Robust Machine Learning: Progress, Challenges, Humans

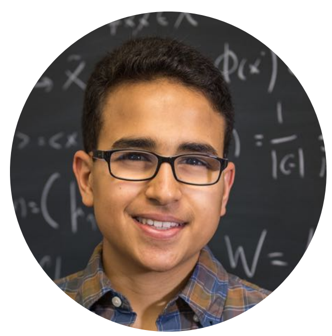
Dimitris Tsipras

@tsiprasd

gradient-science.org

joint work with

Logan Engstrom



Andrew Ilyas

Aleksandar Makelov

Shibani Santurkar

Ludwig Schmidt

Kunal Talwar

Brandon Tran

Alexander Turner

Adrian Vladu

Aleksander Mądry

Deep Learning can be amazing

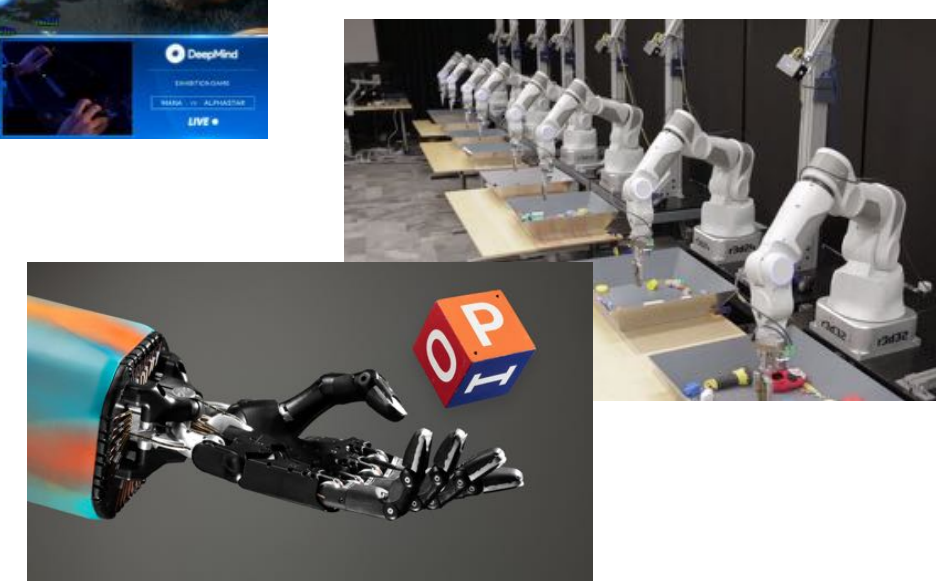
Image Classification

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

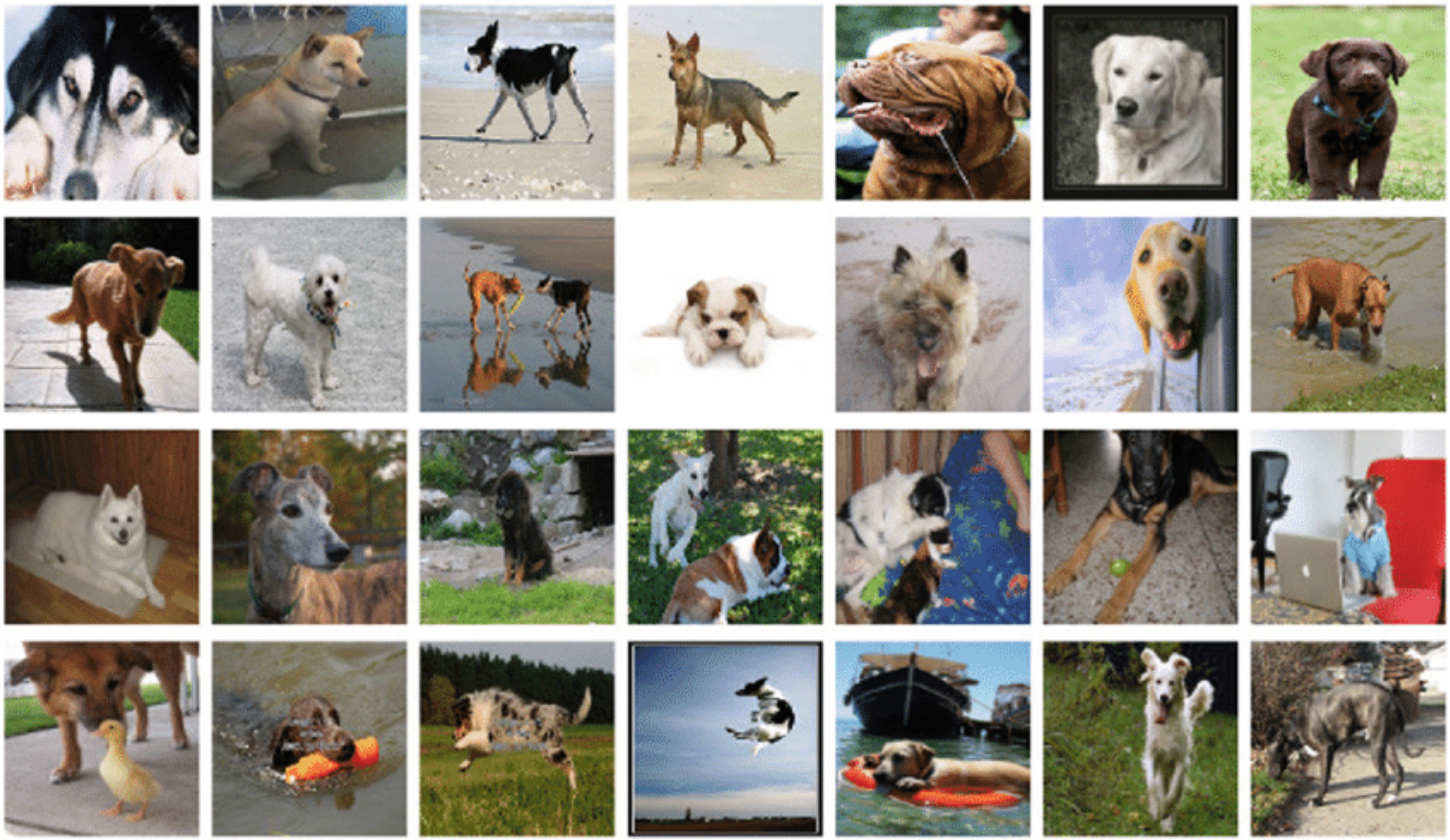
Strategy Games

Realistic Image Generation

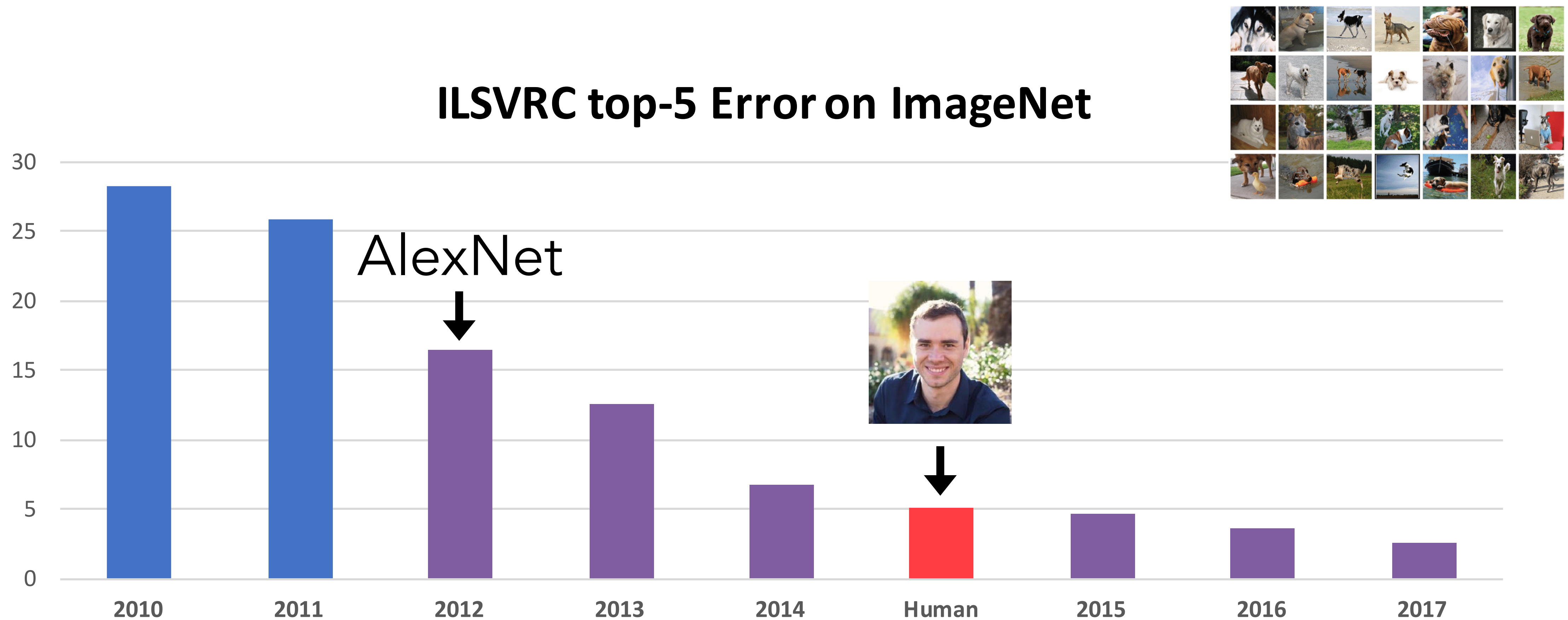


Robotic Manipulation

ImageNet: A success story

