Towards ML You Can Rely On

Aleksander Mądry

@aleks_madry

madry-lab.ml

Machine Learning: The Success Story?

Image classification

Reinforcement Learning

IS "DEEP LEARNING" A REVOLUTION IN ARTIFICIAL INTELLIGENCE?

Andrew Ng
@AndrewYNg
Follow

"AI is the new electricity!" Electricity transformed countless industries; AI will now do the same.

2016: The Year That Deep Learning Took Over t

WHY DEEP LEARNING IS SUDDENLY CHANGING YOUR LIFE

Input sentence:	Translation (PBMT):	Translation (GNMT):	Translation (human):
李克強此行將啟動中加總理年度對話機制，與加拿大總理杜魯多舉行兩國總理首次年度對話。	Li Keqiang premier added this line to start the annual dialogue mechanism with the Canadian Prime Minister Trudeau two prime ministers held its first annual session.	Li Keqiang will start the annual dialogue mechanism with Prime Minister Trudeau of Canada and hold the first annual dialogue between the two premiers.	Li Keqiang will initiate the annual dialogue mechanism between premiers of China and Canada during this visit, and hold the first annual dialogue with Premier Trudeau of Canada.

Machine translation

Machine Learning: The Success Story?

IS "DEEP LEARNING" A REVOLUTION IN ARTIFICIAL INTELLIGENCE?

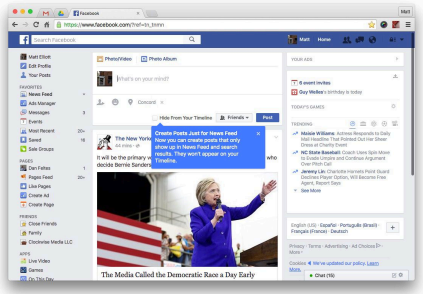
Andrew Ng
@AndrewYNg
Follow

"AI is the new electricity!" Electricity transformed countless industries; AI will now do the same.

2016: The Year That Deep Learning Took Over t

WHY DEEP LEARNING IS SUDDENLY CHANGING YOUR LIFE

Things are great,
so what's the problem?



Is our ML truly ready for deployment?

Overarching questions:

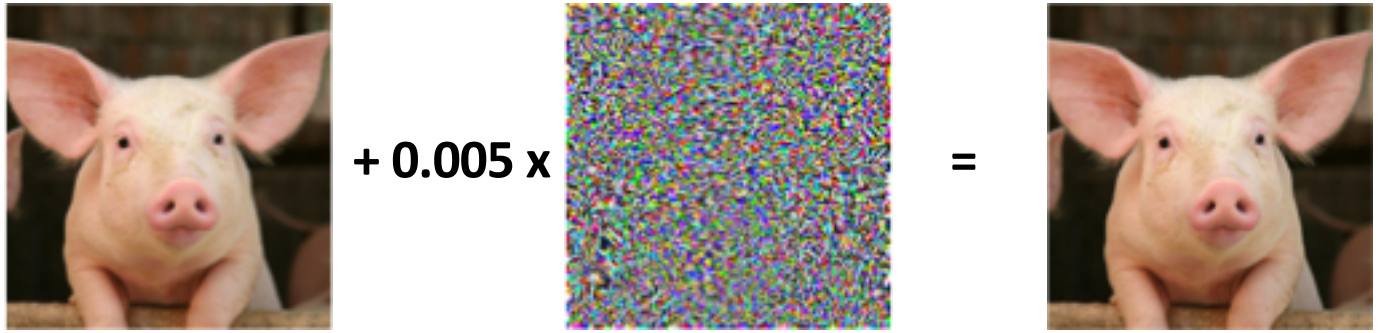
→ Do we **really** understand how/why/**if** our ML tools work?

→ Can we truly rely on these tools? **Today**

Can We Truly Rely on ML?

Adversarial Examples

“pig” (91%) **“airliner” (99%)**



[Goodfellow et al. 2014]: Imperceptible noise can fool state-of-the-art classifiers

“revolver” **“mouse trap”**

[Engstrom Tran Tsipras Schmidt **M** 2018]:
Rotation + Translation Suffices

[Athalye Engstrom Ilyas Kwok 2017]:
3D-printed **turtle** model classified as **rifle** from most viewpoints

Should we be worried?

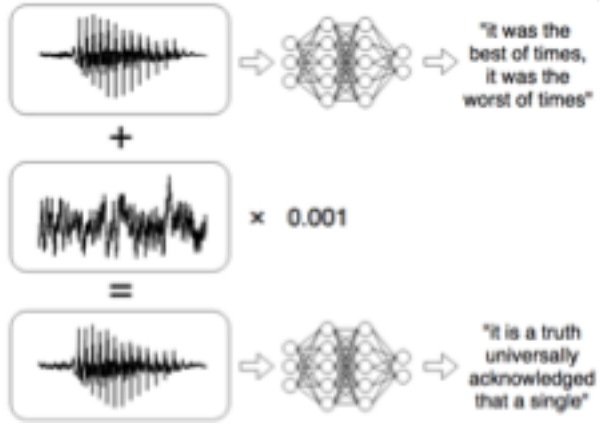
Why Is This Brittleness of ML a Problem?

