis: accuracy (higher is better)
  - The accuracy increases from almost 0% with no defense, to over 60% for most settings

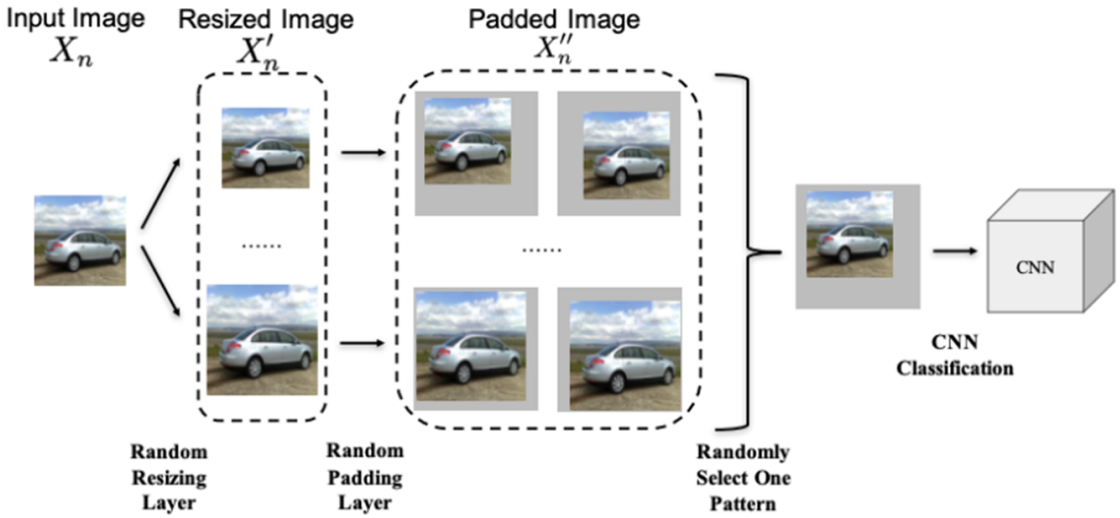


Stochastic/Randomized Gradients

- ***Stochastic/Randomized Gradients*** methods apply some form of randomization of the DNN model as a defense strategy to fool the adversary
  - E.g., train a set of classifiers, and during the testing phase randomly select one classifier to predict the class labels
  - Because the adversary does not know which model was used for prediction, the attack success rate is reduced

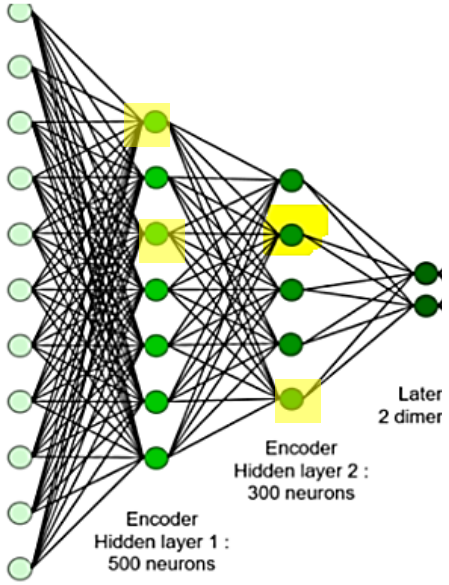
Stochastic/Randomized Gradients

- Xie (2018) Mitigating Adversarial Effects Through Randomization
- The defense approach applies random resizing and padding to improve the robustness to adversarial attacks
  - Images are first resized to several different widths and heights
  - Random padding with 0s is added to all four sides of the resized images
- For each image, prediction vectors are obtained for 30 randomized versions of the image, and the average value is adopted as the final prediction



Stochastic/Randomized Gradients

- Dhillon (2018) Stochastic Activation Pruning for Robust Adversarial Defense
- ***Stochastic Activation Pruning*** removes a random subset of neurons' activations in each layer
  - The remaining output activations in each layer are rescaled to normalize the inputs to the subsequent layer
  - This approach is similar to dropout layers



Gradient masking is a false sense of security

- All the above defenses break down by adaptive attacks
  - E.g., by applying black-box attacks which do not use gradients
- Therefore, gradient masking is proven to be a false sense of security

Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples

Anish Athalye [*1] Nicholas Carlini [*2] David Wagner [2]

Abstract

We identify obfuscated gradients, a kind of gradient masking, as a phenomenon that leads to a false sense of security in defenses against adversarial examples. While defenses that cause obfuscated gradients appear to defeat iterative optimization-based attacks, we find defenses relying on this effect can be circumvented. We describe characteristic behaviors of defenses exhibiting the effect, and for each of the three types of obfuscated gradients we discover, we develop attack techniques to overcome it. In a case study, examining non-certified white-box-secure defenses at ICLR 2018, we find obfuscated gradients are a common occurrence, with 7 of 9 defenses relying on obfuscated apparent robustness against iterative optimization attacks: *obfuscated gradients*, a term we define as a special case of gradient masking (Papernot et al., 2017). Without a good gradient, where following the gradient does not successfully optimize the loss, iterative optimization-based methods cannot succeed. We identify three types of obfuscated gradients: *shattered gradients* are nonexistent or incorrect gradients caused either intentionally through non-differentiable operations or unintentionally through numerical instability; *stochastic gradients* depend on test-time randomness; and *vanishing/exploding gradients* in very deep computation result in an unusable gradient.

We propose new techniques to overcome obfuscated gradients caused by these three phenomena. We address gradient shattering with a new attack technique we call Backward

Outline of the Lecture

- Gradient masking
  - Shattered gradient based method
  - Stochastic/randomized gradient based method
- Adversarial training
  - FGSM adversarial training
  - Ensemble adversarial training
  - PGD adversarial training
  - Trade-off between robustness and accuracy
  - TRADES

Adversarial Training

- ***Adversarial training*** trains the target classification model using adversarial examples

$$\min_{C} \mathbb{E}_{x,y\sim\mathcal{D}^n} \max_{x'\in\Delta(x)} \text{Loss}(C(x')), \text{y})$$

  - The adversarial examples are produced to attack the latest iterate of classifier
  - By adding adversarial examples x' with true label y to the training set, the model will learn that x' belongs to the class y
- Adversarial training is one of the most successful defenses so far