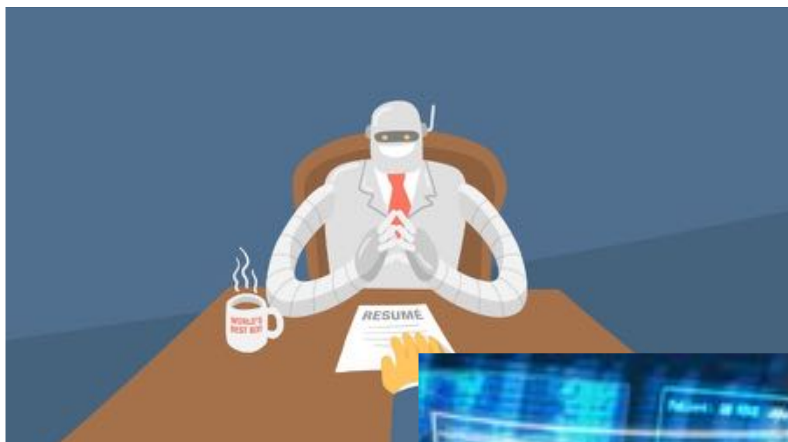


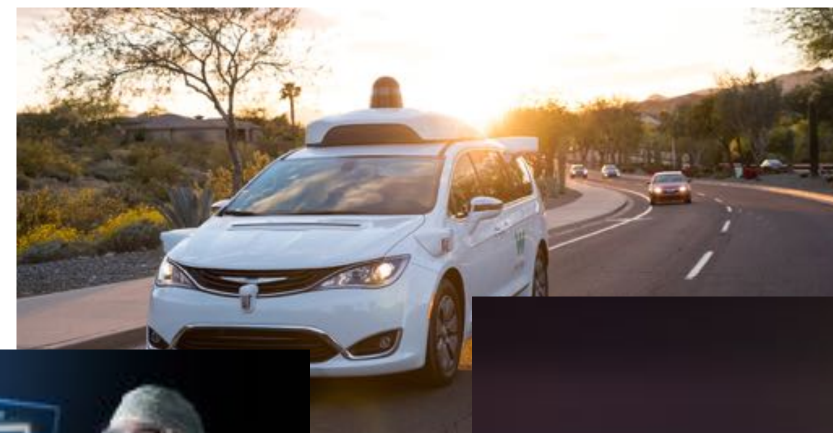
ImageNet: A success story

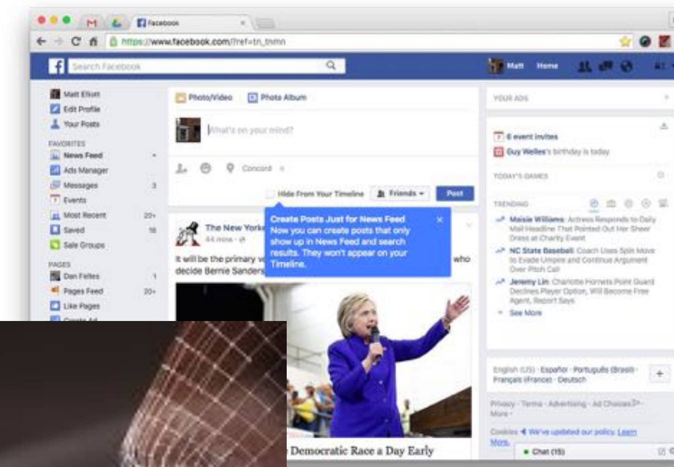


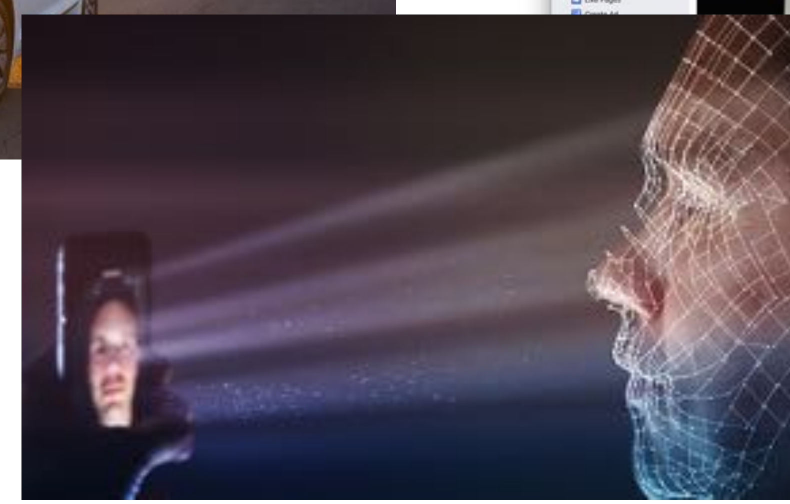
Have we achieved truly super-human performance?

Real-world deployment









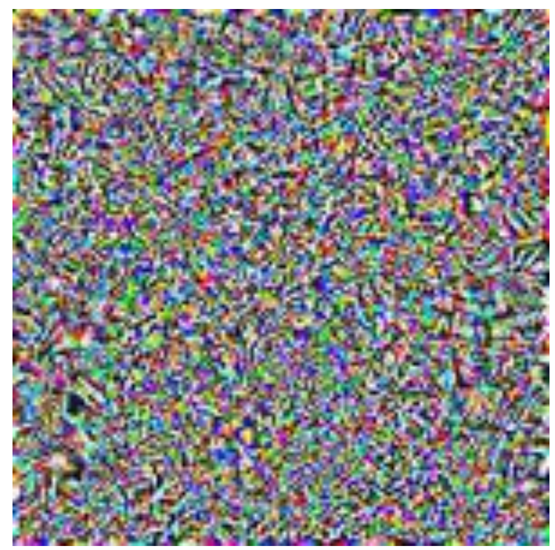
Are ML systems ready for the real world?

