Gradient-based Adversarial Attacks to Deep Neural Networks in Limited Access Settings

Yash Sharma
Advisor: Sam Keene

April 15, 2019

Overview

- Extend white-box attacks to limited access settings.
  $\rightarrow$ ZOO: Uses the finite difference method to estimate the gradients for optimization from the output scores. (black-box)
  $\rightarrow$ EAD: Incorporates L_1 minimization to encourage sparsity in the perturbation, hence generating more transferable adversarial examples. (no-box)
- Demonstrate that these attacks can succeed against recently proposed state-of-the-art defenses.

Background

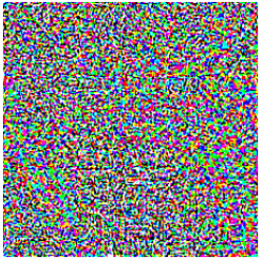
Adversarial Examples

Goodfellow et. al., ICLR 2015

$\boldsymbol{x}$

"panda"
57.7% confidence

$+ .007 \times$



$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"nematode"
8.2% confidence

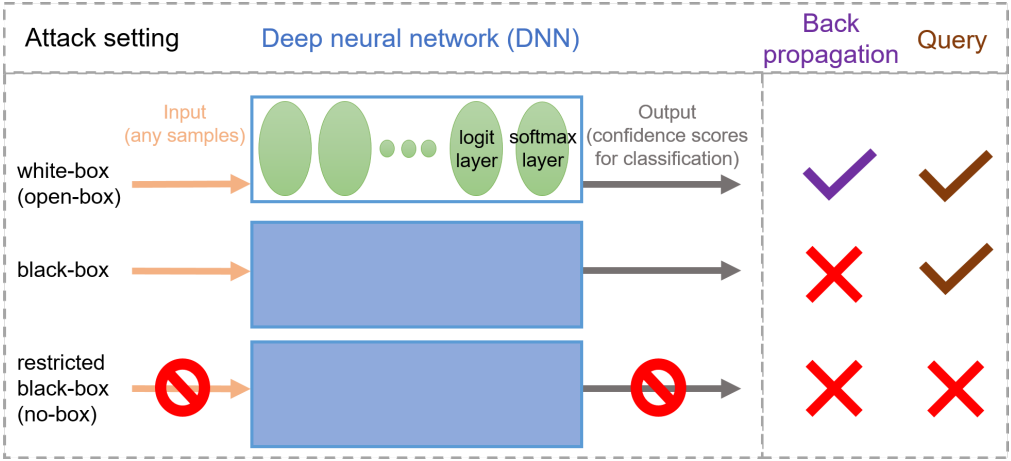
$=$

$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"gibbon"
99.3 % confidence

Attack Settings

Backpropagation computes the gradient of the error function with respect to the neural network weights



Optimization-based Attack

Input image: $\mathbf{x}_0 \in \mathbb{R}^p$, adversarial image: $\mathbf{x} \in \mathbb{R}^p$, target class label: t. Define an optimization problem:

$$\begin{aligned} &\text{minimize}_{\mathbf{x}} \ \|\mathbf{x} - \mathbf{x}_0\|_2^2 + c \cdot f(\mathbf{x}, t) \\ &\text{subject to } \mathbf{x} \in [0,1]^p, \end{aligned} \tag{1}$$

- $\|\mathbf{x} - \mathbf{x}_0\|_2^2$ measures the L_2 distortion
- $f(\mathbf{x}, t)$ is some loss to measure how successful the attack is (smaller is better). How to design it?
- c is a cost constant to trade-off between the two

Carlini & Wagner's (C&W, 2017) Attack

Carlini & Wagner propose to use the following loss:

$$f(\mathbf{x}, t) = \max\{\max_{i \neq t}[Z(\mathbf{x})]_i - [Z(\mathbf{x})]_t, 0\}, \quad (2)$$

$Z(\mathbf{x}) \in \mathbb{R}^K$ is the logit layer outputs (unnormalized probabilities), and the prediction probabilities $F(\mathbf{x})$ are:

$$[F(\mathbf{x})]_k = \frac{\exp([Z(\mathbf{x})]_k)}{\sum_{i=1}^{K} \exp([Z(\mathbf{x})]_i)}, \ \forall\, k \in \{1, \ldots, K\}. \quad (3)$$

- Strongest Attack
- Only works in the white-box case

Black Box: ZOO

Joint work with Pin-Yu Chen (IBM Research), Huan Zhang (UC Davis), Jinfeng Yi (IBM Research), and Cho-Jui Hsieh (UC Davis)

Black Box Attack

Black-box: No access to model parameters;
Can observe model output (prediction probabilities)

