Notice

- Due dates
  - Written Paper Critiques (on 01.24)
  - Term Project (on 01.19 – **Today**)
  - Checkpoint Presentation 1 (on 31st)

- Sign-up (on Canvas)
  - Scribe Lecture Note
  - In-class Paper Presentation / Discussion

CS 499/599: Machine Learning Security

01.19: Adversarial Examples (AE) 4

Mon/Wed 12:00 – 1:50 pm

Sanghyun Hong

sanghyun.hong@oregonstate.edu

Oregon State University

SAIL

Secure AI Systems Lab

Recap

- AE←Security
  - Practical considerations
  - Iterative Methods
  - Real-world exploitation
- AE←Security
  - Motivation
  - C&W Attack
  - Conclusions (and Implications)
- AE←ML
  - Motivation
  - PGD Attack
  - Conclusions (and Implications)

Topics for Today

- AE←Security
  - Practical considerations
  - Iterative Methods
  - Real-world exploitation
- AE←Security
  - Motivation
  - C&W Attack
  - Conclusions (and Implications)
- AE←ML
  - Motivation
  - PGD Attack
  - Conclusions (and Implications)

Madry et al, Towards Deep Learning Models Resistant to Adversarial Attacks

Motivation

- We still do know know…
  - RQ 1: What should be **the strongest attack** we use for evaluating NNs?
  - RQ 2: How can we train NNs **robust to** AEs?

Revisit the Threat Model

- Suppose
  - D: data distribution
  - (x, y): a datapoint in D; $x \in R^d$ and $y \in [k]$; $x \in [0, 1]$
  - f: a neural network; θ: its parameters
  - $L(\theta, x, y)$: a loss function
- Attacker's Goal
  - Find an $x^{adv} = x + \delta$ such that $f\left(x^{adv}\right) \neq y$ while $||\delta||_p \leq \varepsilon$

Revisit the Threat Model

- Suppose
  - D: data distribution
  - (x, y): a datapoint in D; $x \in R^d$ and $y \in [k]$; $x \in [0, 1]$
  - f: a neural network; θ: its parameters
  - $L(\theta, x, y)$: a loss function
- Attacker's Goal
  - Find an $x^{adv} = x + \delta$ such that $\max_{\delta \in S} L\left(\theta, x^{adv}, y\right)$ while $||\delta||_p \leq \varepsilon$
- Our Goal
  - Train a f; find θ such that $\min_{\theta} \rho(\theta)$ where $\rho(\theta) = \mathrm{E}_{(x,y)\sim D}\left[L\left(\theta, x^{adv}, y\right)\right]$

Revisit the Threat Model

- Suppose
  - D: data distribution
  - (x, y): a datapoint in D; $x \in R^d$ and $y \in [k]$; $x \in [0, 1]$
  - f: a neural network; θ: its parameters
  - $L(\theta, x, y)$: a loss function
- Unified view (saddle point problem)
  - Find $\min_{\theta} \rho(\theta)$ where $\rho(\theta) = \mathrm{E}_{(x,y)\sim D}\left[\max_{\delta \in S} L(\theta, x + \delta, y)\right]$ while $||\delta||_p \leq \varepsilon$

RQ 1: What Should Be the Strongest Attack We Use?

- Projected Gradient Descent (PGD)