→ Security

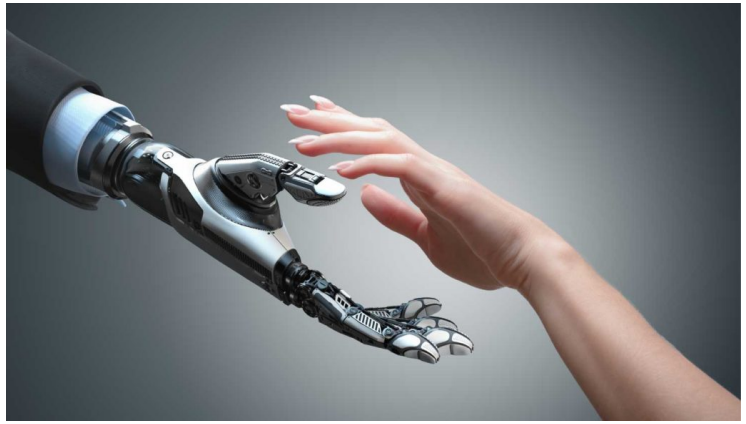
→ Safety

→ ML Alignment

[Sharif et al. 2016]:
Glasses that fool face recognition

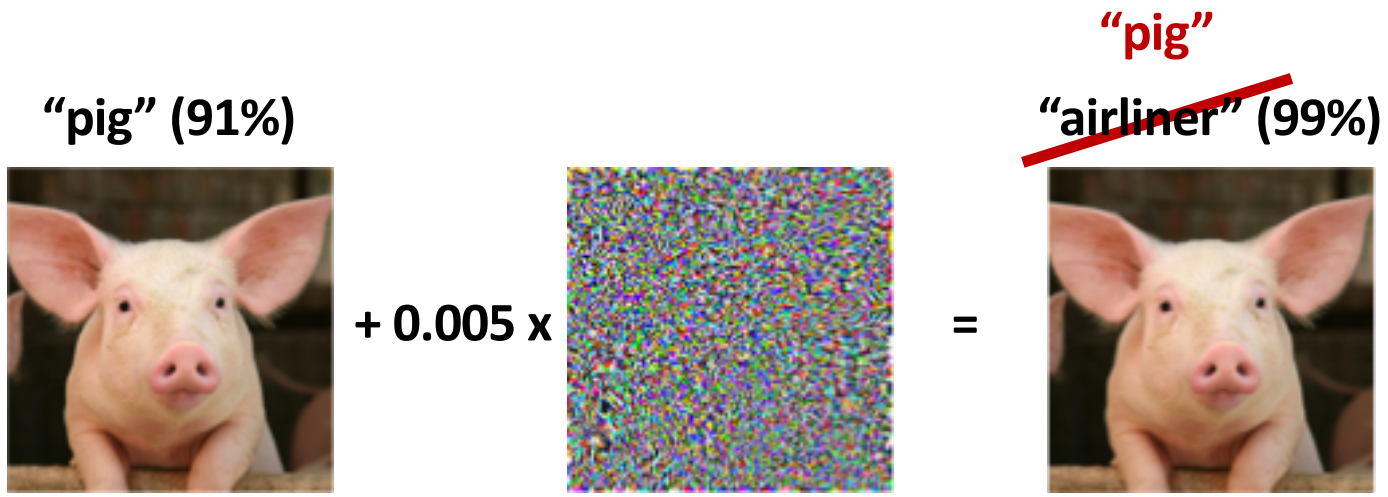


[Carlini Wagner 2018]:
Voice commands that are unintelligible to humans



Need to understand the "failure modes" of ML

Towards Adversarially Robust Models

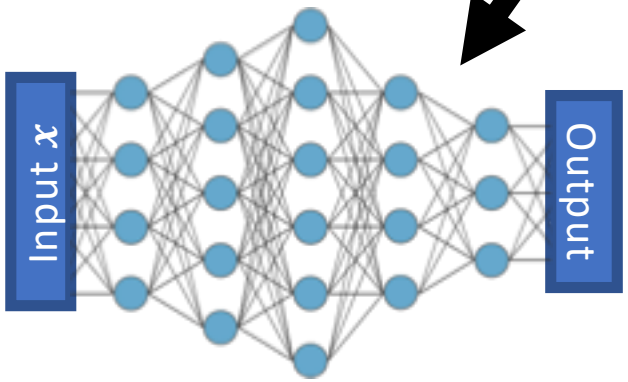


Where Do Adversarial Examples Come From?

Model Parameters Input Correct Label

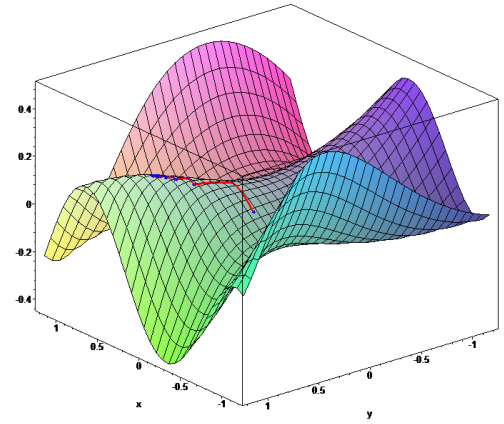
Goal of training: $min_\theta \; loss(\theta, x, y)$

Differentiable



To get an adv. example: $max_\delta \; loss(\theta, x + \delta, y)$

Can use gradient descent method to find good θ

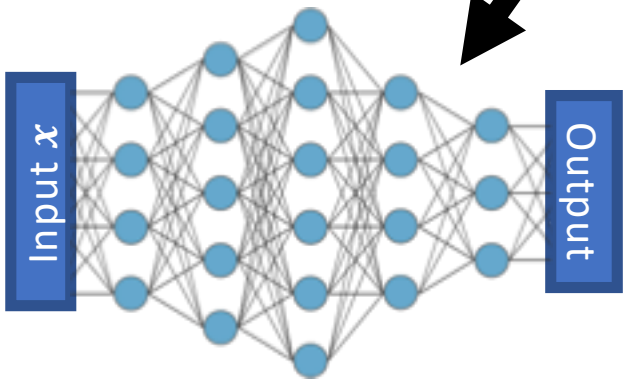


Where Do Adversarial Examples Come From?

Differentiable

Model Parameters Input Correct Label

Goal of training: $min_\theta \; loss(\theta, x, y)$



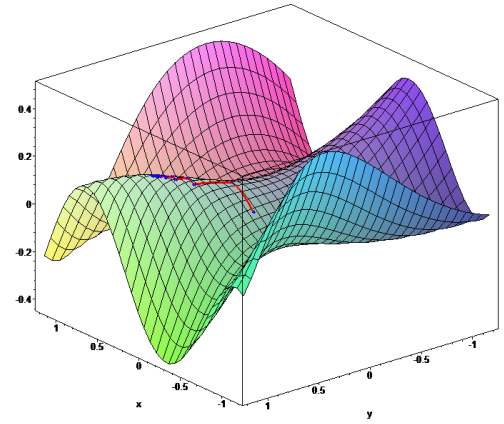
To get an adv. example: $max_\delta \; loss(\theta, x + \delta, y)$

Which δ are allowed?

Any δ that is small wrt

- ℓ_p-norm
- Rotation and/or translation
- VGG feature perturbation
- …

Can use gradient descent method to find bad δ



Towards ML Models that Are Adv. Robust

[M Makelov Schmidt Tsipras Vladu 2018]

Key observation: Lack of adv. robustness is **NOT** at odds with what we currently want our ML models to achieve

Standard generalization: $\mathbb{E}_{(x,y)\sim D}\left[loss(\theta, x, y)\right]$

But: Adversarial noise is of ”measure zero”

Need: Adv. robust generalization:

This is a **robustness** guarantee

$$\mathbb{E}_{(x,y)\sim D}\left[\max_{\boldsymbol{\delta}\in\boldsymbol{\Delta}} loss(\theta, x+\boldsymbol{\delta}, y)\right]$$

Towards ML Models that Are Adv. Robust

[M Makelov Schmidt Tsipras Vladu 2018]

Resulting training primitive:

$$\min_{\theta} \max_{\delta \in \Delta} loss(\theta, x + \boldsymbol{\delta}, y)$$

Finding a robust model

Finding an "attack"

To improve the model: Train on **good** attacks
(aka as "adversarial training" **[Goodfellow Shlens Szegedy '15]**)

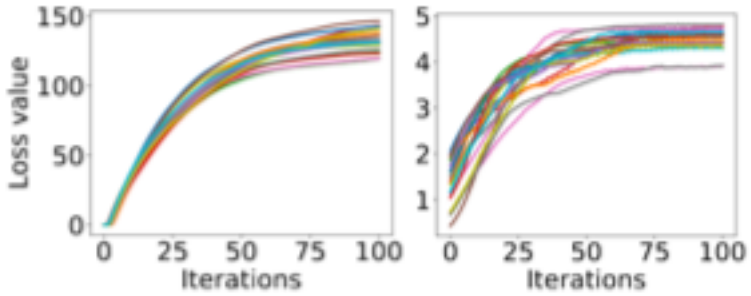
Does this work?