Previous Approach

- Transferability based attack using learned substitute model (Papernot et al, 2017)
  $\Rightarrow$ Success rate lower than C&W (model mismatch)
  $\Rightarrow$ Computational cost (substitute model training)

Our Black-box Attack Formulation

Input image: $\mathbf{x}_0$, adversarial image: $\mathbf{x}$, target class label: t. Define the following optimization problem:

$$\begin{aligned} &\text{minimize}_{\mathbf{x}} \ \|\mathbf{x} - \mathbf{x}_0\|_2^2 + c \cdot f(\mathbf{x}, t) \\ &\text{subject to } \mathbf{x} \in [0,1]^p, \end{aligned} \tag{4}$$

We propose to use the following loss function:

$$f(\mathbf{x}, t) = \max\{\max_{i \neq t} \log[F(\mathbf{x})]_i - \log[F(\mathbf{x})]_t, 0\}, \tag{5}$$

where $F(\mathbf{x}) \in \mathbb{R}^K$ is the blackbox output (probabilities)

Zeroth Order Optimization (ZOO)

Access to $f(\mathbf{x})$ only, no $\nabla f(\mathbf{x})$ available.
Estimate gradient $\hat{g}_i$ for *each pixel* using the symmetric difference quotient:

$$\hat{g}_i := \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_i} \approx \frac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x} - h\mathbf{e}_i)}{2h}, \tag{6}$$

Then we update each pixel (coordinate) based on its estimated gradient (we use ADAM optimizer).

Challenges of ZOO

Number of Queries = $O(2 \cdot \text{number of pixels})$

For an ImageNet image with resolution $299 \times 299 \times 3$, we need $536,406$ queries to estimate the gradients of all pixels once.

How to reduce the number of queries?

Black-box attack by Coordinate Descent

$\mathbf{x} \in \mathbb{R}^p$ is the input image with p pixels, f is the loss function we defined to find adversarial examples

Algorithm 1 Stochastic Coordinate Descent

1: **while** not converged **do**
2: Randomly pick a coordinate $i \in \{1, \ldots, p\}$
3: Compute an update δ^* by approximately minimizing

$$\arg\min_{\delta} f(\mathbf{x} + \delta \mathbf{e}_i)$$

4: Update $\mathbf{x}_i \leftarrow \mathbf{x}_i + \delta^*$
5: **end while**

In practice we optimize a batch of $B = 128$ coordinates for better efficiency

ZOO-ADAM

Algorithm 2 ZOO-ADAM: Zeroth Order Stochastic Coordinate Descent with Coordinate-wise ADAM

Require: Step size η, ADAM states $M \in \mathbb{R}^p, v \in \mathbb{R}^p, T \in \mathbb{Z}^p$, ADAM hyper-parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$

$M \leftarrow \mathbf{0}, v \leftarrow \mathbf{0}, T \leftarrow \mathbf{0}$

while not converged **do**

  Randomly pick a coordinate $i \in \{1, \cdots, p\}$

  Estimate $\hat{g}_i$ using (6)

  $T_i \leftarrow T_i + 1$

  $M_i \leftarrow \beta_1 M_i + (1 - \beta_1)\hat{g}_i, \quad v_i \leftarrow \beta_2 v_i + (1 - \beta_2)\hat{g}_i^2$

  $\hat{M}_i = M_i/(1 - \beta_1^{T_i}), \quad \hat{v}_i = v_i/(1 - \beta_2^{T_i})$

  $\delta^* = -\eta \dfrac{\hat{M}_i}{\sqrt{\hat{v}_i} + \epsilon}$

  Update $\mathbf{x}_i \leftarrow \mathbf{x}_i + \delta^*$

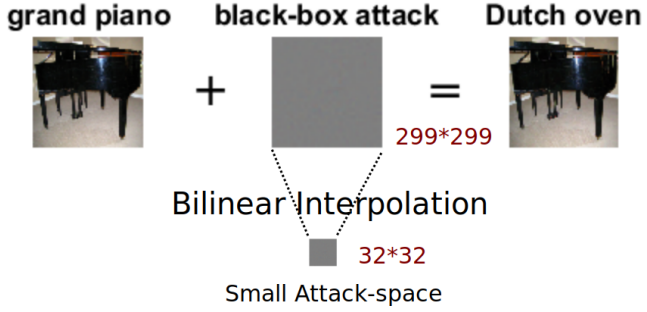
end while

Attack-space Dimension Reduction

- *Attack-space* is the image space that we search for adversarial noise.
- Instead of searching in the original image's space, we can search in a smaller space (with less pixels) using dimension reduction techniques.
- This greatly reduces the number of pixels to optimize and make the attack practical for large images.