Core issue: Brittleness

"pig" (91%)

+0.005x

adversarial noise



=

"airliner" (99%)

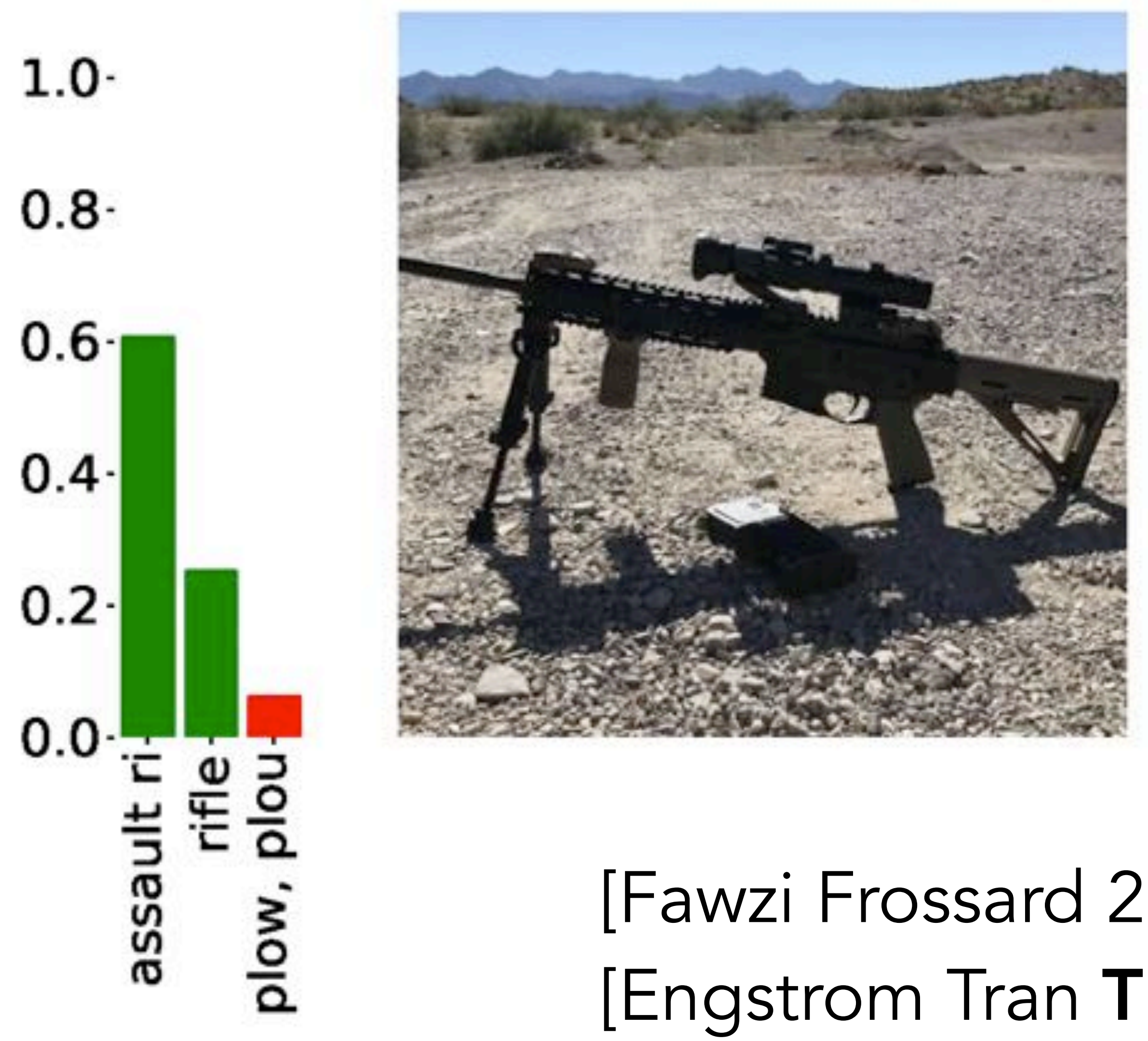
[Szegedy et al. 2013]

Long history in "standard" ML:
[Biggio et al. 2013] [Dalvi et al. 2004][Lowd Meek 2005] [Globerson Roweis 2006][Kolcz Teo 2009][Barreno et al. 2010] [Biggio et al. 2010][Biggio et al. 2014][Srndic Laskov 2013]

Real-world perturbations?