Adversarial Training

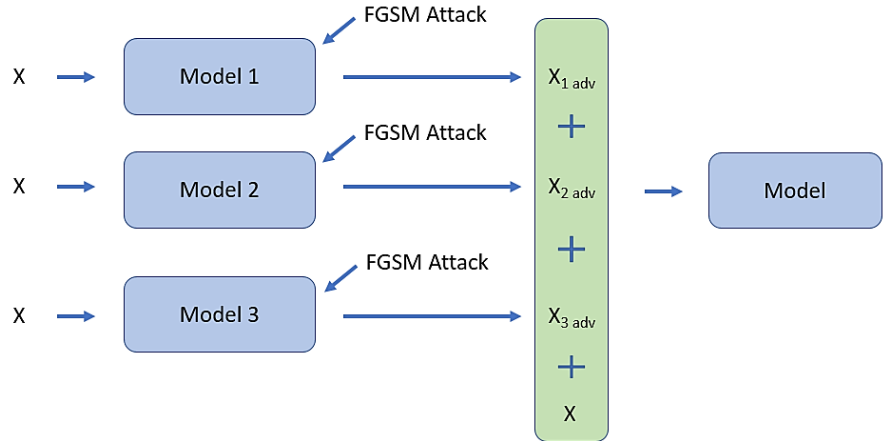
- Goodfellow (2015) Explaining and Harnessing Adversarial Examples

$$\min_{C} \mathbb{E}_{x,y\sim\mathcal{D}^n} \max_{x'\in\Delta(x)} \text{Loss}(C(x')), \text{y})$$

- The paper suggests using FGSM attack to solve the inner maximization
  - Adversarial examples created by FGSM were added to the training set to increase the model robustness
  - **Limitation:** the robust model is vulnerable to adversarial examples created by other attacks (e.g., PGD attacks)

Adversarial Training

- Tramer (2017) Ensemble Adversarial Training: Attacks and Defenses
- ***Ensemble Adversarial Training*** uses a set of adversarial examples created by several fixed classifiers to train the model
  - Model 1, Model 2, and Model 3 with different architectures are trained
  - For each input sample x, FGSM is used to create adversarial samples $x_{1\ adv}$, $x_{2\ adv}$, and $x_{3\ adv}$ using the three models
  - A classifier is trained using the clean sample x and the adversarial samples created by all three models $x_{1\ adv}$, $x_{2\ adv}$, and $x_{3\ adv}$
- The performance highly depends on the robustness of Models 1, 2, and 3



Adversarial Training

- Madry (2017) Towards Deep Learning Models Resistant to Adversarial Attacks

$$\min_{c} \mathbb{E}_{x,y\sim\mathcal{D}^n} \max_{x'\in\Delta(x)} \text{Loss}(c(x')), \text{y})$$

- This paper suggests using PGD attack to solve the inner maximization
  - PGD can find stronger adversarial example around an input sample *x*
- The trained model demonstrated good robustness on MNIST and CIFAR-10
  - Due to the high computational cost for created PGD samples, it is difficult to scale to large datasets such as ImageNet
  - Xie et al. (2019) Feature Denoising for Improving Adversarial Robustness
    - 128 Nvidia V100 GPUs and 52 hours to adversarially train a ResNet-152 model

Adversarial Training

- **Limitation:** Adversarial training suffers from a reduced accuracy on clean samples, known as robustness-accuracy trade-off
  - y-axis: the difference between the classification error of an adversarially trained model and a natually trained model
  - Adversarial training reduces the natural accuracy for about 3%~7%
  - Increasing the size of the dataset reduces the gap
  - Increasing the perturbation size increases the gap
  - Why the trade-off?

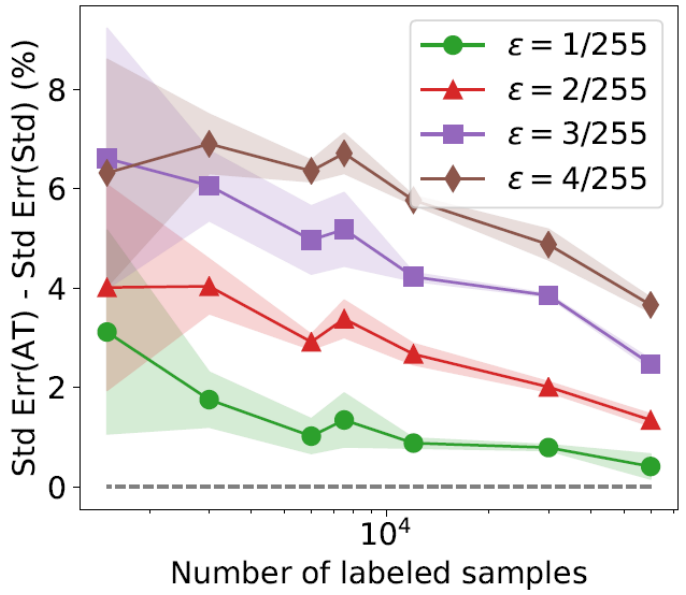


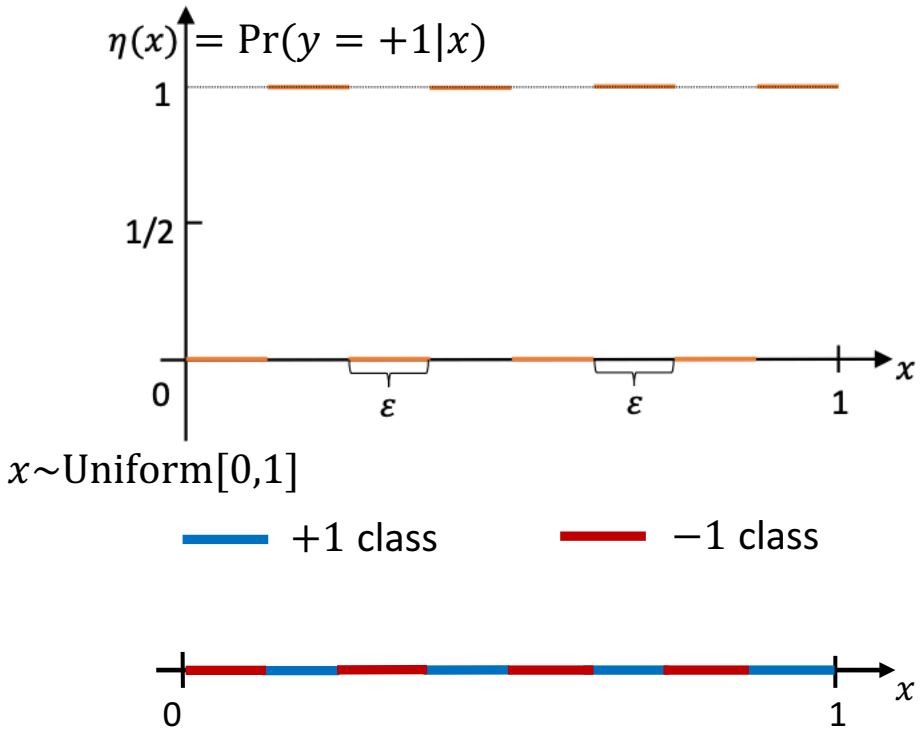
Figure from: Madry (2018) Towards Deep Learning Models Resistant to Adversarial Attacks