Attack-space Dimension Reduction

For images, size scaling is easy and fast. We craft noise at small size and then upscale it to the input image size. Input image is untouched.

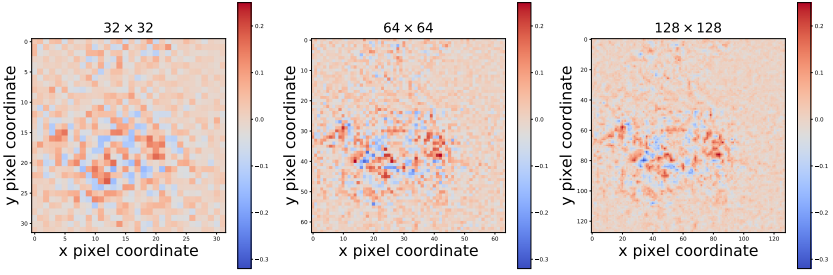


But what if 32×32 is not big enough?

Hierarchical Attack (on bagel)

Gradually increase the dimension of attack space after some iterations.

$32 \times 32 \rightarrow 64 \times 64 \rightarrow 128 \times 128$



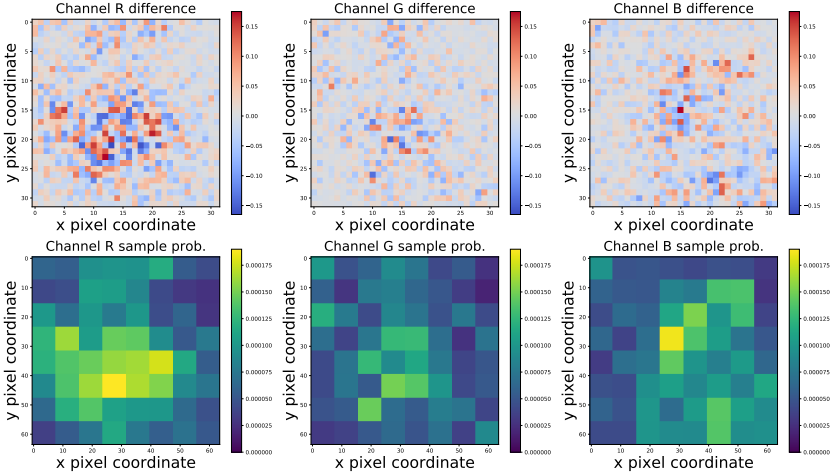
bagel

Hierarchical attack

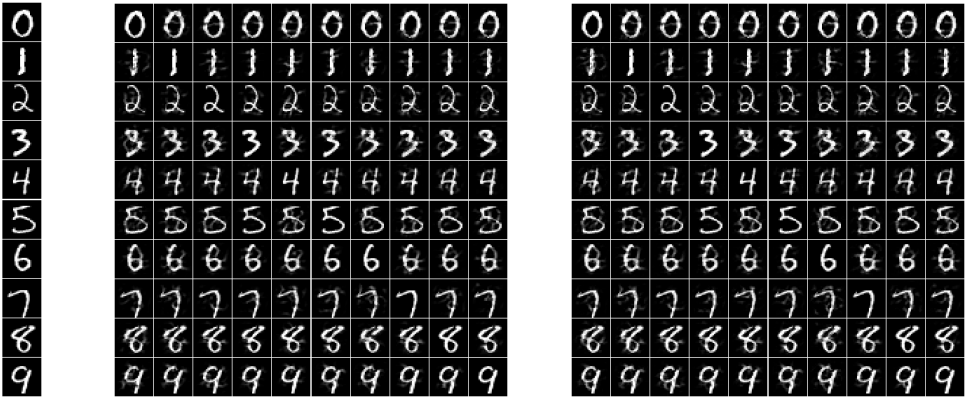
Most changed pixels are around the center of bagel?

Importance Sampling (on bagel)

Importance determined by the magnitude of changes in a certain region (we use maxpooling).