[Athalye Engstrom Ilyas Kwok 2017]

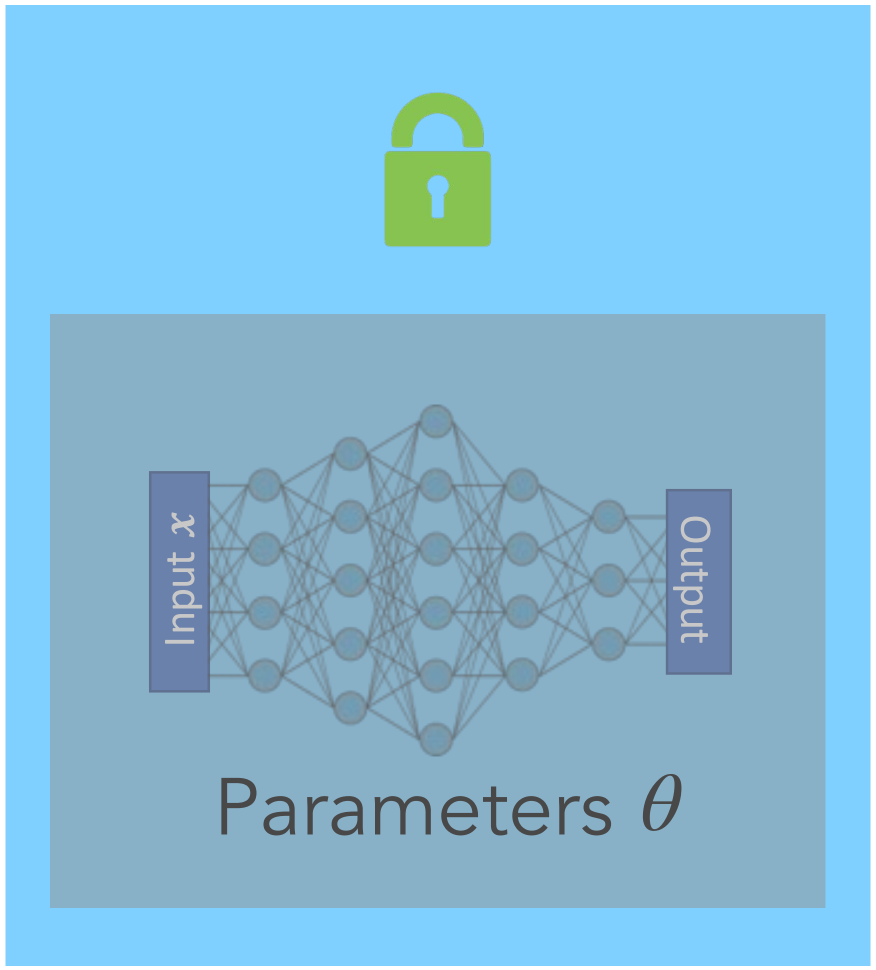
More natural examples?

