Adversarial Examples are Not Easily Detected: Bypassing Ten Detection Methods

Nicholas Carlini, David Wagner
University of California, Berkeley
Background
Neural Networks

• I assume knowledge of neural networks ...
• This talk: neural networks for classification
  • Specifically image-based classification
Background: Adversarial Examples

• Given an input $X$ classified as label $T$ ...
• ... it is easy to find an $X'$ close to $X$
• ... so that $F(X') \neq T$
Constructing Adversarial Examples

• Formulation: given input $x$, find $x'$ where
  minimize $d(x, x') + L(x')$
  such that $x'$ is "valid"

• Where $L(x')$ is a loss function minimized when $F(x') \neq T$ and maximized when $F(x') = T$

• Solve via gradient descent
CIFAR-10

Normal       Adversarial

Truck        Airplane
This is decidedly bad
But also: ripe opportunity for research!
Mitigating Evasion Attacks to Deep Neural Networks via Region-based Classification. Xiaoyu Cao, Neil Zhenqiang Gong

APE-GAN: Adversarial Perturbation Elimination with GAN. Shiwei Shen, Guoqing Jin, Ke Gao, Yongdong Zhang

A Learning Approach to Secure Learning. Linh Nguyen, Arunesh Sinha


Ensemble Methods as a Defense to Adversarial Perturbations Against Deep Neural Networks. Thilo Strauss, Markus Hanselmann, Andrej Junginger, Holger Ulmer

MagNet: a Two-Pronged Defense against Adversarial Examples. Dongyu Meng, Hao Chen

CuRTAIL: ChaRacterizing and Thwarting AdversarIal deep Learning. Bita Darvish Rouhani, Mohammad Samragh, Tara Javidi, Farinaz Koushanfar

Efficient Defenses Against Adversarial Attacks. Valentina Zantedeschi, Maria-Irina Nicolae, Ambrish Rawat

Learning Adversary-Resistant Deep Neural Networks. Qinglong Wang, Wenbo Guo, Kaixuan Zhang, Alexander G. Ororbia II, Xinyu Xing, Xue Liu, C. Lee Giles

SafetyNet: Detecting and Rejecting Adversarial Examples Robustly. Jiajun Lu, Theerasit Issaranon, David Forsyth

Enhancing Robustness of Machine Learning Systems via Data Transformations. Arjun Nitin Bhagoji, Daniel Cullina, Bink Sitawarin, Prateek Mittal

Towards Deep Learning Models Resistant to Adversarial Attacks. Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, Adrian Vladu

Towards Robust Deep Neural Networks with BANG. Andras Rozsa, Manuel Gunther, Terrance E. Boult

Deep Variational Information Bottleneck. Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, Kevin Murphy

NO Need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles. Jiajun Lu, Hussein Sibai, Evan Fabry, David Forsyth
Research Question:
Which of these defenses are robust?
Focus of this talk: detection schemes
Normal Classifier

7

Classifier
Normal Classifier
Detector & Classifier
Detector & Classifier

Detector

Classifier
This Talk:

1. How to evaluate a defense
2. Comment on explored directions
Defense #1: PCA-based detection

PCA-based detection

- Hypothesis: Adversarial examples rely on later principle components
  - ... and valid images don't ...
  - ... so let's detect use of high components
It works!
Attack:

Only modify regions of the image that are also used in normal images.
Original

Adversarial (unsecured)

Adversarial (with detector)
Lesson 1: Separate the artifacts of one attack vs intrinsic properties of adversarial examples
Lesson 2:

MNIST is insufficient
CIFAR is better
Defense #2:

Additional Neural Network Detection

Adversarial Training

(7, 7)
(3, 3)
(7, n)
(3, n)
Adversarial Training

(7, y)
(?
, y)
(7, n)
(?, n)
Sounds great.
Sounds great.

But we already know it's easy to fool neural networks ...
... so just construct adversarial examples to

1. be misclassified
2. not be detected
Breaking Adversarial Training

- minimize $d(x, x') + L(x')$
  such that $x'$ is "valid"

- Old: $L(x')$ measures loss of classifier on $x'$
Breaking Adversarial Training

• minimize \( d(x, x') + L(x') + M(x') \)
  such that \( x' \) is "valid"

• Old: \( L(x') \) measures loss of \textbf{classifier} on \( x' \)

• New: \( M(x') \) measures loss of \textbf{detector} on \( x' \)
Original
Adversarial (unsecured)
Adversarial (with detector)
Lesson 3:

Minimize over (compute gradients through) the full defense
Defense #3: Network Randomization

Randomized Classifier
Randomized Classifier
Breaking Randomization

• minimize \( \text{d}(x, x') + L(x') \)
  such that \( x' \) is "valid"

• Old: \( L(x') \) measures loss of network on \( x' \)
Breaking Randomization

- minimize \( d(x, x') + E[L(x')] \)
  such that \( x' \) is "valid"

- Old: \( L(x') \) measures loss of network on \( x' \)

- Now: \( E[L(x')] \) expected loss of network on \( x' \)
Original

Adversarial (unsecured)

Adversarial (with detector)
Original

Adversarial (unsecured)

Adversarial (with detector)
Evaluation Lessons

1. Don't evaluate only on MNIST
2. Minimize over the full defense
3. Use a strong iterative attack
4. Release your source code!

https://nicholas.carlini.com/nn_breaking_detection