Trade-off between Robustness and Accuracy

$R_{rob}(f) := \Pr_{x,y\sim\mathcal{D}}\{\exists x' \in \Delta(x) \text{ s.t. } f(x')y \le 0\}$ $\quad$ $y \in \{+1,-1\}$, classifier $f: \mathcal{X} \to \mathbb{R}$

$$R_{nat}(f) := \Pr_{x,y\sim\mathcal{D}}\{f(x)y \le 0\}$$

- An example of trade-off (for $\Delta(x) = \mathbb{B}_p(x,\varepsilon)$):



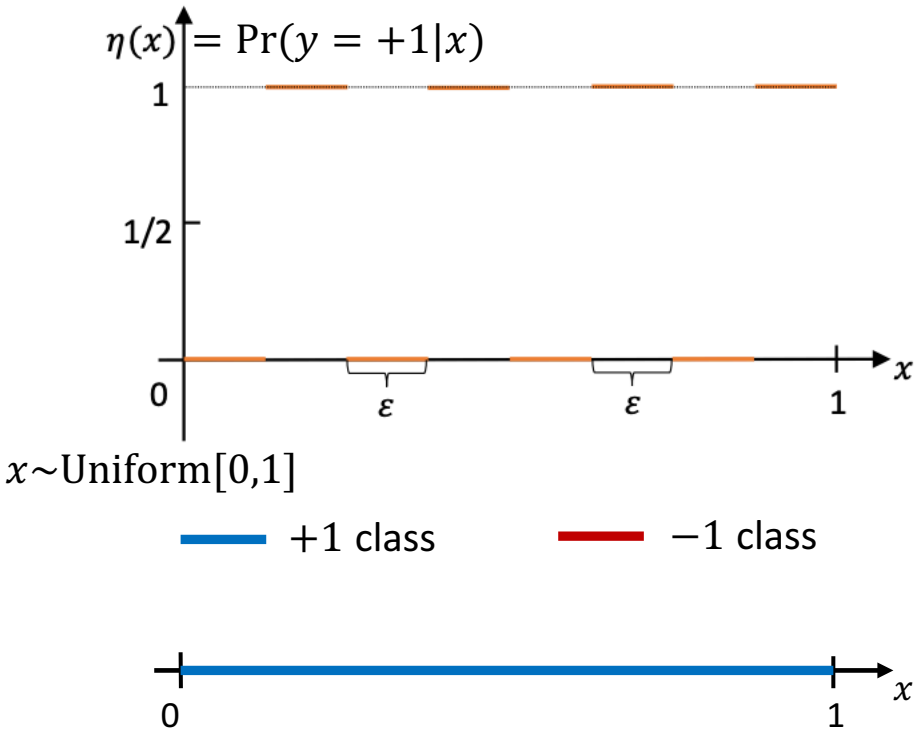
	Bayes Optimal Classifier
$R_{nat}(f)$	0 (best R_{nat})
$R_{rob}(f)$	1 (worst R_{rob})

Trade-off between Robustness and Accuracy

$R_{rob}(f) := \Pr_{x,y\sim\mathcal{D}}\{\exists x' \in \Delta(x) \text{ s.t. } f(x')y \le 0\}$ $\quad$ $y \in \{+1,-1\}$, classifier $f: \mathcal{X} \to \mathbb{R}$

$R_{nat}(f) := \Pr_{x,y\sim\mathcal{D}}\{f(x)y \le 0\}$

- An example of trade-off (for $\Delta(x) = \mathbb{B}_p(x,\varepsilon)$):



	Bayes Optimal Classifier	All +1 Classifier
$R_{nat}(f)$	0 (best R_{nat})	1/2
$R_{rob}(f)$	1 (worst R_{rob})	1/2 (best R_{rob})

Solution: minimize $\min_f R_{nat}(f) + R_{rob}(f)/\lambda$

Computationally, weighted average $R_{nat}(f) + R_{rob}(f)/\lambda$ is non-differentiable.

Classification-Calibrated Surrogate Loss

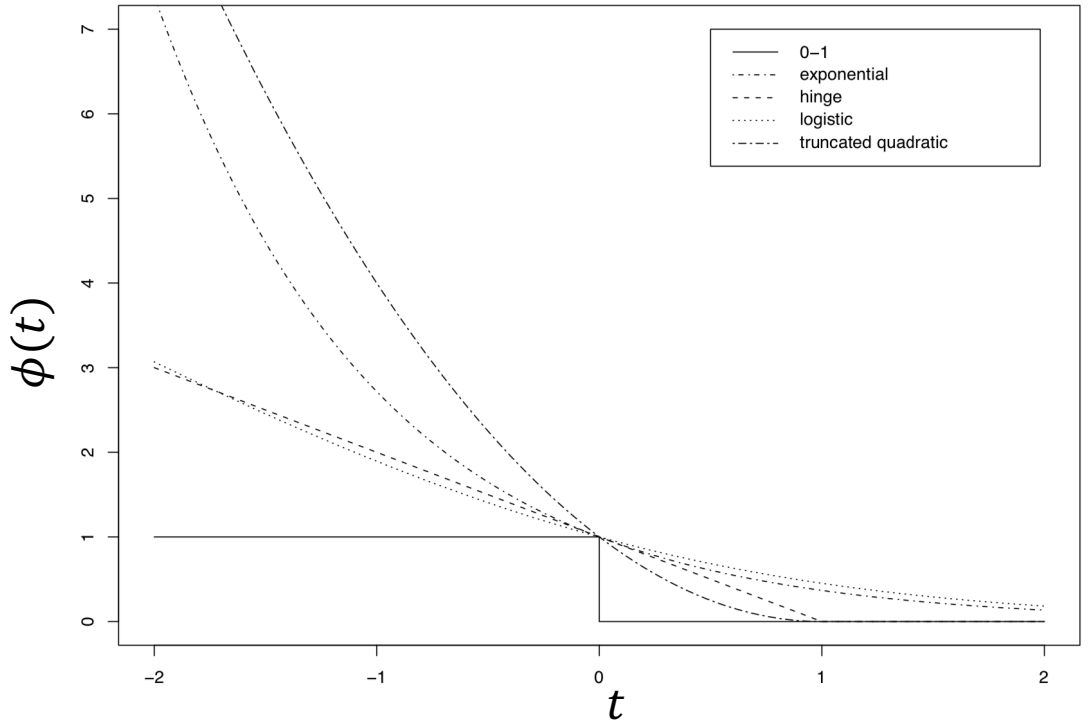
$$R_{rob}(f) := \mathbb{E}_{x,y\sim\mathcal{D}} 1\{\exists x' \in \Delta(x) \text{ s.t. } f(x')y \leq 0\}$$

Can we design a differentiable surrogate loss for the trade-off?