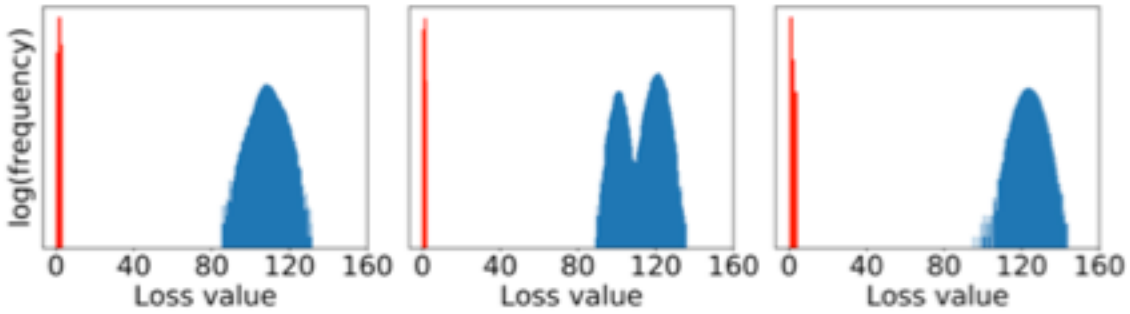
Yes! (In practice)
But certain care is required

Key Components

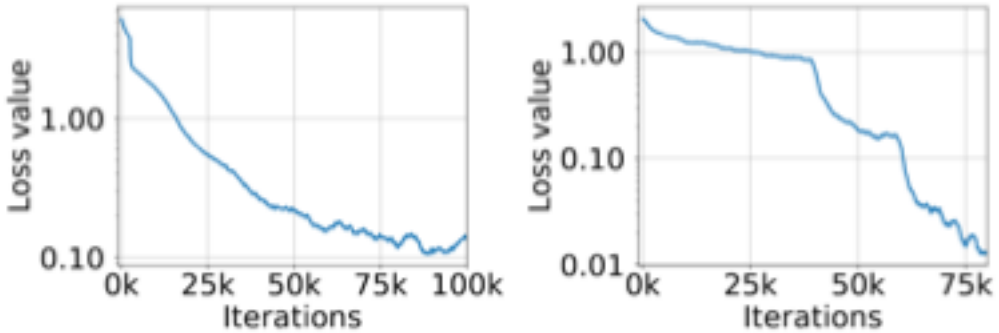
→ **Strong**, **reliable** attacks

→ Sufficient model capacity

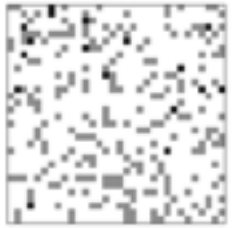
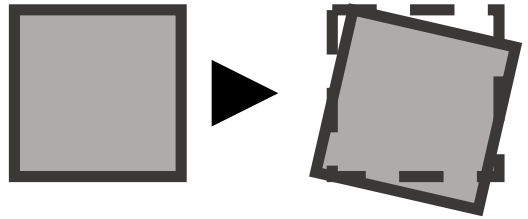




Result: Robustness increases steadily



(a) MNIST (b) CIFAR10

	ℓ_∞-norm	ℓ_2-norm	Rotation + Translation
			
MNIST 	$\epsilon = 0.3/1$	$\epsilon = 2.5/1$	$\epsilon = \pm 3\ px, \pm 30°$
	89%	66%	98%
CIFAR-10 	$\epsilon = 8/255$	$\epsilon = 80/255$	$\epsilon = \pm 3\ px, \pm 30°$
	47%	69%	71% (+vote 82%)**
ImageNet 	$\epsilon = 16/255$		$\epsilon = \pm 30\ px, \pm 30°$
	4% (+ALP 28%)*	-	53% (+vote 57%)**

*[Kannan et al. 2018]

**[Engstrom et al. 2018]

ML via Adversarial Robustness Lens

How does adv. robust ML differ from “standard” ML?

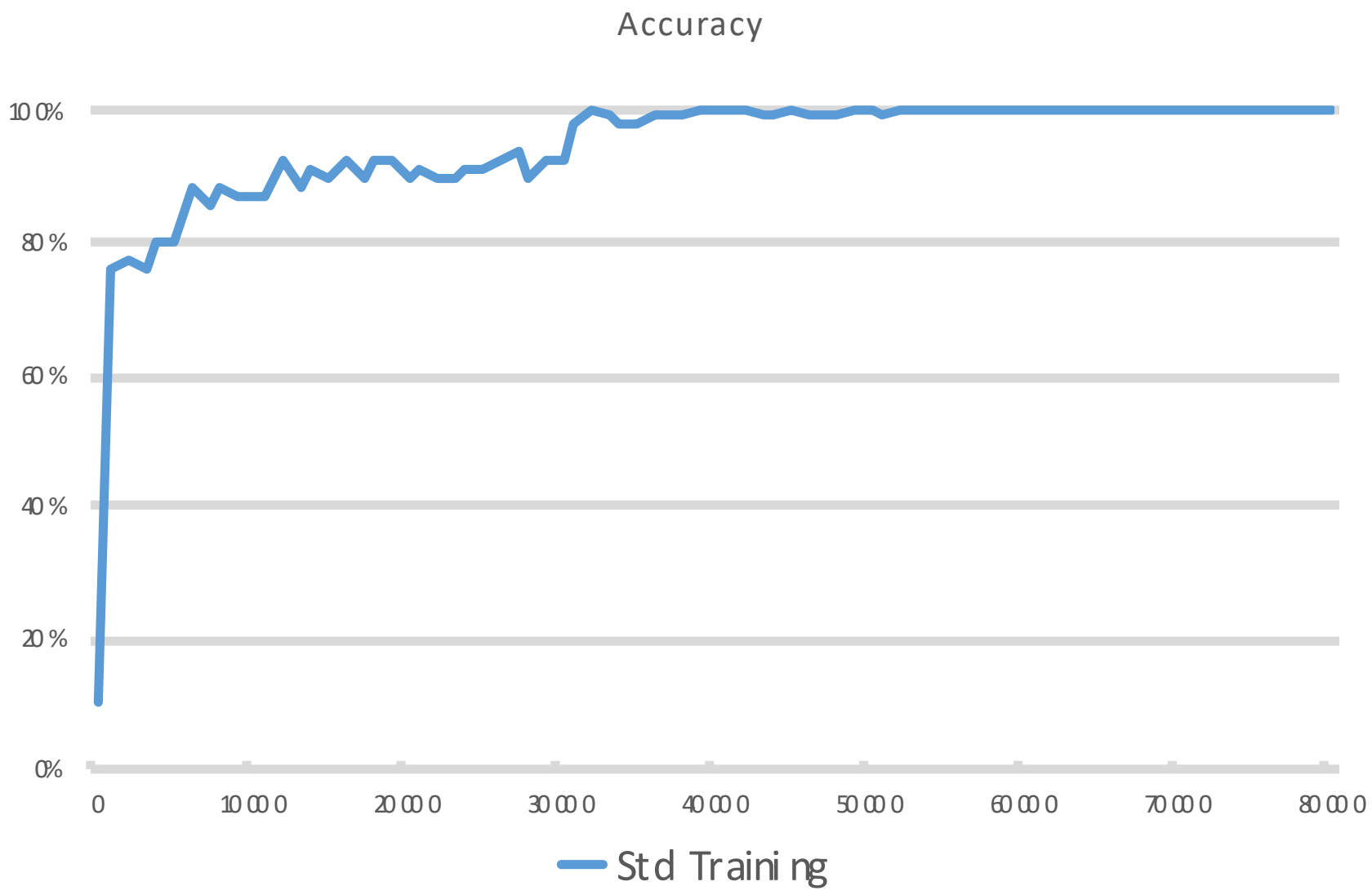
$$\mathbb{E}_{(x,y)\sim D}\left[loss(\theta, x, y)\right]$$