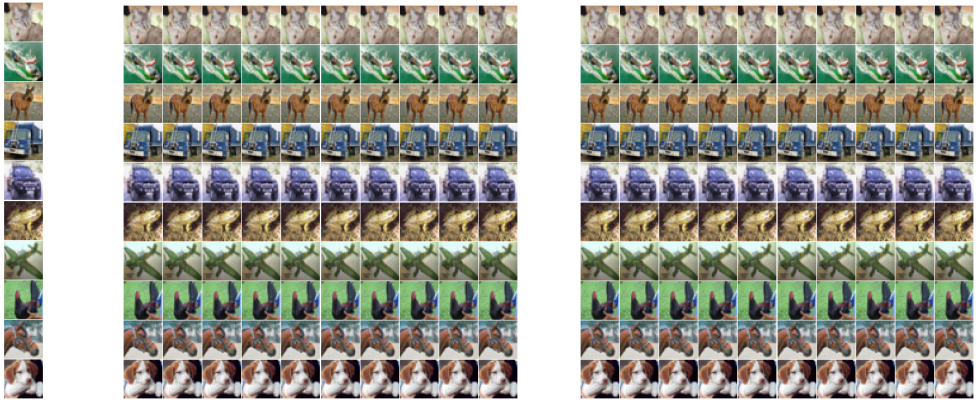
Targeted Attack on MNIST



White-box C&W Black-box ZOO-ADAM

Figure: Row: crafted adversarial examples from original examples in (a). Column: targeted attack class (‘0’ to ‘9’).

Targeted Attack on CIFAR-10



White-box C&W Black-box ZOO-ADAM

Figure: Row: crafted adversarial examples from original examples in (a). Column: targeted attack class.

Attack on MNIST & CIFAR-10

Success rate close to white-box (C & W) attack - nearly 100%. Similar L_2 distortion and reasonable attack time.

	MNIST					
	Untargeted			Targeted		
	Success Rate	Avg. L_2	Avg. Time (per attack)	Success Rate	Avg. L_2	Avg. Time (per attack)
White-box (C&W)	100 %	1.48066	0.48 min	100 %	2.00661	0.53 min
Substitute Model + FGSM	40.6 %	-	0.002 sec (+ 6.16 min)	7.48 %	-	0.002 sec (+ 6.16 min)
Substitute Model + C&W	33.3 %	3.6111	0.76 min (+ 6.16 min)	26.74 %	5.272	0.80 min (+ 6.16 min)
ZOO-ADAM	**100** %	1.49550	1.38 min	**98.9** %	1.987068	1.62 min
	CIFAR-10					
	Untargeted			Targeted		
	Success Rate	Avg. L_2	Avg. Time (per attack)	Success Rate	Avg. L_2	Avg. Time (per attack)
White-box (C&W)	100 %	0.17980	0.20 min	100 %	0.37974	0.16 min
Substitute Model + FGSM	76.1 %	-	0.005 sec (+ 7.81 min)	11.48 %	-	0.005 sec (+ 7.81 min)
Substitute Model + C&W	25.3 %	2.9708	0.47 min (+ 7.81 min)	5.3 %	5.7439	0.49 min (+ 7.81 min)
ZOO-ADAM	**100** %	0.19973	3.43 min	**96.8** %	0.39879	3.95 min

Untargeted Attack on Inception-v3

- black-box attacks to 150 ImageNet test images (size $299 \times 299 \times 3$)
- $2,000$ iterations (within 20 minutes) for each attack
- reduced attack-space: $32 \times 32 \times 3$
- No hierarchical attack or importance sampling

	Success Rate	Avg. L_2
White-box (C&W)	100 %	0.37310
Black-box (**ZOO-ADAM**)	88.9 %	1.19916

Untargeted Attack on Inception-v3

Figure: ImageNet untargeted attack examples

Targeted Attack on Inception-v3

Targeted attack is much harder than untargeted attack, because we want to force the image to be misclassified to specifically one class out of 1,000.

Before Attack: $P(\text{bagel}) = 0.97$, $P(\text{piano}) = 0.000006$
After Attack: $P(\text{bagel}) = 0.006$, $P(\text{piano}) = 0.0061$
L_2 distortion: 3.425

Targeted Attack on Inception-v3

Needs 20,000 iterations to perform this *hard* targeted attack (about 4 hours). Attack-space dimension reduction, hierarchical attack and importance sampling techniques applied.