$$R_{nat}(f) := \mathbb{E}_{x,y\sim\mathcal{D}} 1\{f(x)y \leq 0\}$$

[Bartlett et al.'06] approximate

$$R_{\phi}(f) := \mathbb{E}_{x,y\sim\mathcal{D}} \phi(f(x)y)$$



[Bartlett et al.'06] Convexity, Classification, and Risk Bounds, Journal of the American Statistical Association, 2006

TRADES

minimize difference between $f(x)$ and y for accuracy

$$\min_f [\mathbb{E}_{x,y\sim\mathcal{D}}\, \phi(f(x)y) + \mathbb{E}_{x,y\sim\mathcal{D}} \max_{x'\in\Delta(x)} \phi(f(x)f(x')/\lambda)]$$

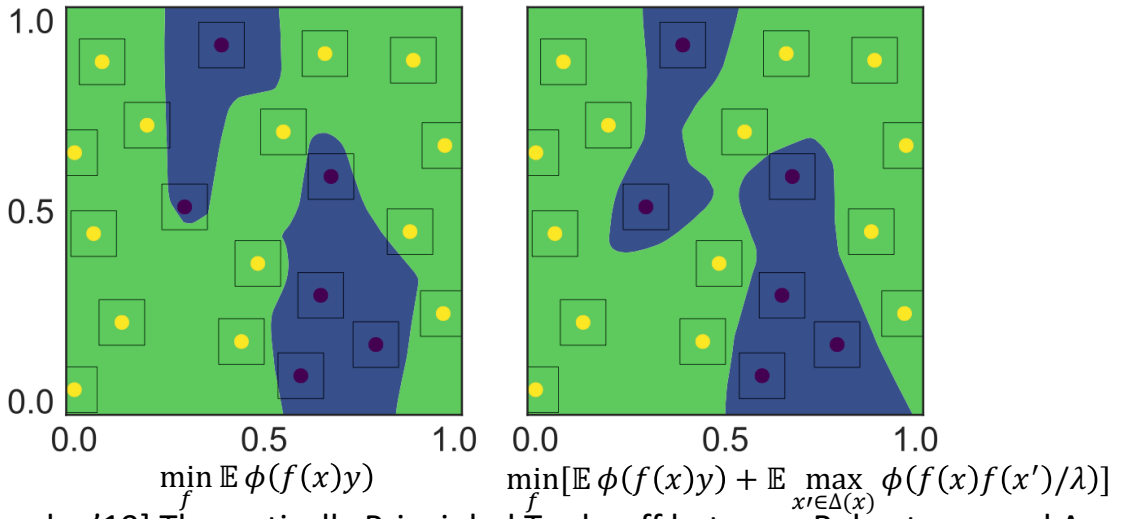
[Zhang, Yu, Jiao, Xing, Ghaoui, Jordan'19] Theoretically Principled Trade-off between Robustness and Accuracy, ICML 2019

TRADES

minimize difference between $f(x)$ and $f(x')$ for robustness

$$\min_f [\mathbb{E}_{x,y\sim\mathcal{D}}\, \phi(f(x)y) + \mathbb{E}_{x,y\sim\mathcal{D}} \max_{x'\in\Delta(x)} \phi(f(x)f(x')/\lambda)]$$

Gradient Ascent



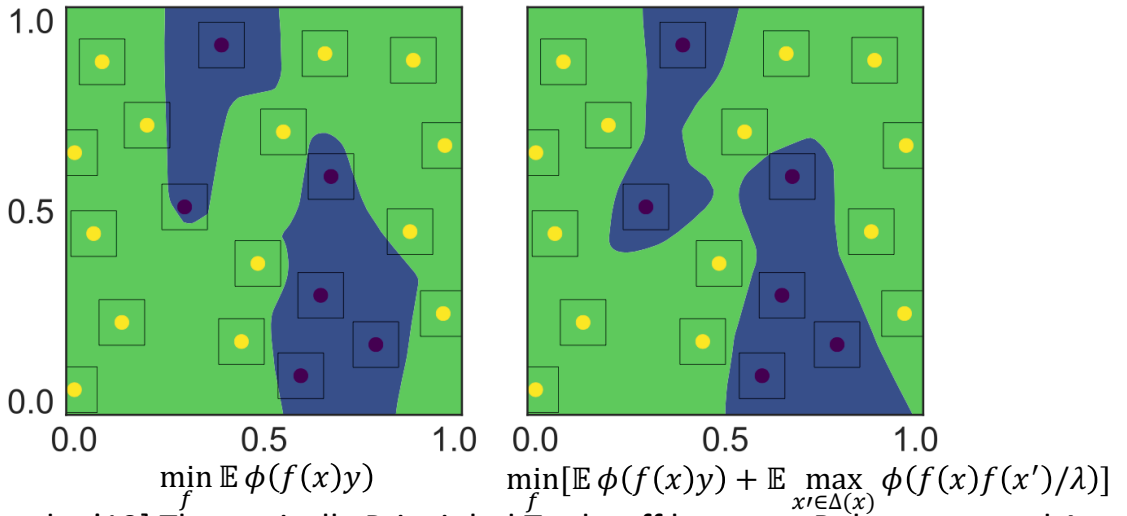
[Zhang, Yu, Jiao, Xing, Ghaoui, Jordan'19] Theoretically Principled Trade-off between Robustness and Accuracy, ICML 2019

TRADES

TRADES Loss(f):

$$\min_f [\mathbb{E}_{x,y\sim\mathcal{D}}\, \phi(f(x)y) + \mathbb{E}_{x,y\sim\mathcal{D}} \max_{x'\in\Delta(x)} \phi(f(x)f(x')/\lambda)]$$

Gradient Ascent



[Zhang, Yu, Jiao, Xing, Ghaoui, Jordan'19] Theoretically Principled Trade-off between Robustness and Accuracy, ICML 2019

Theoretical Results

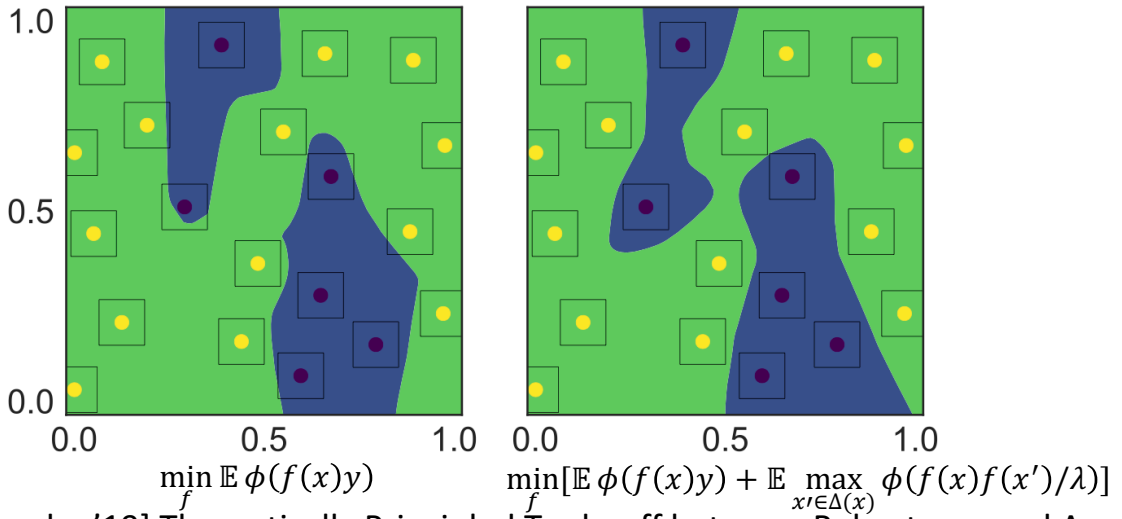
Theorem 1 (Informal, upper bound, Zhang et al.'19):