vs

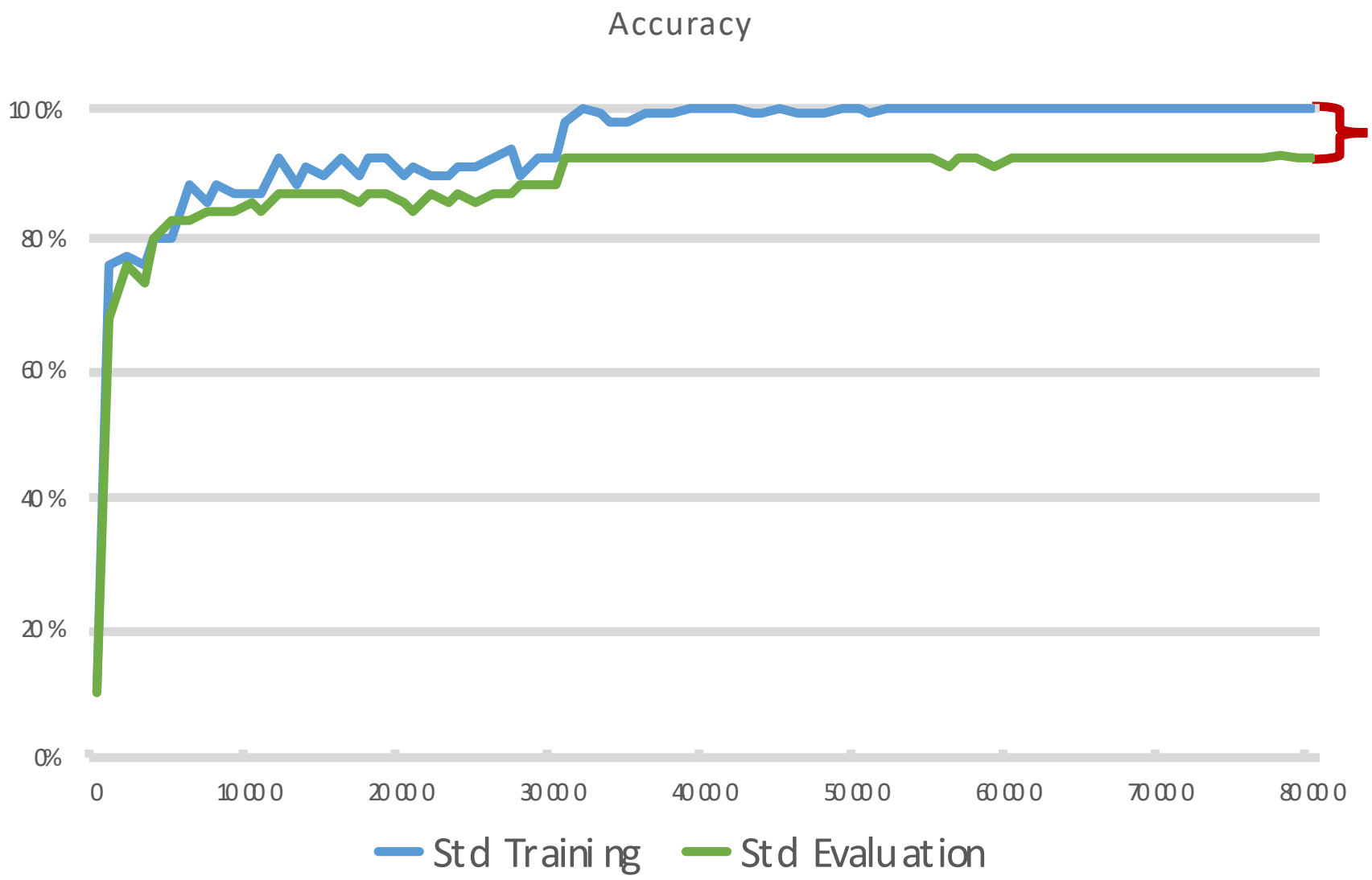
$$\mathbb{E}_{(x,y)\sim D}\left[\max_{\boldsymbol{\delta}\in\boldsymbol{\Delta}} loss(\theta, x+\boldsymbol{\delta}, y)\right]$$

(This goes **beyond** deep learning!)

Do Robust Deep Networks Overfit?

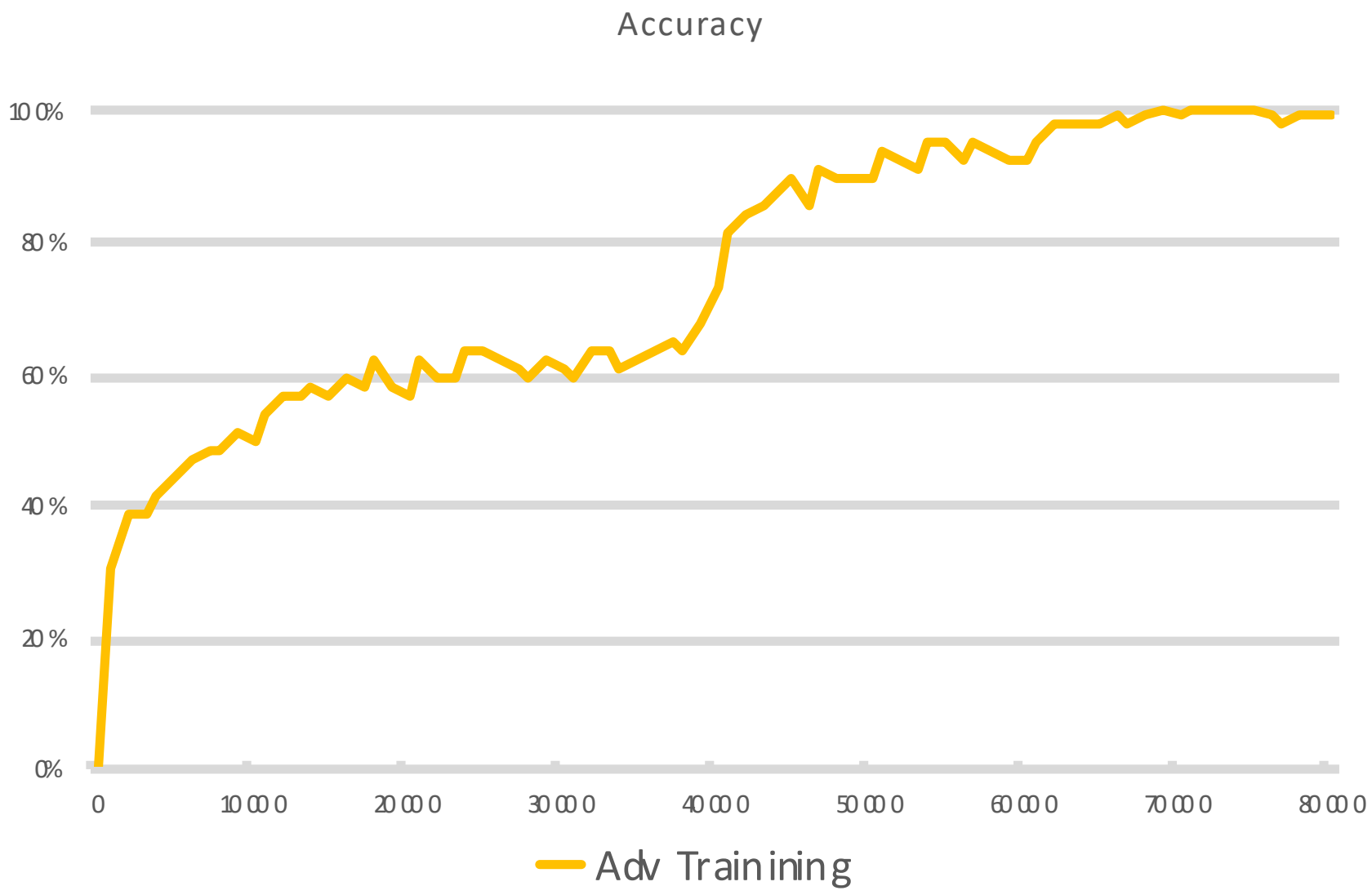


Do Robust Deep Networks Overfit?