Targeted Attack on Inception-v3

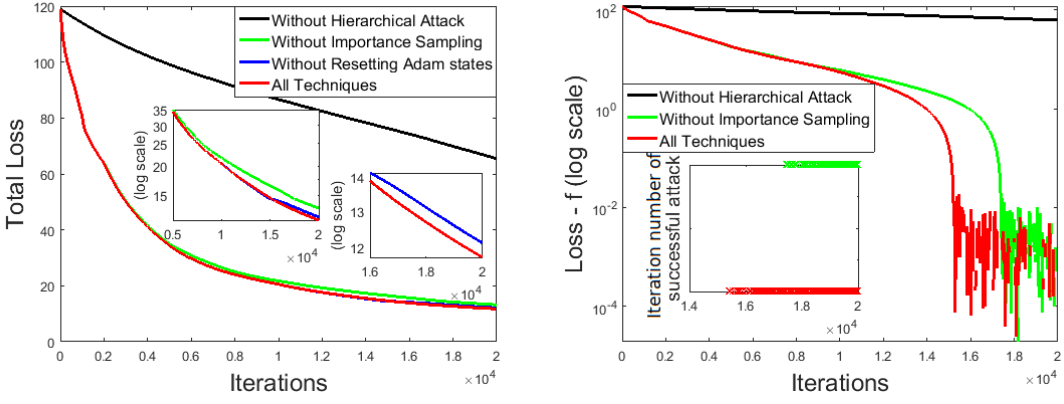


Figure: Left: the total loss $\|\mathbf{x} - \mathbf{x}_0\|_2^2 + c \cdot f(\mathbf{x}, t)$ versus iterations. Right: $c \cdot f(\mathbf{x}, t)$ versus iterations in log scale. When $c \cdot f(\mathbf{x}, t)$ reaches 0, a valid attack is found.

Targeted Attack on Inception-v3

40 images from ImageNet test set, random target:

- 30.0% success within 2,000 iterations
- 72.5% success within 5,000 iterations
- 82.5% success rate within 10,000 iterations
- 95.0% success rate within 20,000 iterations
- Average L_2 distortion: 2.108

Conclusions

1. Zeroth Order Optimization (ZOO) based black-box attacks to deep neural networks can be applied to large images by using the proposed *attack-space dimension reduction*, *hierarchical attack* and *importance sampling* techniques.
2. ZOO can achieve a success rate similar to white-box attacks, without relying on transferability or training an extra substitute model.

No Box: EAD

Joint work with Pin-Yu Chen (IBM Research), Huan Zhang (UC Davis), Jinfeng Yi (IBM Research), and Cho-Jui Hsieh (UC Davis)

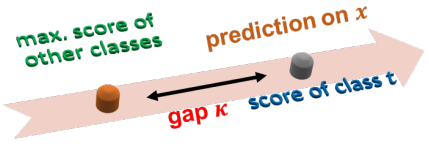
Carlini & Wagner's (C&W, 2017) Attack

Targeted attack formulation:

$$\begin{aligned} &\text{minimize}_{\mathbf{x}} \ \|\mathbf{x} - \mathbf{x}_0\|_2^2 + c \cdot f(\mathbf{x}, t) \\ &\text{subject to } \mathbf{x} \in [0,1]^p, \end{aligned} \tag{7}$$

C&W loss function:

$$f(\mathbf{x}, t) = \max\{\max_{i \neq t}[Z(\mathbf{x})]_i - [Z(\mathbf{x})]_t, -\kappa\}, \tag{8}$$



Elastic-net Optimization

- Elastic-net: $min_{\mathbf{z}}\ f(\mathbf{z}) + \lambda_1 \|\mathbf{z}\|_1 + \lambda_2 \|\mathbf{z}\|_2^2$
  $\Rightarrow$ Group feature selection for high-dimensional machine learning problems
- C&W: $min_{\mathbf{x}}\ \|\mathbf{x} - \mathbf{x}_0\|_2^2 + c \cdot f(\mathbf{x}, t)$
  $\Rightarrow$ Elastic-net: $\lambda_1 = 0$, $\lambda_2 = \frac{1}{c}$
- Why L_1?
  $\Rightarrow$ Convex regularizer that encourages sparsity in the perturbation
- Goal: Craft robust adversarial examples by limiting unnecessary noise in the perturbation

EAD Algorithm

Formulation:

$$\begin{aligned} &\text{minimize}_{\mathbf{x}}\ c \cdot f(\mathbf{x}, t) + \|\mathbf{x} - \mathbf{x}_0\|_2^2 + \beta\|\mathbf{x} - \mathbf{x}_0\|_1 \\ &\text{subject to } \mathbf{x} \in [0,1]^p \end{aligned} \tag{9}$$

Solution: Iterative Soft Thresholding Algorithm (ISTA)

$$[S_\beta(\mathbf{z})]_i = \begin{cases} \min\{\mathbf{z}_i - \beta, 1\}, & \text{if } \mathbf{z}_i - \mathbf{x}_{0i} > \beta; \\ \mathbf{x}_{0i}, & \text{if } |\mathbf{z}_i - \mathbf{x}_{0i}| \leq \beta; \\ \max\{\mathbf{z}_i + \beta, 0\}, & \text{if } \mathbf{z}_i - \mathbf{x}_{0i} < -\beta, \end{cases} \tag{10}$$