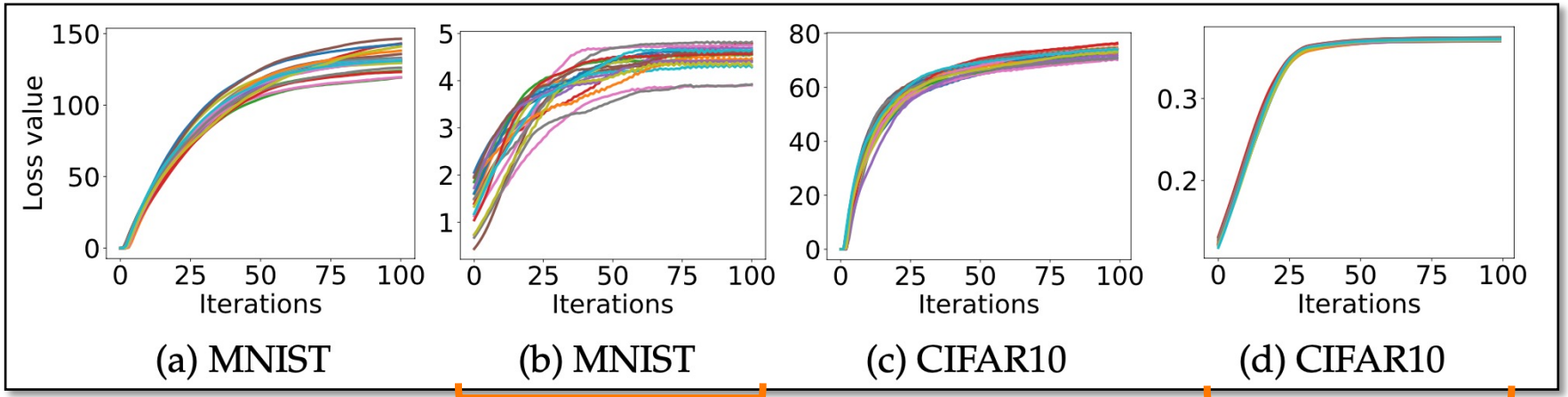
$$x^{t+1} = \Pi_{x+\mathcal{S}}\left(x^t + \alpha\,\mathrm{sgn}(\nabla_x L(\theta, x, y))\right).$$

FGSM

  - Q1: Is it a tractable problem?
  - Q2: Can training f against PGD provide robustness to all first-order adversaries?

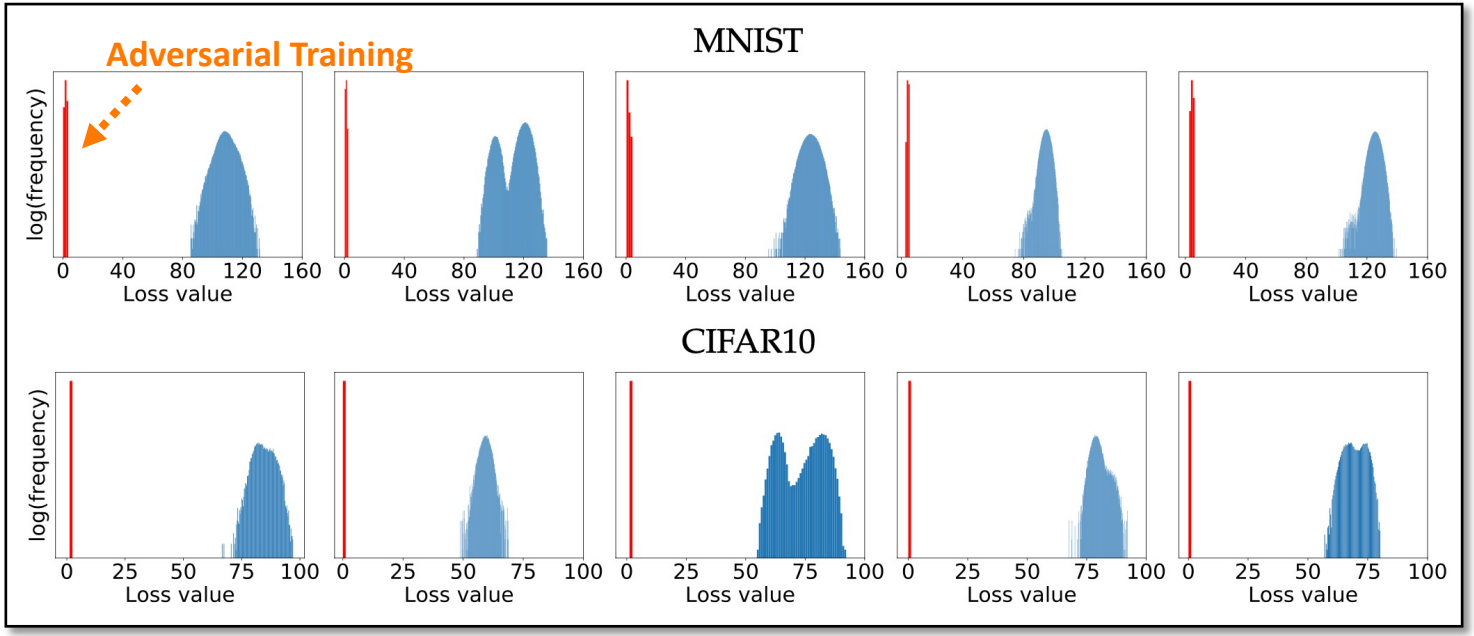
RQ 1: What Should Be the Strongest Attack We Use?