For any distribution $\mathcal{D}$, f, $\Delta(x)$ and $\lambda > 0$, we have $R_{rob}(f) - R^*_{nat} \leq \text{TRADES Loss}(f) - R^*_{\phi}$.

- R^*_{nat}: minimal value of $R_{nat}(f)$ over all classifiers f
- R^*_{ϕ}: minimal value of $R_{\phi}(f) := \mathbb{E}_{x,y\sim\mathcal{D}}\phi(f(x)y)$ over all classifiers f
- ϕ: classification-calibrated surrogate loss

TRADES Loss(f):

$$\min_f[\mathbb{E}_{x,y\sim\mathcal{D}}\,\phi(f(x)y) + \mathbb{E}_{x,y\sim\mathcal{D}}\max_{x'\in\Delta(x)}\phi(f(x)f(x')/\lambda)]$$



[Zhang, Yu, Jiao, Xing, Ghaoui, Jordan'19] Theoretically Principled Trade-off between Robustness and Accuracy, ICML 2019

Theoretical Results

Theorem 1 (Informal, upper bound, Zhang et al.'19):

For any distribution $\mathcal{D}$, f, $\Delta(x)$ and $\lambda > 0$, we have $R_{rob}(f) - R_{nat}^* \leq \text{TRADES Loss}(f) - R_\phi^*$.

- R_{nat}^*: minimal value of $R_{nat}(f)$ over all classifiers f
- R_ϕ^*: minimal value of $R_\phi(f) := \mathbb{E}_{x,y\sim\mathcal{D}}\phi(f(x)y)$ over all classifiers f
- ϕ: classification-calibrated surrogate loss

Theorem 2 (Informal, lower bound, Zhang et al.'19):

For any $\Delta(x)$, there exist a data distribution $\mathcal{D}$, a classifier f, and an $\lambda > 0$ such that

$$R_{rob}(f) - R_{nat}^* \geq \text{TRADES Loss}(f) - R_\phi^*.$$

[Zhang, Yu, Jiao, Xing, Ghaoui, Jordan'19] Theoretically Principled Trade-off between Robustness and Accuracy, ICML 2019

Experiments --- CIFAR10 with 8-intensity level attacks

Defense	Defense type	Under which attack	Dataset	Distance	Natural Accuracy $\mathcal{A}_{\text{nat}}(f)$	Robust Accuracy $\mathcal{A}_{\text{rob}}(f)$
Buckman et al. (2018)	gradient mask	Athalye et al. (2018)	CIFAR10	0.031 (ℓ_∞)	-	0%
Ma et al. (2018)	gradient mask	Athalye et al. (2018)	CIFAR10	0.031 (ℓ_∞)	-	5%
Dhillon et al. (2018)	gradient mask	Athalye et al. (2018)	CIFAR10	0.031 (ℓ_∞)	-	0%
Song et al. (2018)	gradient mask	Athalye et al. (2018)	CIFAR10	0.031 (ℓ_∞)	-	9%
Na et al. (2017)	gradient mask	Athalye et al. (2018)	CIFAR10	0.015 (ℓ_∞)	-	15%
Wong et al. (2018)	robust opt.	FGSM^{20} (PGD)	CIFAR10	0.031 (ℓ_∞)	27.07%	23.54%
Madry et al. (2018)	robust opt.	FGSM^{20} (PGD)	CIFAR10	0.031 (ℓ_∞)	87.30%	**47.04%**

$$\min_f \mathbb{E} \max_{x\prime \in \mathbb{B}(x,\varepsilon)} \phi(f(x')y)$$ (by Madry et al.)

TRADES ($1/\lambda = 1.0$)	regularization	FGSM^{20} (PGD)	CIFAR10	0.031 (ℓ_∞)	88.64%	49.14%
TRADES ($1/\lambda = 6.0$)	regularization	FGSM^{20} (PGD)	CIFAR10	0.031 (ℓ_∞)	84.92%	**56.61%**

$$\min_f [\mathbb{E}\, \phi(f(x)y) + \mathbb{E} \max_{x\prime \in \mathbb{B}(x,\varepsilon)} \phi(f(x)f(x')/\lambda)]$$ (TRADES)

TRADES ($1/\lambda = 6.0$)	regularization	LBFGSAttack	CIFAR10	0.031 (ℓ_∞)	84.92%	81.58%
TRADES ($1/\lambda = 1.0$)	regularization	MI-FGSM	CIFAR10	0.031 (ℓ_∞)	88.64%	51.26%
TRADES ($1/\lambda = 6.0$)	regularization	MI-FGSM	CIFAR10	0.031 (ℓ_∞)	84.92%	57.95%
TRADES ($1/\lambda = 1.0$)	regularization	C&W	CIFAR10	0.031 (ℓ_∞)	88.64%	84.03%
TRADES ($1/\lambda = 6.0$)	regularization	C&W	CIFAR10	0.031 (ℓ_∞)	84.92%	81.24%
Samangouei et al. (2018)	gradient mask	Athalye et al. (2018)	MNIST	0.005 (ℓ_2)	-	55%
Madry et al. (2018)	robust opt.	FGSM^{40} (PGD)	MNIST	0.3 (ℓ_∞)	99.36%	96.01%
TRADES ($1/\lambda = 6.0$)	regularization	FGSM^{40} (PGD)	MNIST	0.3 (ℓ_∞)	99.48%	96.07%
TRADES ($1/\lambda = 6.0$)	regularization	C&W	MNIST	0.005 (ℓ_2)	99.48%	99.46%