Interpretation: General and Robust

EAD-ISTA

Algorithm 3 Elastic-Net Attacks to DNNs (EAD)

1: **Input:** original labeled image $(\mathbf{x}_0, t_0)$, target attack class t, attack transferability parameter κ, L_1 regularization parameter β, step size α_k, # of iterations I
2: **Output:** adversarial example $\mathbf{x}$
3: Let $g(x) = c \cdot f(x, t) + \|\mathbf{x} - \mathbf{x}_0\|_2^2$
4: Initialization: $\mathbf{x}^{(0)} = \mathbf{y}^{(0)} = \mathbf{x}_0$
5: **for** $k = 0$ to $I - 1$ **do**
6: $\quad \mathbf{x}^{(k+1)} = S_\beta(\mathbf{y}^{(k)} - \alpha_k \nabla g(\mathbf{y}^{(k)}))$
7: $\quad \mathbf{y}^{(k+1)} = \mathbf{x}^{(k+1)} + \frac{k}{k+3}(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})$
8: **end for**
9: Decision rule: determine $\mathbf{x}$ from successful examples in $\{\mathbf{x}^{(k)}\}_{k=1}^{I}$ (EN rule or L_1 rule).

Adversarial Examples

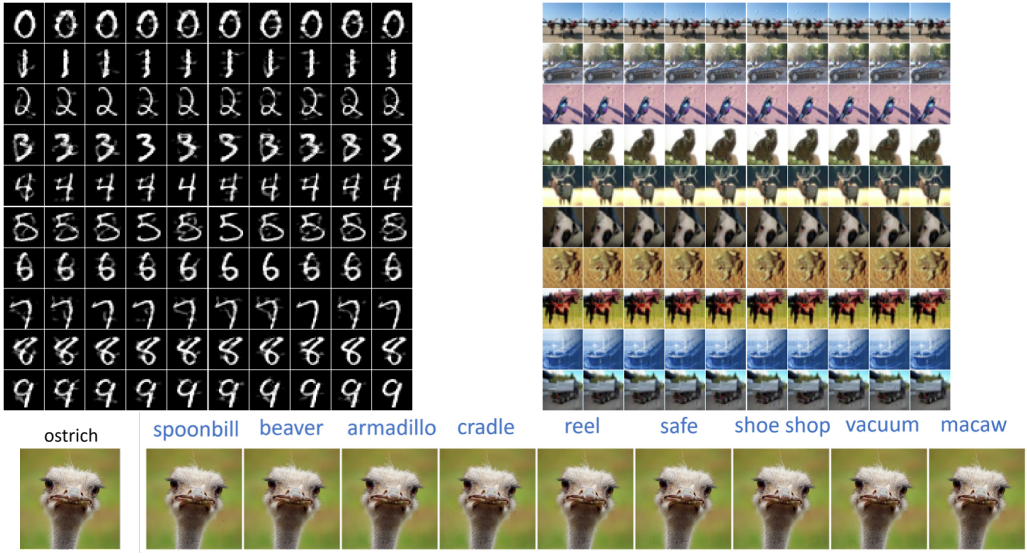


Figure: MNIST, CIFAR-10, ImageNet

Performance (Targeted)

EAD attains 100% ASR and the least L_1 distorted adversarial examples.

	MNIST				CIFAR10				ImageNet			
Attack method	ASR	L_1	L_2	L_∞	ASR	L_1	L_2	L_∞	ASR	L_1	L_2	L_∞
C&W (L_2)	**100**	22.46	**1.972**	0.514	**100**	13.62	**0.392**	0.044	**100**	232.2	**0.705**	0.03
FGM-L_1	39	53.5	4.186	0.782	48.8	51.97	1.48	0.152	1	61	0.187	0.007
FGM-L_2	34.6	39.15	3.284	0.747	42.8	39.5	1.157	0.136	1	2338	6.823	0.25
FGM-L_∞	42.5	127.2	6.09	0.296	52.3	127.81	2.373	0.047	3	3655	7.102	0.014
I-FGM-L_1	**100**	32.94	2.606	0.591	**100**	17.53	0.502	0.055	77	526.4	1.609	0.054
I-FGM-L_2	**100**	30.32	2.41	0.561	**100**	17.12	0.489	0.054	**100**	774.1	2.358	0.086
I-FGM-L_∞	**100**	71.39	3.472	**0.227**	**100**	33.3	0.68	**0.018**	**100**	864.2	2.079	**0.01**
EAD (EN rule)	**100**	**17.4**	2.001	0.594	**100**	**8.18**	0.502	0.097	**100**	**69.47**	1.563	0.238
EAD (L_1 rule)	**100**	**14.11**	2.211	0.768	**100**	**6.066**	0.613	0.17	**100**	**40.9**	1.598	0.293