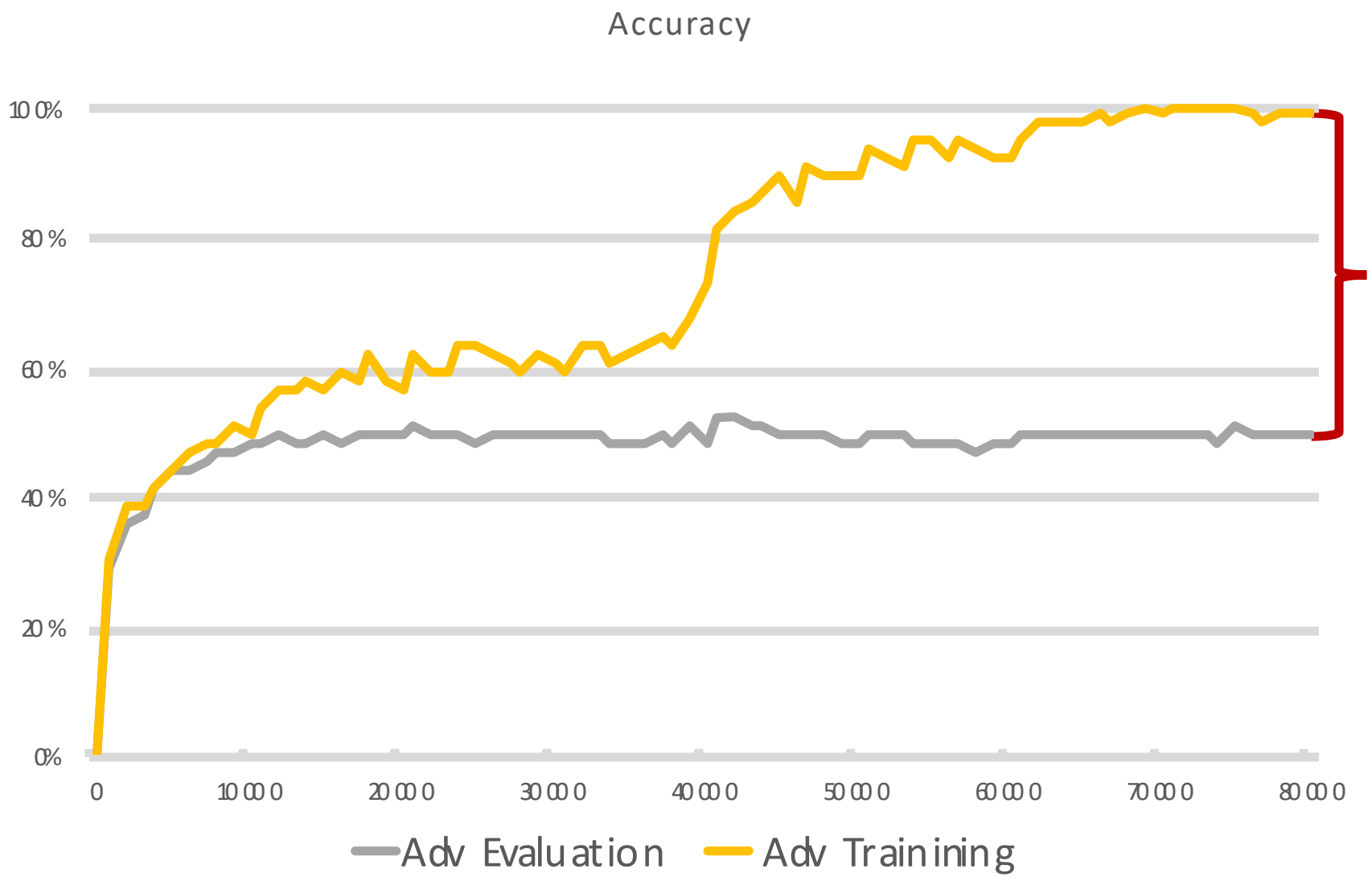


(small)
generalization gap

Do Robust Deep Networks Overfit?



Do Robust Deep Networks Overfit?



(large) generalization gap

→ Regularization does not help

Why?

Adv. Robust Generalization Needs More Data

Theorem [Schmidt Santurkar Tsipras Talwar M 2018]:
Sample complexity of adv. robust generalization can be **significantly larger** than that of "standard" generalization

Specifically: There exists a **d**-dimensional distribution **D** such that:
→ Given a **single** sample $(x_1, y_1) \sim D$ we can find a classifier **C** s.t.

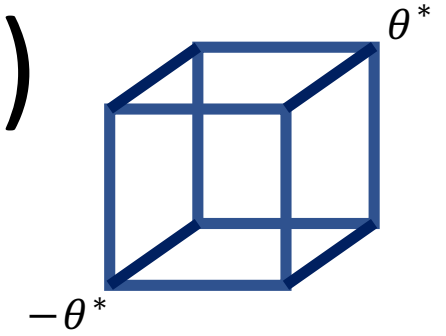
$$\Pr_{(x,y)\sim D}[C(x) = y] > 0.99$$

→ **But**: without seeing $\Omega(\sqrt{d})$ samples from **D,** we cannot find **C** s.t.

$$\Pr_{(x,y)\sim D}[\text{For all } \delta \in \Delta, C(x+\delta) = y] > 1/2 + O(1/d)$$

where $\Delta = \{\delta : \max_i |\delta_i| \leq \varepsilon\}$ and $\varepsilon = O(1/d^{1/4})$

Hard Distribution (for Linear Classifiers)



→ Fix $\boldsymbol{\theta^* \in_R \{+1,-1\}^d}$

→ To sample from **D:** choose $\boldsymbol{y \in_R \{+1,-1\}}$ and

x= 

$$x_i = \begin{cases} y\theta_i & \text{w.p. } ½ + \tau \\ -y\theta_i & \text{w.p. } ½ - \tau \end{cases}$$

Observe: If $(x_1,y_1)\sim D$ and we choose $C(x)= \langle w, x\rangle$ where $w= y_1 x_1$

Then, for $(x,y)\sim D$, $C(x)= y(4d$

→ If $\tau = C/d^{1/4}$, the classifier is almo

Can show: Unless we see $\Omega(\sqrt{d})$ samples, **no linear** classifier will work

But: Consider $\delta^* = -\varepsilon\, y\cdot sign(w)$ $(\in \Delta)$, then, for $(x,y)\sim D$,

$$C(x+\delta^*)= y(4d\tau^2 \pm O(d^{1/2}) - \varepsilon\,|w|_1) = y(4d\tau^2 \pm O(d^{1/2}) - \varepsilon\, d)$$

→ If $\varepsilon = C'/d^{1/4}$, the classifier is **always incorrect**

How About Non-Linear Classifiers?

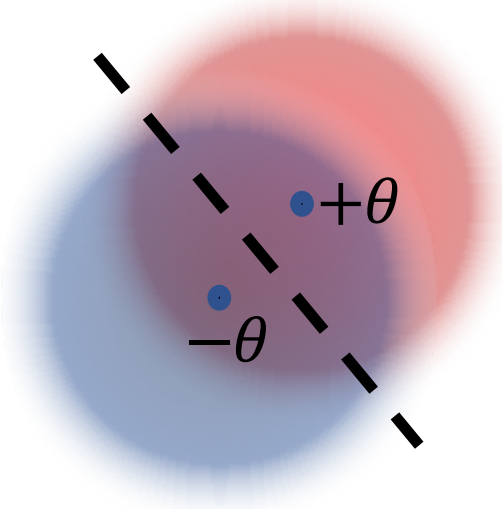
Observe: If we round our input **x'** to the nearest hypercube vertex,