Overview of This Talk

Robustness

Paradigms

Adversarial Examples

Random Noises

Mixed Random/Adversarial Corruptions

Empirical Defenses

Certified Defenses

Adversarial Defenses

Norm-Bounded Adversarial Example

Unrestricted Adversarial Example

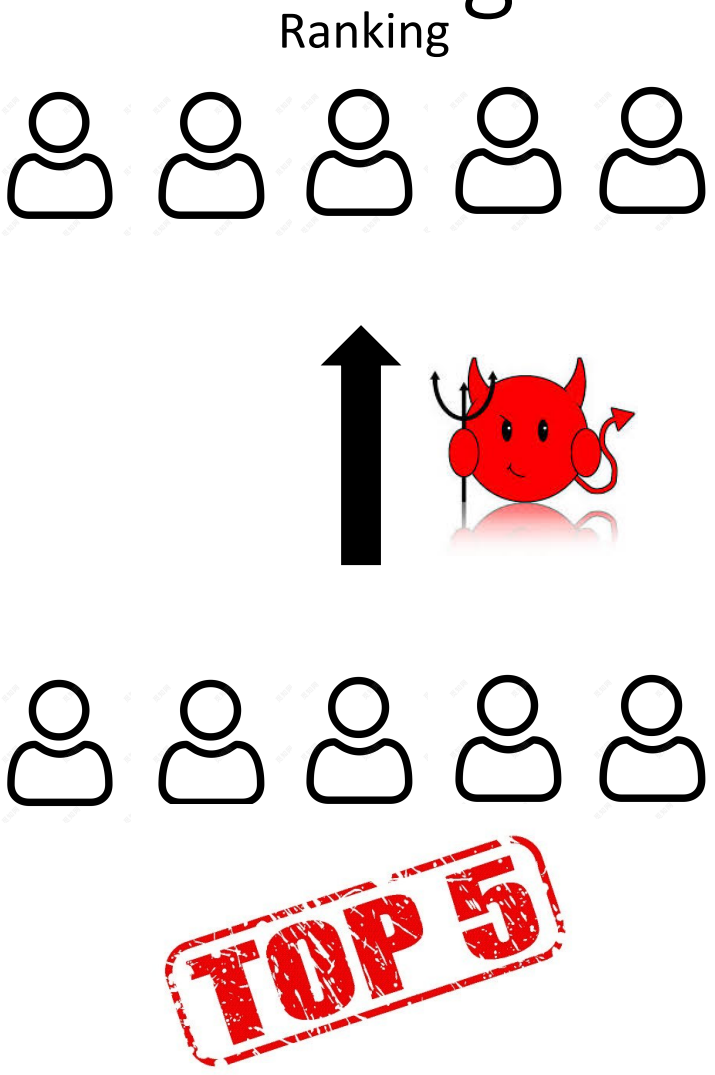
Positive Results

Hardness Results

Significant Experimental Results via Case Study

Applications

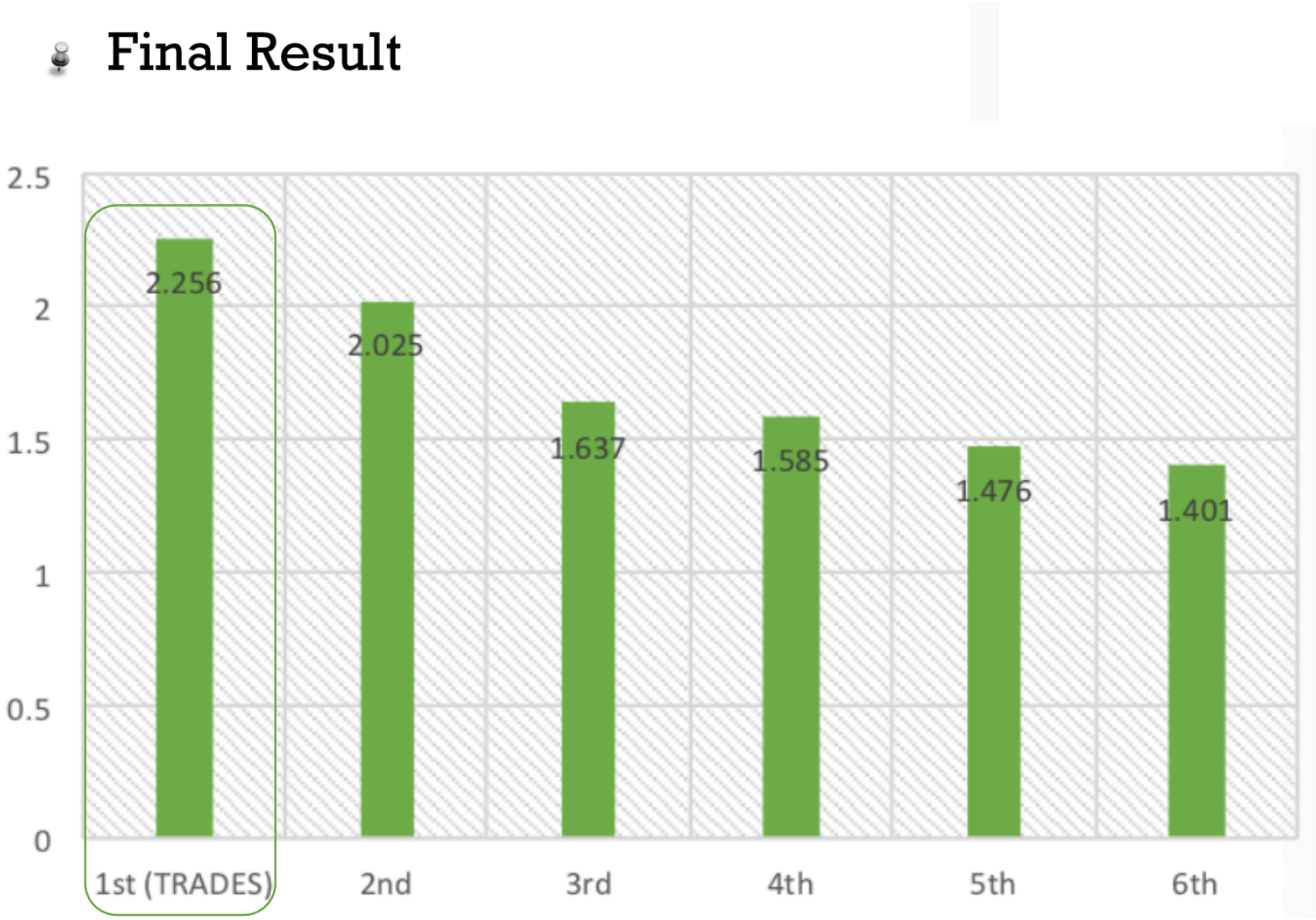
Case Study I: NeurIPS'18 Adversarial Vision Challenge



- Evaluation criterion
  - 400+ teams, ~3,000 submissions
  - ImageNet dataset
  - Model Track and Attack Track
  - Participants in the two tracks play against each other