- Q1: Is This Tractable Problem?
  - Concentration of the maximum loss found by PGD
    - Restart PGD from many random points and measure the loss



RQ 1: What Should Be the Strongest Attack We Use?

- Q1: Is This Tractable Problem?
  - Concentration of the maximum loss found by PGD
    - Restart PGD from many random points and measure the loss



Conclusions (So far)

- Lessons
  - RQ 1: What should be **the strongest attack** we use for evaluating NNs?
    - PGD (a first-order adversary)

Topics for Today

- AE←ML
  - Motivation
  - PGD Attack
  - Conclusions (and Implications)
- Transferability
  - Motivation
  - Evaluate Transferability
  - Improve Transferability
  - Connection to Models' Properties
  - Conclusions (and Implications)

Motivation

- Transferability Is Important
  - Adversaries may not have white-box knowledge
  - But they can find (or built) surrogate models (easily ?!)

Motivation

- Research Questions
  - RQ 1: How much do adversarial examples **transfer** between models?
  - RQ 2: (If they don't) How can we **improve** the transferability of AEs?
  - RQ 3: How much do AEs transfer in **real-world scenarios**?
  - RQ 4: Why do they **transfer**?

RQ 1: How Much Do AEs Transfer Between Models?

- Setup
  - ImageNet (not the MNIST models)
    - Use 100 images chosen randomly from the test-set
  - ResNet-50/-101/-152, GoogleNet, and VGG-16

- Metric
  - Matching rate: the accuracy of AEs crafted on Model A transfer to Model B (target)
  - Distortion: the root mean square deviation

- Attacks
  - Optimization-based attack (similar to C&W)
  - Fast Gradient-based attack (similar to PGD)
  - Two scenarios: non-targeted and targeted attacks

RQ1: How Much Do AEs Transfer Between Models?

- Results from Non-targeted Attacks

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
ResNet-152	22.83	0%	13%	18%	19%	11%
ResNet-101	23.81	19%	0%	21%	21%	12%
ResNet-50	22.86	23%	20%	0%	21%	18%
VGG-16	22.51	22%	17%	17%	0%	5%
GoogLeNet	22.58	39%	38%	34%	19%	0%

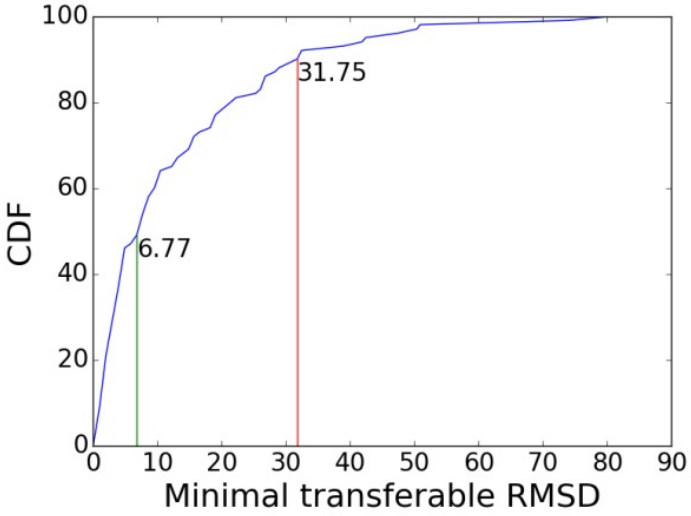
Panel A: Optimization-based approach

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
ResNet-152	23.45	4%	13%	13%	20%	12%
ResNet-101	23.49	19%	4%	11%	23%	13%
ResNet-50	23.49	25%	19%	5%	25%	14%
VGG-16	23.73	20%	16%	15%	1%	7%
GoogLeNet	23.45	25%	25%	17%	19%	1%

Panel B: Fast gradient approach

RQ1: How Much Do AEs Transfer Between Models?