Then, for **(x,y)~D** and **any δ ∈ Δ,** **x'= x + δ → x**

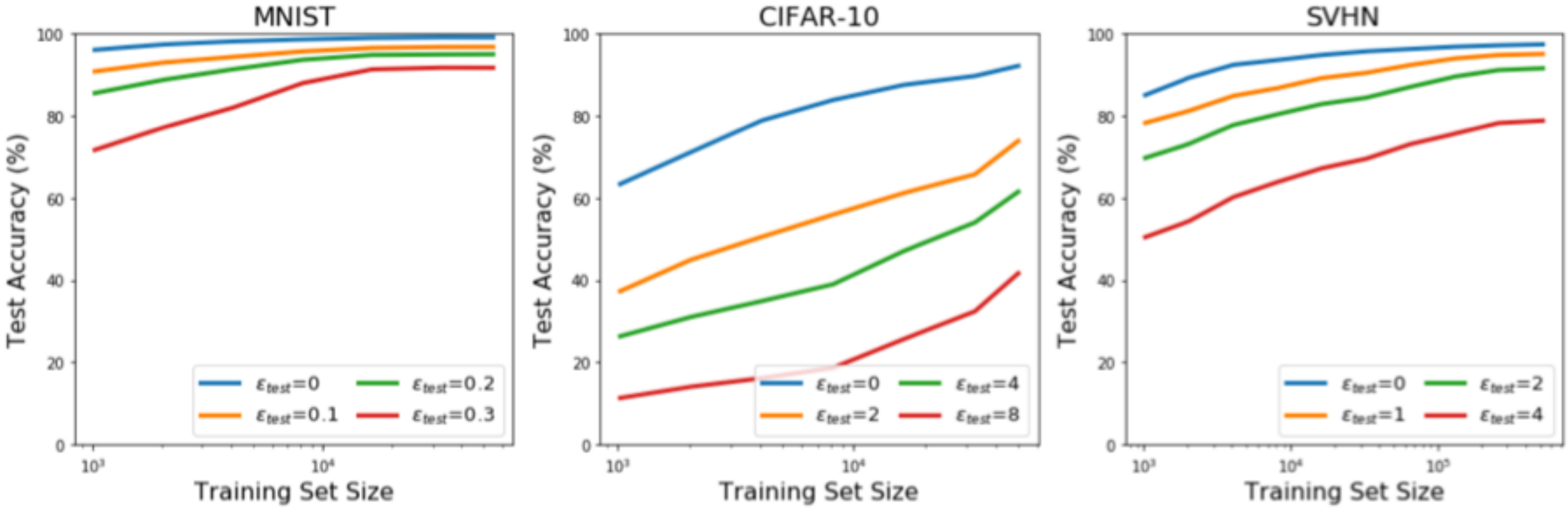
→ Adversarial perturbations have no effect (and thus are not a problem)

Maybe adversarial robustness does not need more data after all?

No: There are distributions for which the sample complexity separation arises for **any** classifier



How Does This Look Like in Practice?



Does Being Robust Help “Standard” Generalization?

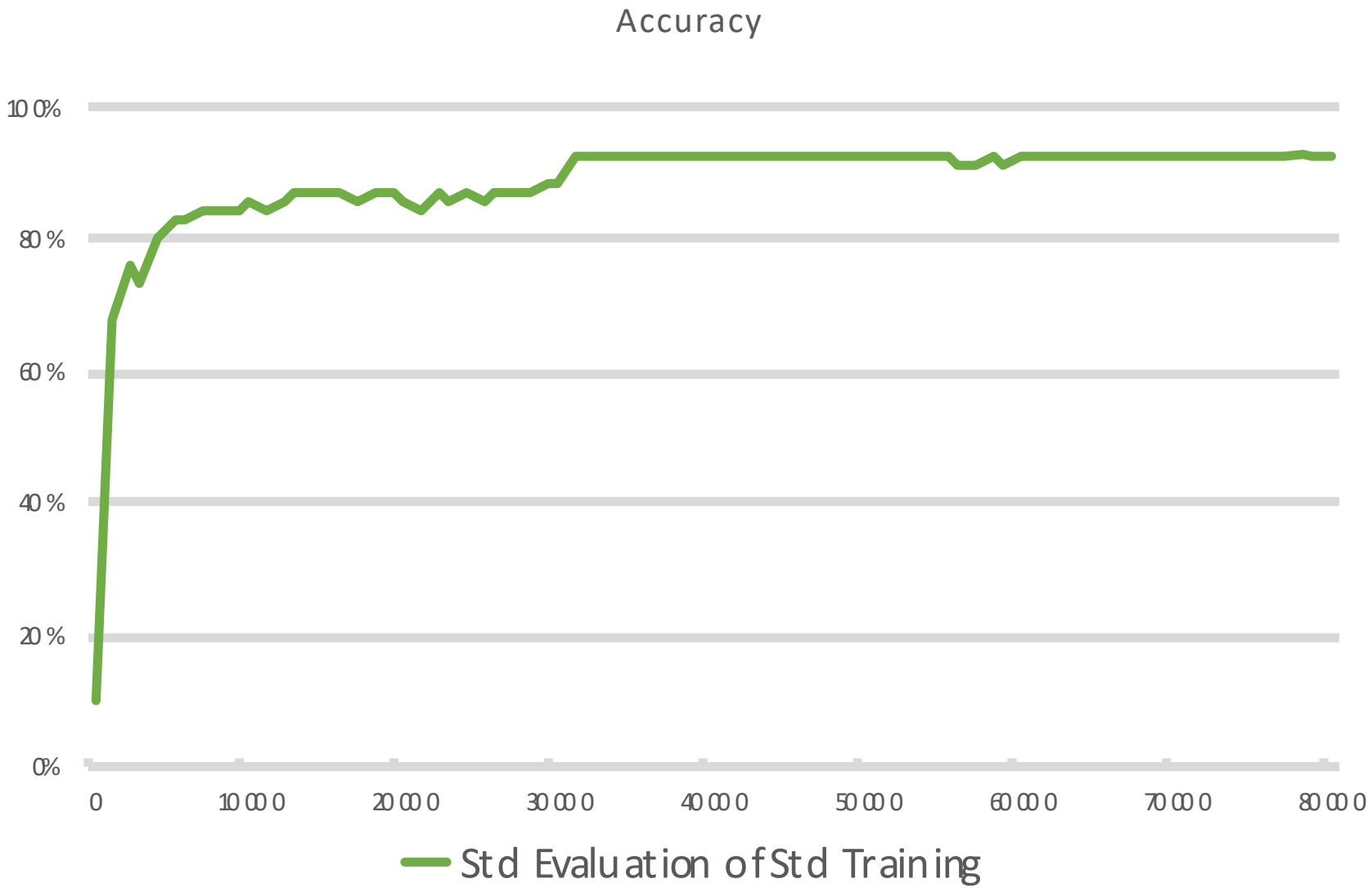
Data augmentation: Enlarging your training set by adding “invariantly transformed” version of the original training inputs