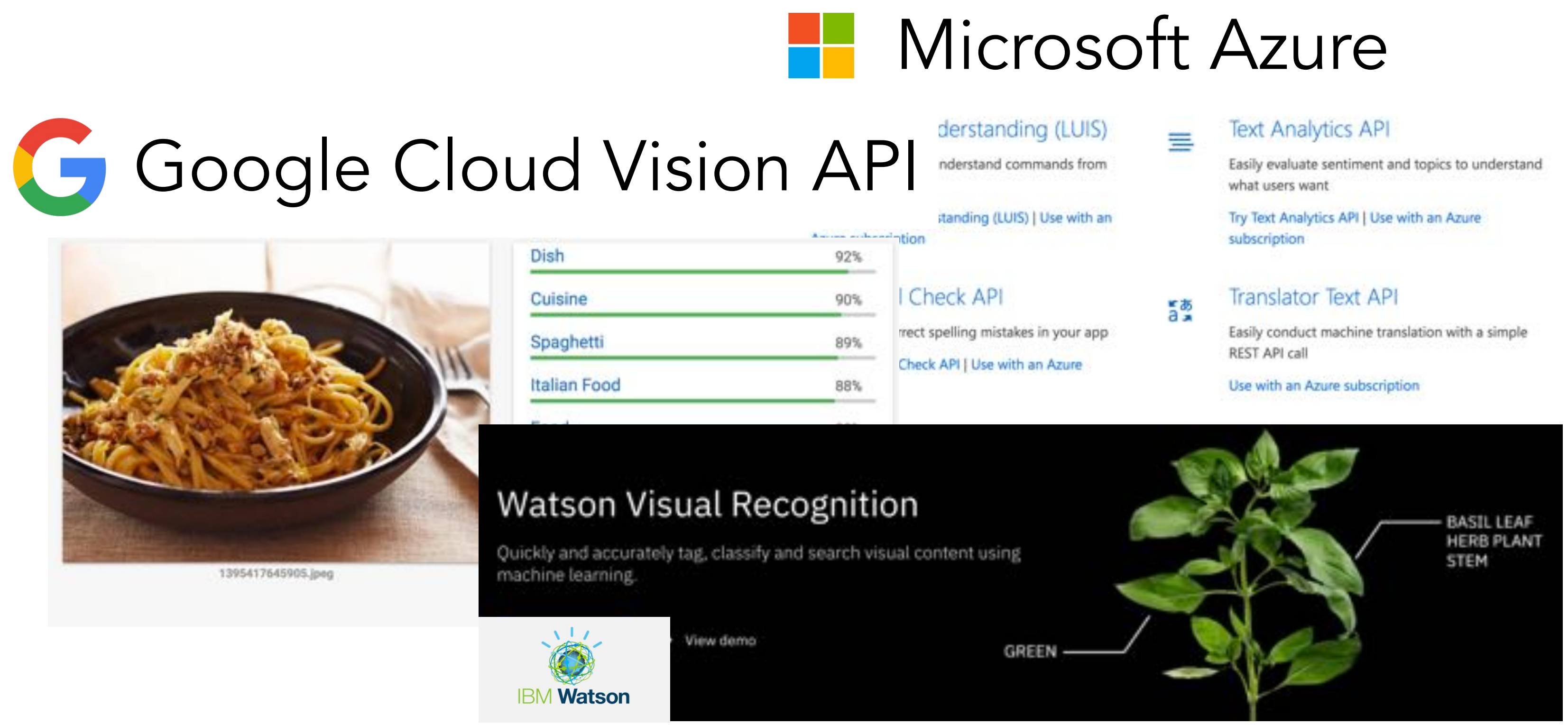


[Fawzi Frossard 2015]
[Engstrom Tran **T** Schmidt Madry 2017]

Training on rotations does not solve the problem

Black-box attacks?



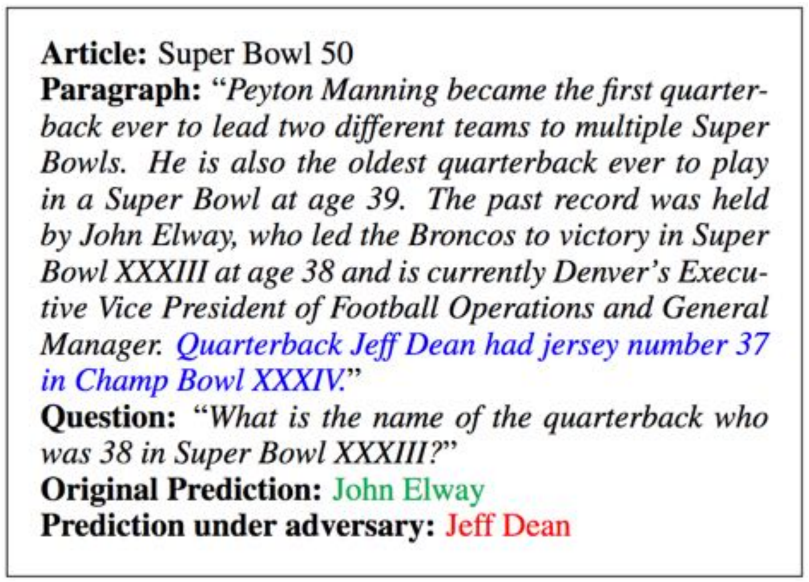


Does black-box mean secure? **No.**

Query attacks: Directly use input-output queries [Chen et al. 2017]

Transfer attacks: Just attack a similar model [Szegedy et al. 2013, Papernot et al. 2016]

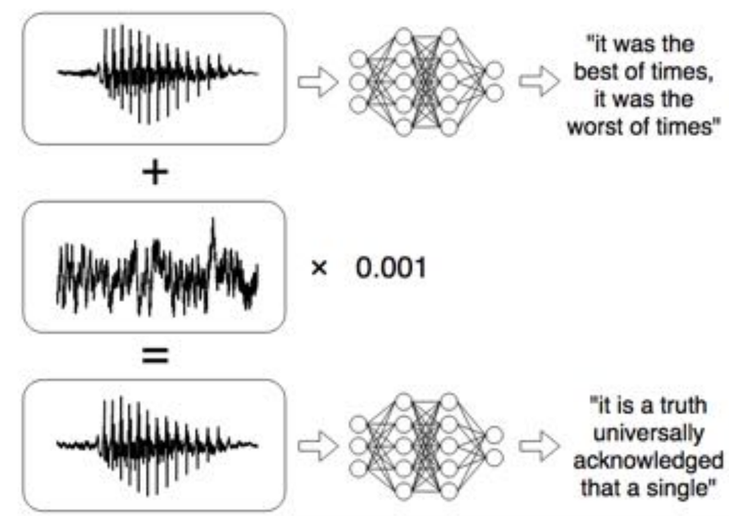
Beyond images?



Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*"
Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