Case Study I: NeurIPS'18 Adversarial Vision Challenge

y-axis: mean ℓ_2 perturbation distance to let a classifier make a mistake



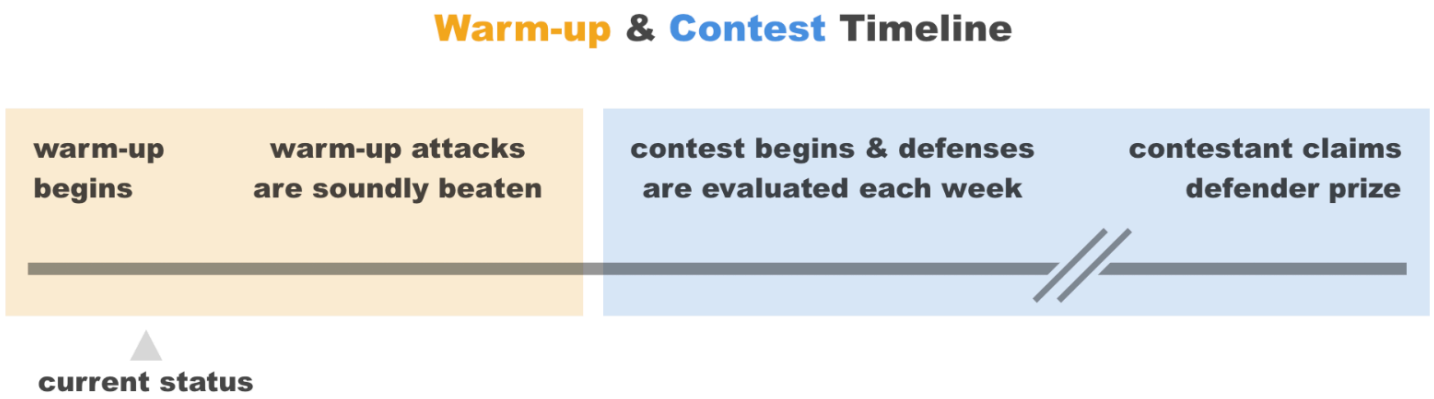
Case Study II: Unrestricted Adversarial Examples Challenge

Google

Unrestricted Adversarial Examples Challenge build passing

In the Unrestricted Adversarial Examples Challenge, attackers submit arbitrary adversarial inputs, and defenders are expected to assign low confidence to difficult inputs while retaining high confidence and accuracy on a clean, unambiguous test set. You can learn more about the motivation and structure of the contest in our recent paper

This repository contains code for the warm-up to the challenge, as well as the public proposal for the contest. We are currently accepting defenses for the warm-up.



Case Study II: Unrestricted Adversarial Examples Challenge

The class of bicycle

The class of bird

Case Study II: Unrestricted Adversarial Examples Challenge

Our methodology:

$$\min_f [\mathbb{E}\,\phi(f(x)y) + \mathbb{E}\max_{x' \in \Delta(x)} \phi(f(x)f(x')/\lambda)]$$

Choose the adversarial reachable region as the union of these threat models

Defense	Submitted by	Clean data	Common corruptions	Spatial grid attack	SPSA attack	Boundary attack	Submission Date
Pytorch ResNet50 (trained on bird-or-bicycle extras)	TRADES	100.0%	100.0%	99.5%	100.0%	95.0%	Jan 17th, 2019 (EST)
Keras ResNet (trained on ImageNet)	Google Brain	100.0%	99.2%	92.2%	1.6%	4.0%	Sept 29th, 2018
Pytorch ResNet (trained on bird-or-bicycle extras)	Google Brain	98.8%	74.6%	49.5%	2.5%	8.0%	Oct 1st, 2018

Case Study II: Unrestricted Adversarial Examples Challenge

Clean image:

Defense	Submitted by	Clean data	Common corruptions	Spatial grid attack	SPSA attack	Boundary attack	Submission Date
Pytorch ResNet50 (trained on bird-or-bicycle extras)	TRADES	100.0%	100.0%	99.5%	100.0%	95.0%	Jan 17th, 2019 (EST)
Keras ResNet (trained on ImageNet)	Google Brain	100.0%	99.2%	92.2%	1.6%	4.0%	Sept 29th, 2018
Pytorch ResNet (trained on bird-or-bicycle extras)	Google Brain	98.8%	74.6%	49.5%	2.5%	8.0%	Oct 1st, 2018

Case Study II: Unrestricted Adversarial Examples Challenge

Clean image:

Corrupted image:

Defense	Submitted by	Clean data	Common corruptions	Spatial grid attack	SPSA attack	Boundary attack	Submission Date
Pytorch ResNet50 (trained on bird-or-bicycle extras)	TRADES	100.0%	100.0%	99.5%	100.0%	95.0%	Jan 17th, 2019 (EST)
Keras ResNet (trained on ImageNet)	Google Brain	100.0%	99.2%	92.2%	1.6%	4.0%	Sept 29th, 2018
Pytorch ResNet (trained on bird-or-bicycle extras)	Google Brain	98.8%	74.6%	49.5%	2.5%	8.0%	Oct 1st, 2018

Case Study II: Unrestricted Adversarial Examples Challenge

Clean image:

Corrupted image:

Defense	Submitted by	Clean data	Common corruptions	Spatial grid attack	SPSA attack	Boundary attack	Submission Date
Pytorch ResNet50 (trained on bird-or-bicycle extras)	TRADES	100.0%	100.0%	99.5%	100.0%	95.0%	Jan 17th, 2019 (EST)
Keras ResNet (trained on ImageNet)	Google Brain	100.0%	99.2%	92.2%	1.6%	4.0%	Sept 29th, 2018
Pytorch ResNet (trained on bird-or-bicycle extras)	Google Brain	98.8%	74.6%	49.5%	2.5%	8.0%	Oct 1st, 2018

Case Study II: Unrestricted Adversarial Examples Challenge

Clean image:

Corrupted image:

Defense	Submitted by	Clean data	Common corruptions	Spatial grid attack	SPSA attack	Boundary attack	Submission Date
Pytorch ResNet50 (trained on bird-or-bicycle extras)	TRADES	100.0%	100.0%	99.5%	100.0%	95.0%	Jan 17th, 2019 (EST)
Keras ResNet (trained on ImageNet)	Google Brain	100.0%	99.2%	92.2%	1.6%	4.0%	Sept 29th, 2018
Pytorch ResNet (trained on bird-or-bicycle extras)	Google Brain	98.8%	74.6%	49.5%	2.5%	8.0%	Oct 1st, 2018

Case Study II: Unrestricted Adversarial Examples Challenge

Clean image:

Corrupted image:

Adversarial example around the decision boundary

Defense	Submitted by	Clean data	Common corruptions	Spatial grid attack	SPSA attack	Boundary attack	Submission Date
Pytorch ResNet50 (trained on bird-or-bicycle extras)	TRADES	100.0%	100.0%	99.5%	100.0%	95.0%	Jan 17th, 2019 (EST)
Keras ResNet (trained on ImageNet)	Google Brain	100.0%	99.2%	92.2%	1.6%	4.0%	Sept 29th, 2018
Pytorch ResNet (trained on bird-or-bicycle extras)	Google Brain	98.8%	74.6%	49.5%	2.5%	8.0%	Oct 1st, 2018