→ A popular and effective technique for improving “standard” generalization

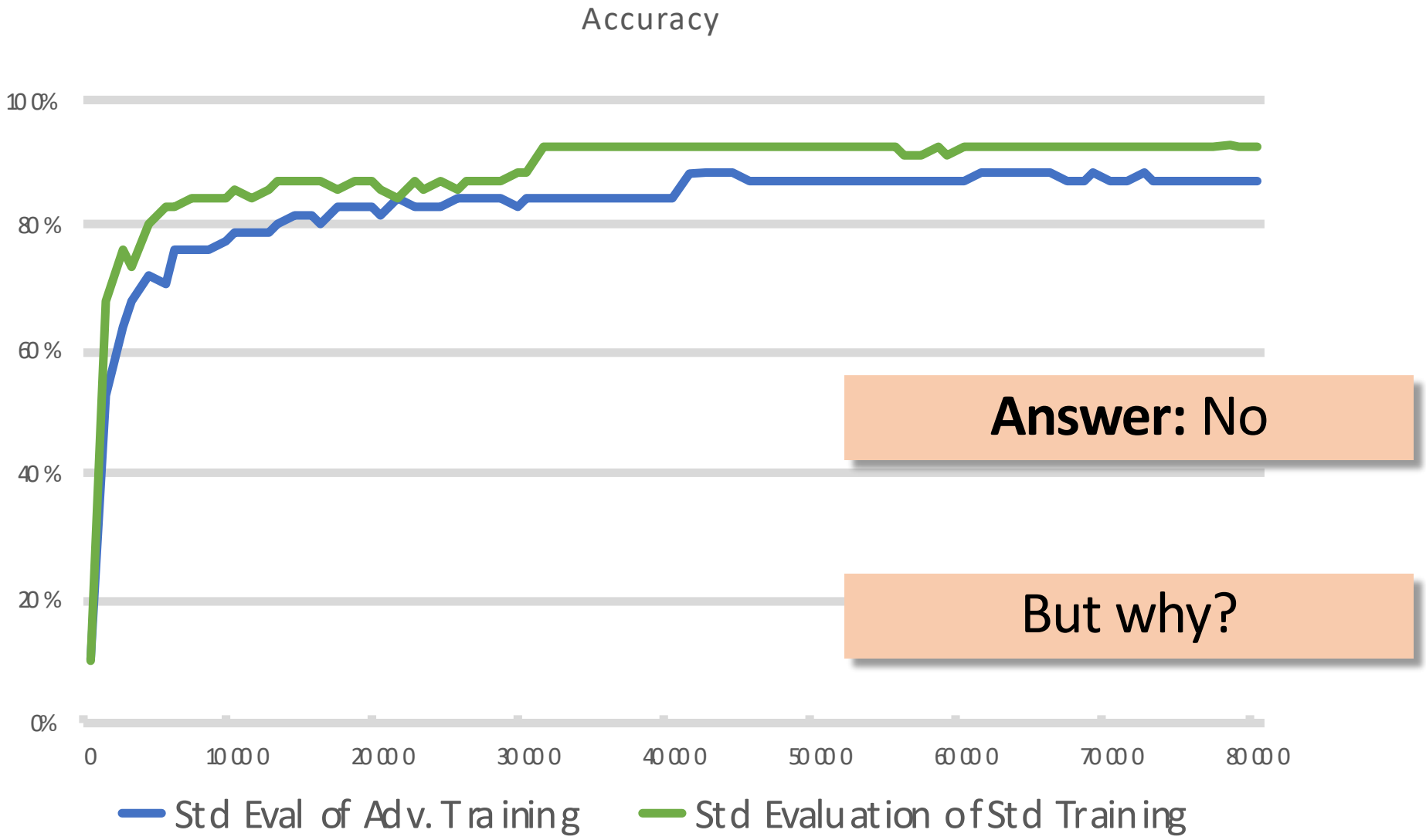
→ Adversarial training (training on adv. perturbed version of training inputs) = an “ultimate” version of data augmentation? (since we train on the ”most confusing” version of the training set)

Does adversarial training improve “standard” generalization?

Does Being Robust Help “Standard” Generalization?



Does Being Robust Help “Standard” Generalization?



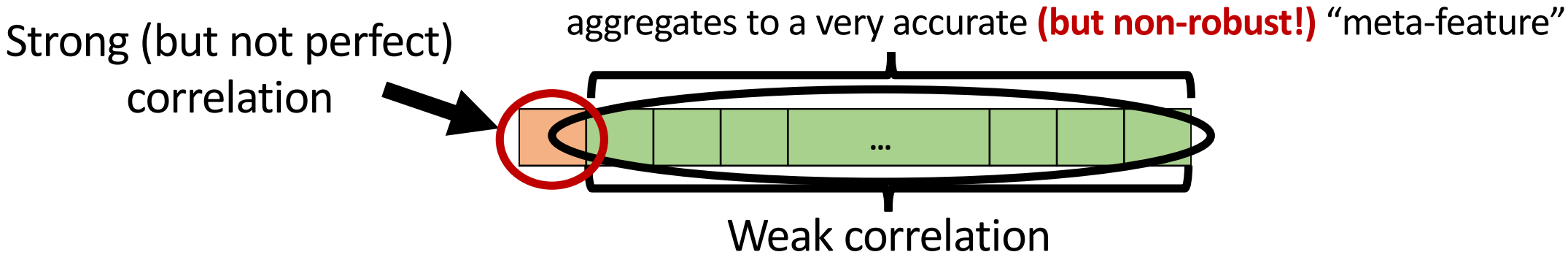
Does Being Robust Help "Standard" Generalization?

Theorem [Tsipras* Santurkar* Engstrom* Turner M 2018]:
There is no "free lunch", i.e., when training a classifier, there can be an inherent trade-off between "standard" accuracy and adv. robustness

Basic intuition: There is "robust" vs. "non-robust" feature effect

→ For "std." accuracy, every feature that has **any** correlation with correct label is useful

→ In adv. robust setting, every used feature increases the "exposure", so it is sometimes better to drop a feature if it is not sufficiently useful



→ "Standard" training will give a classifier that relies on the "meta-feature"

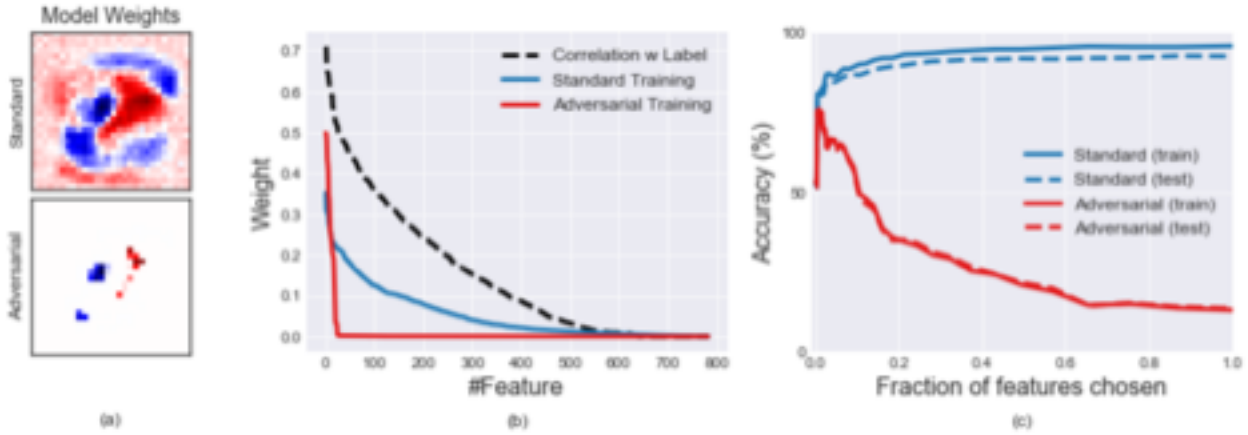
→ Adversarial training will only rely on the single strongly correlated feature

Does Being Robust Help "Standard" Generalization?