[Jia Liang 2017]: Irrelevant sentences confuse reading comprehension models

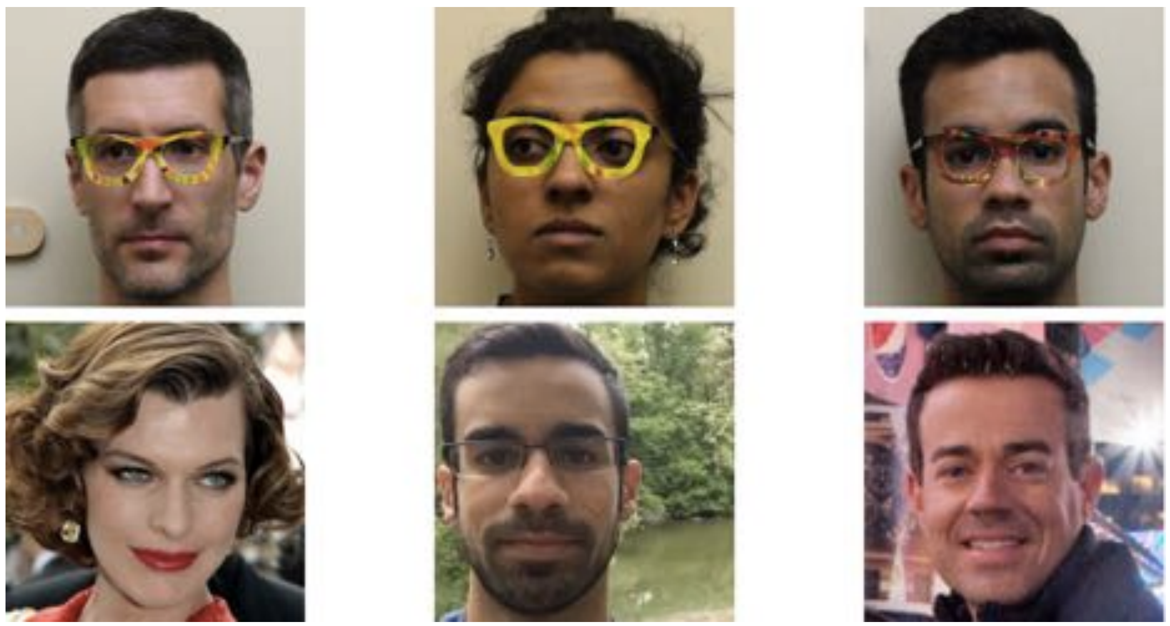
[Carlini Wagner. 2018]: Can arbitrarily confuse a speech recognition system



[Grosse et al. 2017]: Small changes can bypass malware detection systems

Why should we care?

Security



[Sharif et al. 2016]

[Evtimov et al. 2018]

Already issues with **spam** and **content filtering**

Reliability

What we expect from AI

What we (sometimes) get

ML models are
very brittle

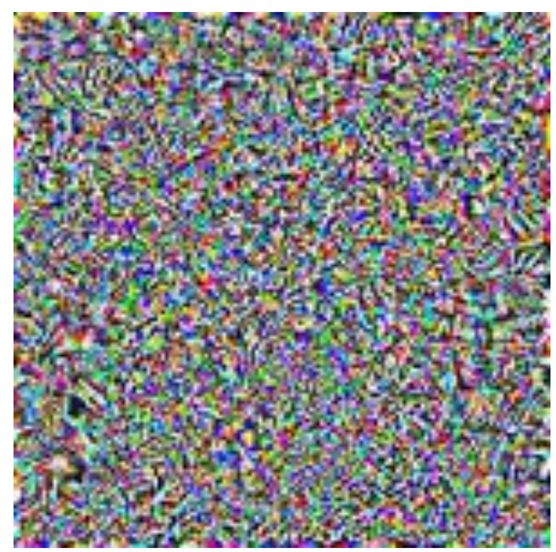
Human Alignment

How are DL models making predictions?

"pig" (91%)

+0.005x

adversarial noise



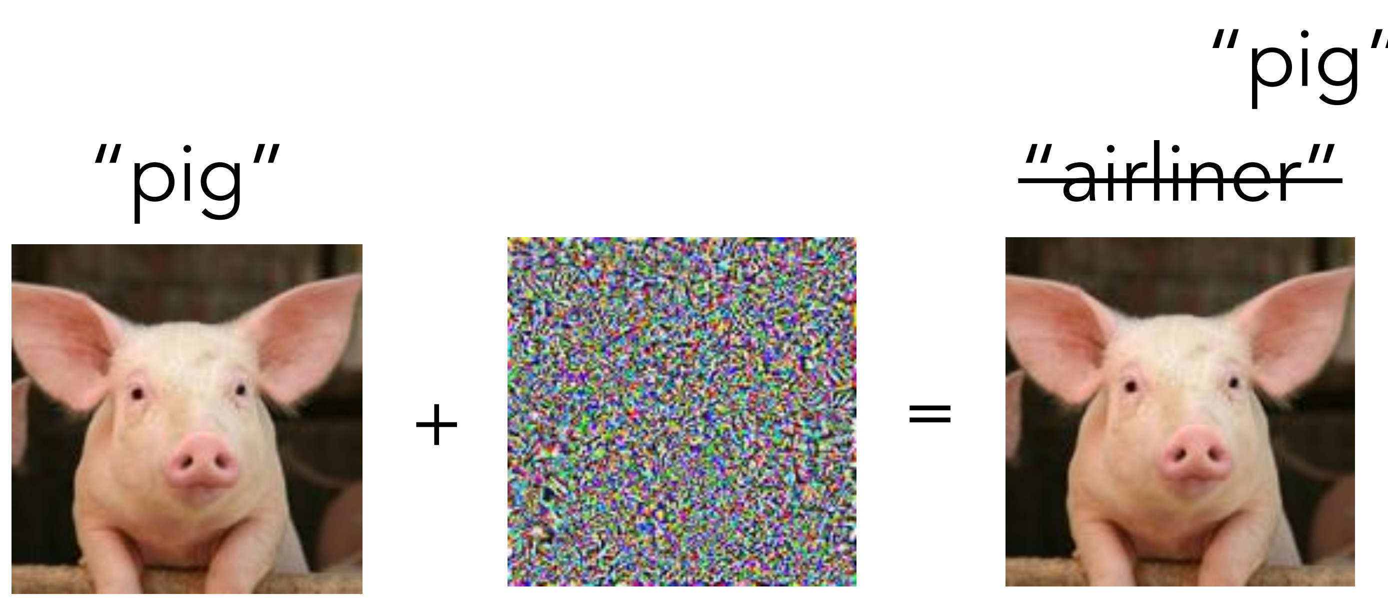
=

"airliner" (99%)