- Distortion vs. Matching Rate
  - VGG-16 to ResNet-152



(a) Fast Gradient

RQ1: How Much Do AEs Transfer Between Models?

- Results from Non-targeted Attacks

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
ResNet-152	23.13	100%	2%	1%	1%	1%
ResNet-101	23.16	3%	100%	3%	2%	1%
ResNet-50	23.06	4%	2%	100%	1%	1%
VGG-16	23.59	2%	1%	2%	100%	1%
GoogLeNet	22.87	1%	1%	0%	1%	100%

RQ 2: How Can We Improve the Transferability?

- Key Intuition
  - **Ensemble** approach: use more surrogate models to craft adversarial examples

RQ 2: How Can We Improve the Transferability?

- Results (Optimization-based attacks)

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
-ResNet-152	30.68	38%	76%	70%	97%	76%
-ResNet-101	30.76	75%	43%	69%	98%	73%
-ResNet-50	30.26	84%	81%	46%	99%	77%
-VGG-16	31.13	74%	78%	68%	24%	63%
-GoogLeNet	29.70	90%	87%	83%	99%	11%

	RMSD	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
-ResNet-152	17.17	0%	0%	0%	0%	0%
-ResNet-101	17.25	0%	1%	0%	0%	0%
-ResNet-50	17.25	0%	0%	2%	0%	0%
-VGG-16	17.80	0%	0%	0%	6%	0%
-GoogLeNet	17.41	0%	0%	0%	0%	5%

RQ 3: In the Real-world Scenarios?

- Victim
  - Clarifai.com (~~You can do as well~~)
- Setup
  - ImageNet
    - Choose 100 images randomly from the test-set
  - ResNet-50/-101, GoogleNet and VGG-16
- Metrics
  - Matching rate: the accuracy of AEs crafted on Model A transfer to Clarfai.com
- Attacks
  - Optimization-based attack (similar to C&W)
  - Two scenarios: non-targeted and targeted attacks

RQ 3: In the Real-world Scenarios?

- Results
  - Non-targeted: most of AEs transfer to Clarifai.com
  - Targeted:
    - Just misclassifications:
      - 57% AEs crafted on VGG-16 transfer
      - 76% AEs crafted on the ensemble transfer
    - Misclassification **towards a target label**
      - 2% AEs crafted on VGG-16 transfer
      - 18% AEs crafted on the ensemble transfer

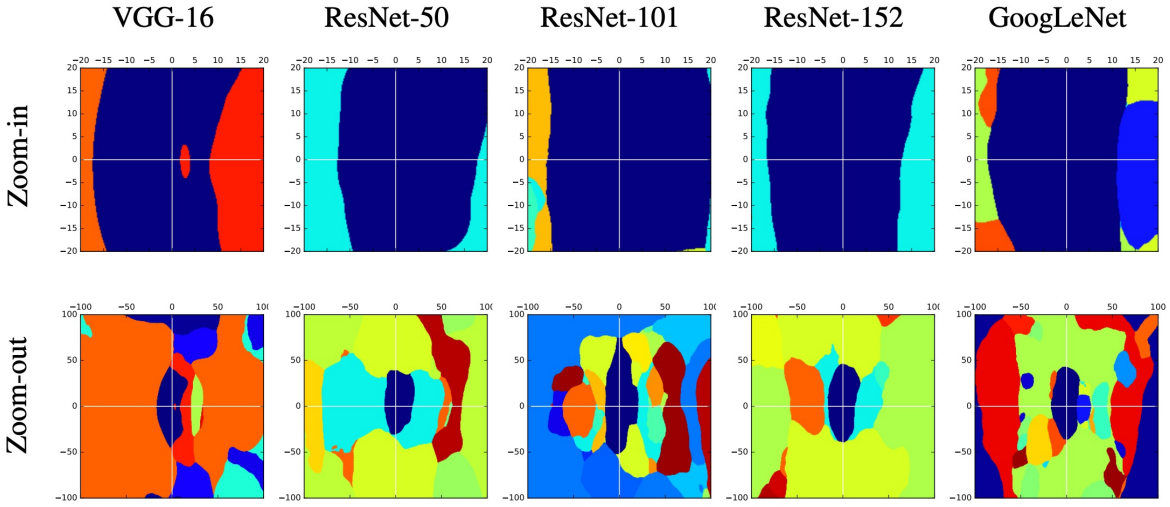
RQ 4: Why Does (Doesn't) It Transfer?