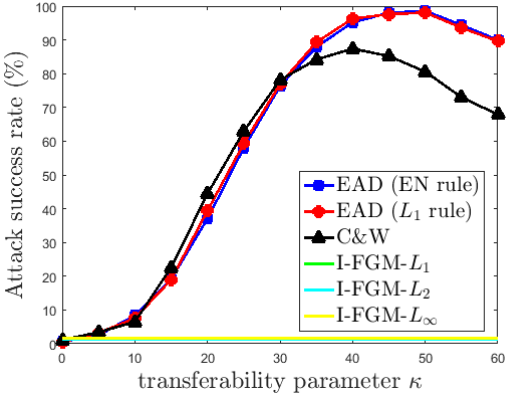
Adversarial Training (MNIST)

Incorporating L_1 examples complements adversarial training and enhances attack difficulty in terms of distortion.

Attack method	Adversarial training	Average case			
		ASR	L_1	L_2	L_∞
C&W (L_2)	None	100	22.46	1.972	0.514
	EAD	100	26.11	2.468	0.643
	C&W	100	24.97	2.47	**0.684**
	EAD + C&W	100	**27.32**	**2.513**	0.653
EAD (L_1 rule)	None	100	14.11	2.211	0.768
	EAD	100	**17.04**	2.653	0.86
	C&W	100	15.49	2.628	**0.892**
	EAD + C&W	100	16.83	**2.66**	0.87

Attack Transferability (MNIST)

Transfer Attack from undefended network to defensively distilled network



Results Against Defenses

Joint work with Pin-Yu Chen (IBM Research)

Ensemble Adversarial Training (ZOO)

- Augment training data with with perturbations transferred from other models.
  $\rightarrow$ State-of-the-art ImageNet defense
  $\rightarrow$ Top-performing model in NIPS 2017 competition
- Perform non-targeted attack with ZOO on defended Inception-v3 and Inception ResNet-v2
  $\rightarrow$ Achieve 100% success rate on 10 random samples against both models
  $\rightarrow$ Visually imperceptible perturbations

Madry Defense Model (EAD)

- A high capacity network trained against PGD, iterative FGSM with random starts.
  $\rightarrow$ State-of-the-art MNIST defense
- Competition: Provided undefended models of the same architecture.
  $\rightarrow$ Transfer to hidden defended model
  $\rightarrow$ Used EAD (EN Rule) with ensemble of 3 models.

Results

EAD yields near 100% ASR in both the targeted and non-targeted cases.

		Targeted				Non-Targeted			
Attack Method	Confidence	ASR	L_1	L_2	L_∞	ASR	L_1	L_2	L_∞
PGD	None	68.5	188.3	8.947	0.6	99.9	270.5	13.27	0.8
I-FGM	None	75.1	144.5	7.406	0.915	99.8	199.4	10.66	0.9
C&W	10	1.1	34.15	2.482	0.548	4.9	23.23	1.702	0.424
	30	69.4	68.14	4.864	0.871	71.3	51.04	3.698	0.756
	50	92.9	117.45	8.041	0.987	99.1	78.65	5.598	0.937
	70	34.8	169.7	10.88	0.994	99	119.4	8.097	0.99
EAD	10	27.4	25.79	3.209	0.876	39.9	19.19	2.636	0.8
	30	85.8	49.64	5.179	0.995	94.5	34.28	4.192	0.971
	50	98.5	93.46	7.711	1	99.6	57.68	5.839	0.999
	70	67.2	148.9	10.36	1	99.8	90.84	7.719	1

Adversarial Examples (Non-Targeted)

Performing elastic-net minimization aids in minimizing visual distortion, even when the L_∞ distortion is large.

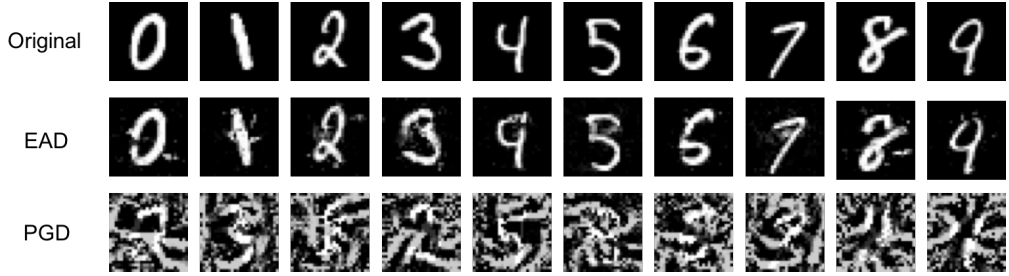


Figure: Visual illustration of adversarial examples crafted in the non-targeted case by EAD and PGD with similar average L_∞ distortion (0.8).