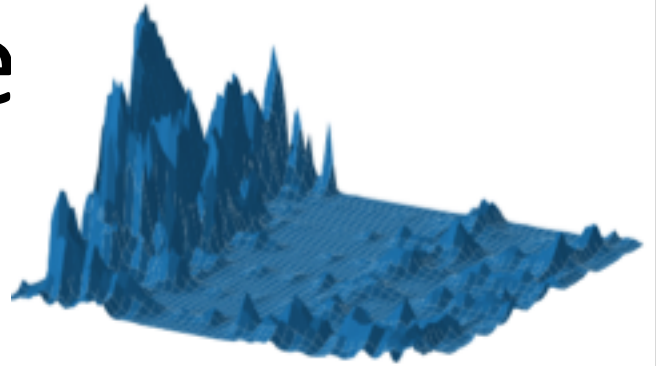
Theorem [Tsipras* Santurkar* Engstrom* Turner M 2018]:
There is no "free lunch", i.e., when training a classifier, there can be an inherent trade-off between "standard" accuracy and adv. robustness

Basic intuition: There is "robust" vs. "non-robust" feature effect
→ For "std." accuracy, every feature that has **any** correlation with correct label is useful
→ In adv. robust setting, every used feature increases the "exposure", so it is sometimes better to drop a feature if it is not sufficiently useful

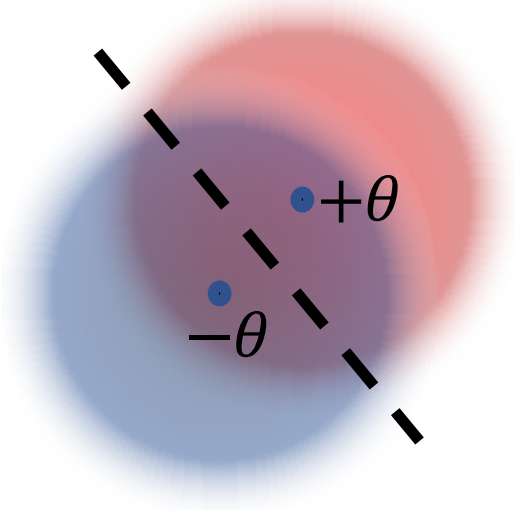
Also gives a new, alternative way of training adv. robust **linear** classifiers



Adversarial Robustness is Not Free

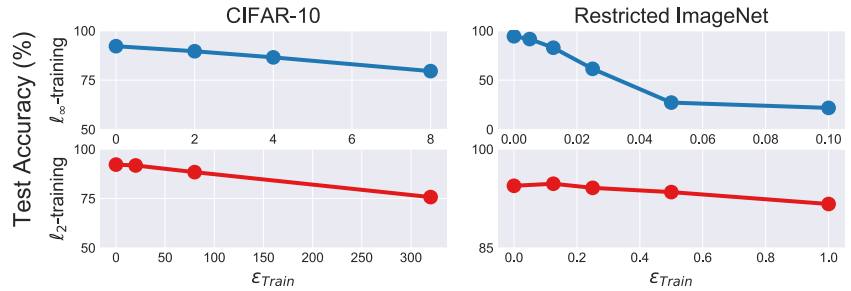


→ Optimization during training more difficult



→ More training data might be required
[Schmidt Santurkar Tsipras Talwar M 2018]

→ Might need to lose "standard" accuracy
[Tsipras* Santurkar* Engstrom* Turner M 2018]



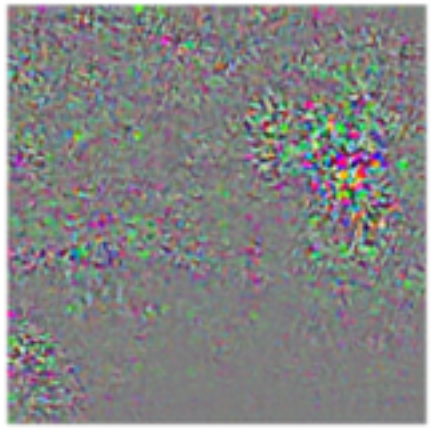
→ There might be also computational barriers too
[Bubeck Price Razenshteyn 2018]

But There Are (Unexpected?) Benefits Too

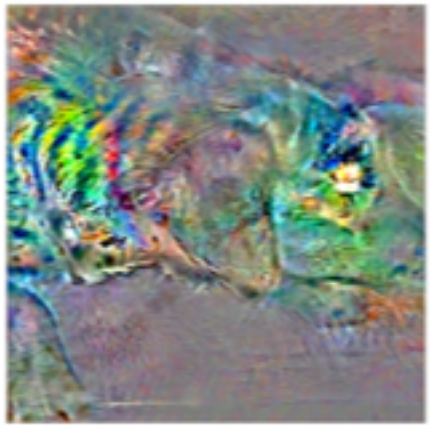
[Tsipras* Santurkar* Engstrom* Turner M 2018]

→ Gradients are more **interpretable** (they yield saliency maps)

Input



gradient of
standard model



gradient of
adv. robust model

→ "Adversarial" examples become **semantically meaningful**

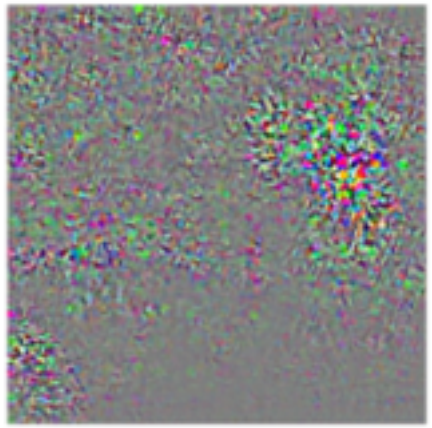
Adversarial example for
standard model

But There Are (Unexpected?) Benefits Too

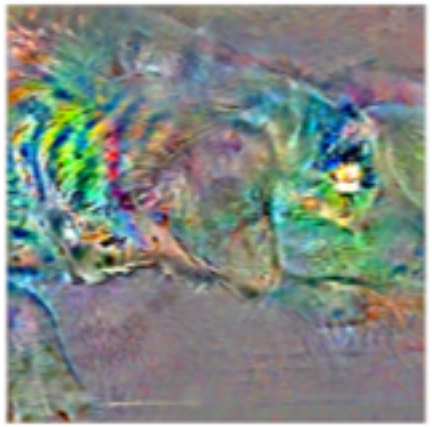
[Tsipras* Santurkar* Engstrom* Turner M 2018]

→ Gradients are more **interpretable** (they yield saliency maps)

Input

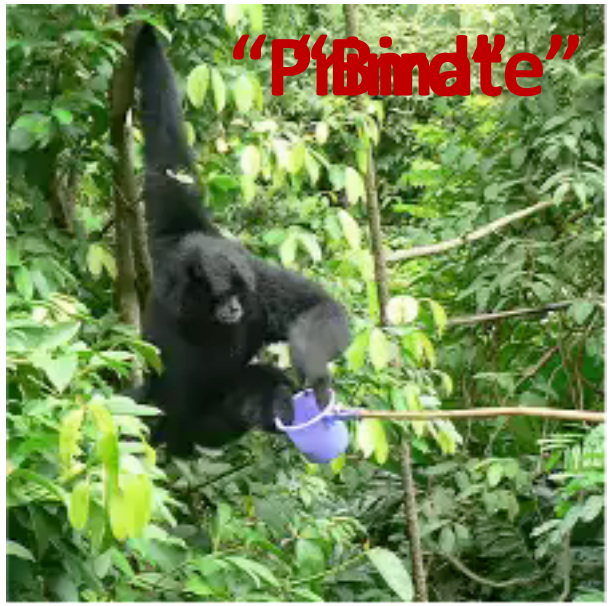


gradient of
standard model



gradient of
adv. robust model

→ "Adversarial" examples become **semantically meaningful**

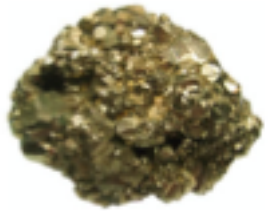


"Adversarial" example for
adv. robust model

Conclusions

→ We're getting somewhere in ML and this is exciting

→ **But:** It is still Wild West out there
(we stuck gold but there is lots of fool's gold too)



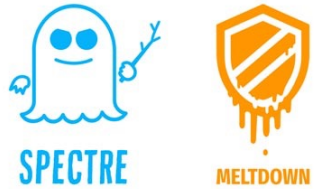
Next frontier: Building ML you can truly rely on

We need to:

→ Attain a principled understanding of core techniques and tools

→ Rethink the whole pipeline from a robustness/safety/security perspective

"Theory mindset" can have a lot of impact here



(But think of this more as "science" than "mathematics")

@aleks_madry

madry-lab.ml