Interpretability of TRADES --- Adversarial Examples by Boundary Attack

The class of bicycle

The class of bird

RobustBench

A standardized benchmark for adversarial robustness

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

EPFL

PRINCETON UNIVERSITY

Rank	Method	Standard accuracy	Robust accuracy	Extra data	Architecture	Venue
1	Uncovering the Limits of Adversarial Training against Norm-Bounded Adversarial Examples *We show the robust accuracy reported in the paper since AutoAttack performs slightly worse (65.88%).*	91.10%	65.87%	☑	WideResNet-70-16	arXiv, Oct 2020
2	Uncovering the Limits of Adversarial Training against Norm-Bounded Adversarial Examples *We show the robust accuracy reported in the paper since AutoAttack performs slightly worse (62.80%).*	89.48%	62.76%	☑	WideResNet-28-10	arXiv, Oct 2020
3	Adversarial Weight Perturbation Helps Robust Generalization	88.25%	60.04%	☑	WideResNet-28-10	NeurIPS 2020
4	Does Network Width Really Help Adversarial Robustness?	85.60%	59.78%	☑	WideResNet-34-15	arXiv, Oct 2020
5	Unlabeled Data Improves Adversarial Robustness	89.69%	59.53%	☑	WideResNet-28-10	NeurIPS 2019

All top 10 methods use TRADES as their training algorithms.

Adaptive Attacks against TRADES

- TRADES motivates new attacks:

*Powered by TRADES CIFAR-10 Challenge on 

Attack	Submitted by	Attack Model	Robust Acc	Time
PGD-20	(initial entry)	ℓ_∞, 8 intensity	56.61%	Jan 24, 2019
PGD-1,000	(initial entry)	ℓ_∞, 8 intensity	56.43%	Jan 24, 2019

[Croce et al.'20] Minimally Distorted Adversarial Examples with A Fast Adaptive Boundary Attack, ICML 2020.
[Gowal et al.'19] An Alternative Surrogate Loss for PGD-based Adversarial Testing, arXiv 2019.
[Tashiro et al.'20] Diversity Can Be Transferred, NeurIPS 2020.

Adaptive Attacks against TRADES

- TRADES motivates new attacks:

*Powered by TRADES CIFAR-10 Challenge on 

Attack	Submitted by	Attack Model	Robust Acc	Time
PGD-20	(initial entry)	ℓ_∞, 8 intensity	56.61%	Jan 24, 2019
PGD-1,000	(initial entry)	ℓ_∞, 8 intensity	56.43%	Jan 24, 2019
fab-attack	U. of Tubingen	ℓ_∞, 8 intensity	53.44%	Jun 7, 2019

[Croce et al.'20] Minimally Distorted Adversarial Examples with A Fast Adaptive Boundary Attack, ICML 2020.
[Gowal et al.'19] An Alternative Surrogate Loss for PGD-based Adversarial Testing, arXiv 2019.
[Tashiro et al.'20] Diversity Can Be Transferred, NeurIPS 2020.

Adaptive Attacks against TRADES

- TRADES motivates new attacks:

*Powered by TRADES CIFAR-10 Challenge on 

Attack	Submitted by	Attack Model	Robust Acc	Time
PGD-20	(initial entry)	ℓ_∞, 8 intensity	56.61%	Jan 24, 2019
PGD-1,000	(initial entry)	ℓ_∞, 8 intensity	56.43%	Jan 24, 2019
fab-attack	U. of Tubingen	ℓ_∞, 8 intensity	53.44%	Jun 7, 2019
MultiTargeted	DeepMind	ℓ_∞, 8 intensity	53.07%	Oct 31, 2019

[Croce et al.'20] Minimally Distorted Adversarial Examples with A Fast Adaptive Boundary Attack, ICML 2020.
[Gowal et al.'19] An Alternative Surrogate Loss for PGD-based Adversarial Testing, arXiv 2019.
[Tashiro et al.'20] Diversity Can Be Transferred, NeurIPS 2020.

Adaptive Attacks against TRADES

- TRADES motivates new attacks:

*Powered by TRADES CIFAR-10 Challenge on 

Attack	Submitted by	Attack Model	Robust Acc	Time
PGD-20	(initial entry)	ℓ_∞, 8 intensity	56.61%	Jan 24, 2019
PGD-1,000	(initial entry)	ℓ_∞, 8 intensity	56.43%	Jan 24, 2019
fab-attack	U. of Tubingen	ℓ_∞, 8 intensity	53.44%	Jun 7, 2019
MultiTargeted	DeepMind	ℓ_∞, 8 intensity	53.07%	Oct 31, 2019
ODI-PGD	Stanford	ℓ_∞, 8 intensity	53.01%	Feb 16, 2020

[Croce et al.'20] Minimally Distorted Adversarial Examples with A Fast Adaptive Boundary Attack, ICML 2020.
[Gowal et al.'19] An Alternative Surrogate Loss for PGD-based Adversarial Testing, arXiv 2019.
[Tashiro et al.'20] Diversity Can Be Transferred, NeurIPS 2020.

Adaptive Attacks against TRADES

- TRADES motivates new attacks:

*Powered by TRADES CIFAR-10 Challenge on 

Attack	Submitted by	Attack Model	Robust Acc	Time
PGD-20	(initial entry)	ℓ_∞, 8 intensity	56.61%	Jan 24, 2019
PGD-1,000	(initial entry)	ℓ_∞, 8 intensity	56.43%	Jan 24, 2019
fab-attack	U. of Tubingen	ℓ_∞, 8 intensity	53.44%	Jun 7, 2019
MultiTargeted	DeepMind	ℓ_∞, 8 intensity	53.07%	Oct 31, 2019
ODI-PGD	Stanford	ℓ_∞, 8 intensity	53.01%	Feb 16, 2020
CAA	Xiaofeng Mao	ℓ_∞, 8 intensity	52.94%	Dec 14, 2020
EWR-PGD	Ye Liu	ℓ_∞, 8 intensity	52.92%	Dec 20, 2020

[Croce et al.'20] Minimally Distorted Adversarial Examples with A Fast Adaptive Boundary Attack, ICML 2020.
[Gowal et al.'19] An Alternative Surrogate Loss for PGD-based Adversarial Testing, arXiv 2019.
[Tashiro et al.'20] Diversity Can Be Transferred, NeurIPS 2020.

Summary

- Gradient masking
  - Shattered gradient based method
  - Stochastic/randomized gradient based method
  - Gradient masking is a false sense of security
- Adversarial training
  - FGSM adversarial training
  - Ensemble adversarial training
  - PGD adversarial training
  - Trade-off between robustness and accuracy
  - TRADES