Feature Squeezing (EAD)

- Relies on applying input transformations to reduce the degrees of freedom available to an adversary.
  $\rightarrow$ Reduce the color bit-depth of images.
  $\rightarrow$ Using smoothing (both local and non-local).
- Detection: the model's original and squeezed predictions are compared using the L_1 norm.
  $\rightarrow$ Multiple feature squeezers are combined by outputting the maximum distance.
  $\rightarrow$ Threshold chosen which is exceeded by no more than 5% of legitimate samples.

Results (MNIST)

EAD yields near 100% ASR in both the targeted and non-targeted cases.

		Non-Targeted				Targeted							
						Next				LL			
Attack Method	Confidence	ASR	L_1	L_2	L_∞	ASR	L_1	L_2	L_∞	ASR	L_1	L_2	L_∞
I-FGSM	None	100%	196.0	10.17	0.900	78%	169.8	8.225	0.881	67%	188.1	9.091	0.991
C&W	10	0%	21.05	1.962	0.568	0%	31.94	2.748	0.655	0%	37.78	3.207	0.732
	20	15%	27,21	2.472	0.665	10%	40.51	3.419	0.763	24%	47.86	3.977	0.820
	30	64%	34.30	3.019	0.754	67%	47.43	3.973	0.842	91%	59.56	4.811	0.888
	40	87%	42.04	3.590	0.831	97%	61.12	4.938	0.922	100%	72.88	5.715	0.939
EAD	10	24%	11.44	2.286	0.879	7%	19.69	3.114	0.942	7%	23.99	3.481	0.955
	20	80%	15.26	2.766	0.921	65%	26.80	3.752	0.964	78%	31.81	4.122	0.972
	30	95%	20.17	3.264	0.957	97%	35.50	4.449	0.983	93%	39.68	4.769	0.991
	40	97%	26.50	3.803	0.972	100%	44.75	5.114	0.992	100%	50.21	5.532	0.997

Results (CIFAR-10)

EAD yields near 100% ASR in both the targeted and non-targeted cases.

		Non-Targeted				Targeted							
						Next				LL			
Attack Method	Confidence	ASR	L_1	L_2	L_∞	ASR	L_1	L_2	L_∞	ASR	L_1	L_2	L_∞
I-FGSM	None	100%	81.18	1.833	0.070	100%	212.0	4.979	0.299	100%	214.9	5.042	0.300
C&W	10	32%	10.51	0.274	0.033	0%	14.25	0.368	0.042	0%	17.36	0.445	0.049
	30	78%	28.80	0.712	0.073	51%	37.11	0.901	0.083	6%	41.51	1.006	0.093
	50	96%	59.32	1.416	0.130	98%	82.54	1.954	0.169	94%	90.17	2.129	0.179
	70	100%	120.2	2.827	0.243	100%	201.2	4.713	0.375	100%	212.2	4.962	0.403
EAD	10	46%	6.371	0.379	0.079	10%	8.187	0.508	0.109	0%	10.17	0.597	0.121
	30	78%	18.94	0.876	0.146	51%	25.98	1.090	0.166	23%	29.58	1.209	0.175
	50	94%	42.36	1.550	0.206	96%	62.90	2.094	0.247	90%	70.23	2.296	0.275
	70	100%	83.14	2.670	0.317	100%	157.9	4.466	0.477	100%	172.8	4.811	0.502

Adversarial Examples (Non-Targeted)

Figure: First row: Original, Subsequent rows: $\kappa = \{10, 20, 30\}$.

Figure: First row: Original, Subsequent rows: $\kappa = \{10, 30, 50\}$.

Conclusion

Summary

- Validated effectiveness of ZOO as the state-of-the-art black-box attack.
- Validated effectiveness of EAD as the state-of-the-art no-box attack.
- Demonstrated attacks can succeed against state-of-the-art defenses.
  $\rightarrow$ Ensemble Adversarial Training: ZOO
  $\rightarrow$ Madry Defense Model: EAD
  $\rightarrow$ Feature Squeezing: EAD

Future Work

- Explore gradient-free optimization strategies, like Genetic Algorithms.
  $\rightarrow$ Estimating the gradient is costly (ZOO)
- Extend black-box attack to real-world partial information settings.
  $\rightarrow$ Top-N classes outputted
- Extend algorithms to other domains.
  $\rightarrow$ Text + Speech

Thank you!