- H1: Gradient directions are not aligned
  - Evaluate
    - Compute the gradients of inputs from the models
    - Compute the cosine similarity between the gradients from two different models
  - Results

	ResNet-152	ResNet-101	ResNet-50	VGG-16	GoogLeNet
ResNet-152	1.00	–	–	–	–
ResNet-101	0.04	1.00	–	–	–
ResNet-50	0.03	0.03	1.00	–	–
VGG-16	0.02	0.02	0.02	1.00	–
GoogLeNet	0.01	0.01	0.01	0.02	1.00

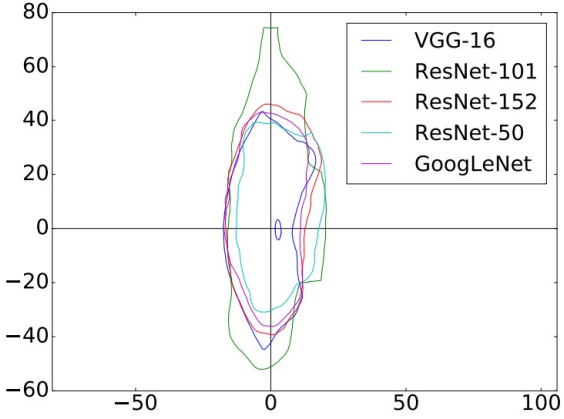
RQ 4: Why Does (Doesn't) It Transfer?

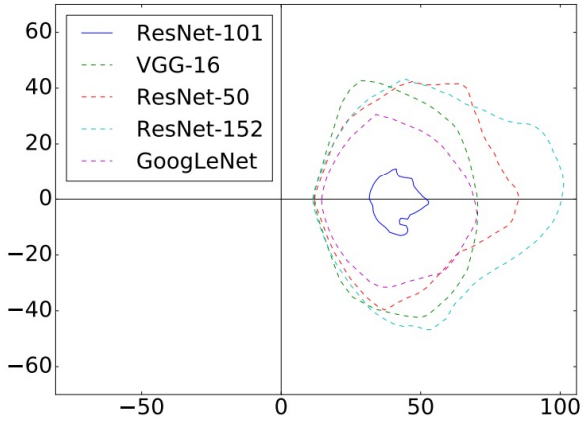
- H2: Transferability May Be Related to Decision Boundary Characteristics
  - Evaluate
    - Take a sample image, and two orthogonal gradient directions
    - Perturb the sample along each direction and measure the labels
  - Results



RQ 4: Why Does (Doesn't) It Transfer?

- H3: H2 May Hold for Ensemble Cases Too
  - Evaluate
    - Take a sample image, and two orthogonal gradient directions
    - Perturb the sample along each direction and measure the labels
  - Results





Conclusion

- Research Questions
  - RQ 1: How much do adversarial examples **transfer** between models?
    - Non-targeted attacks transfer with a high success rate (30 – 90% MR)
    - Targeted attacks transfer much less than non-targeted (0 – 6% MR)
  - RQ 2: (If they don't) How can we **improve** the transferability of AEs?
    - Use the ensemble approach; use more surrogate models
    - Improves MR in both cases
  - RQ 3: How much do AEs transfer in **real-world scenarios**?
    - Non-targeted transfer mostly
    - Targeted attacks transfer up to 18% when the ensemble approach is used
  - RQ 4: Why do they **transfer**?

Recap

- AE←ML
  - Motivation
  - PGD Attack
  - Conclusions (and Implications)
- Transferability
  - Motivation
  - Evaluate Transferability
  - Improve Transferability
  - Connection to Models' Properties
  - Conclusions (and Implications)

Thank You!

Mon/Wed 12:00 – 1:50 pm

Sanghyun Hong

https://secure-ai.systems/courses/MLSec/W22

Oregon State University

SAIL

Secure AI Systems Lab