Why is this important to the model?

How do we train robust models?

Our focus:



How do we find adv. examples?

Standard training

model parameters, input, label

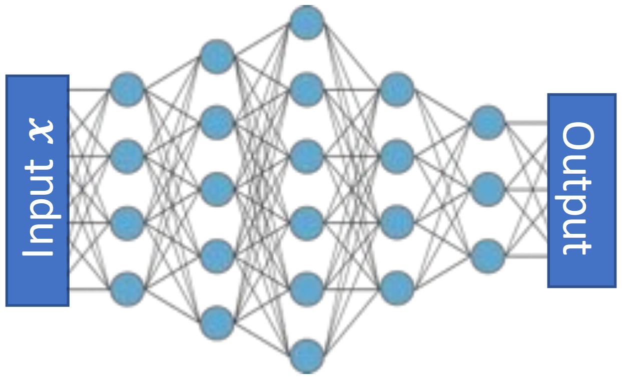
$$\min_{\theta} \mathbb{E}_{x,y\sim D}\,[loss(\theta, x, y)]$$

Adversarial attacks

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

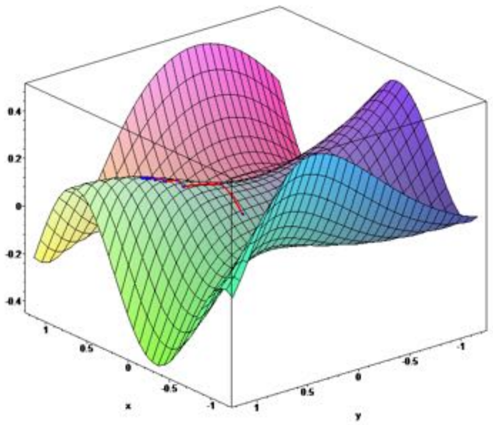
Allowed perturbations: pixel-wise, rotations, …

differentiable



Parameters θ

Gradient Descent to find θ



How do we train robustly?

Key observation: Adversarial examples are **not** at odds with standard learning

Standard Generalization:

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

Adversarially Robust Generalization:

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

Explicit set of invariances

Towards robust models

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

finding a robust model

(Stochastic) Gradient Descent on θ

finding a worst-case perturbation

(Projected) Gradient Descent on δ

(How do we get gradients of the max?)

Theorem (Danskin): Gradient at maximizer → Gradient of max

$$\nabla_y \max_x f(x,y) = \nabla_y f(x^\star, y) \qquad x^\star = \arg\max_x f(x,y)$$

Towards robust models

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

finding a robust model

finding a worst-case perturbation

Improve robustness: Train on perturbed inputs

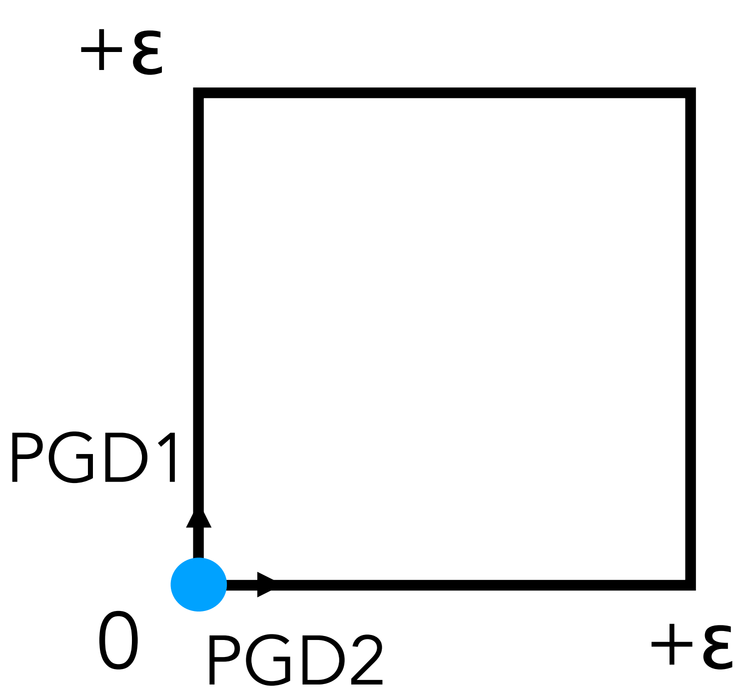
(aka "adversarial training" [Goodfellow et al. 2015])

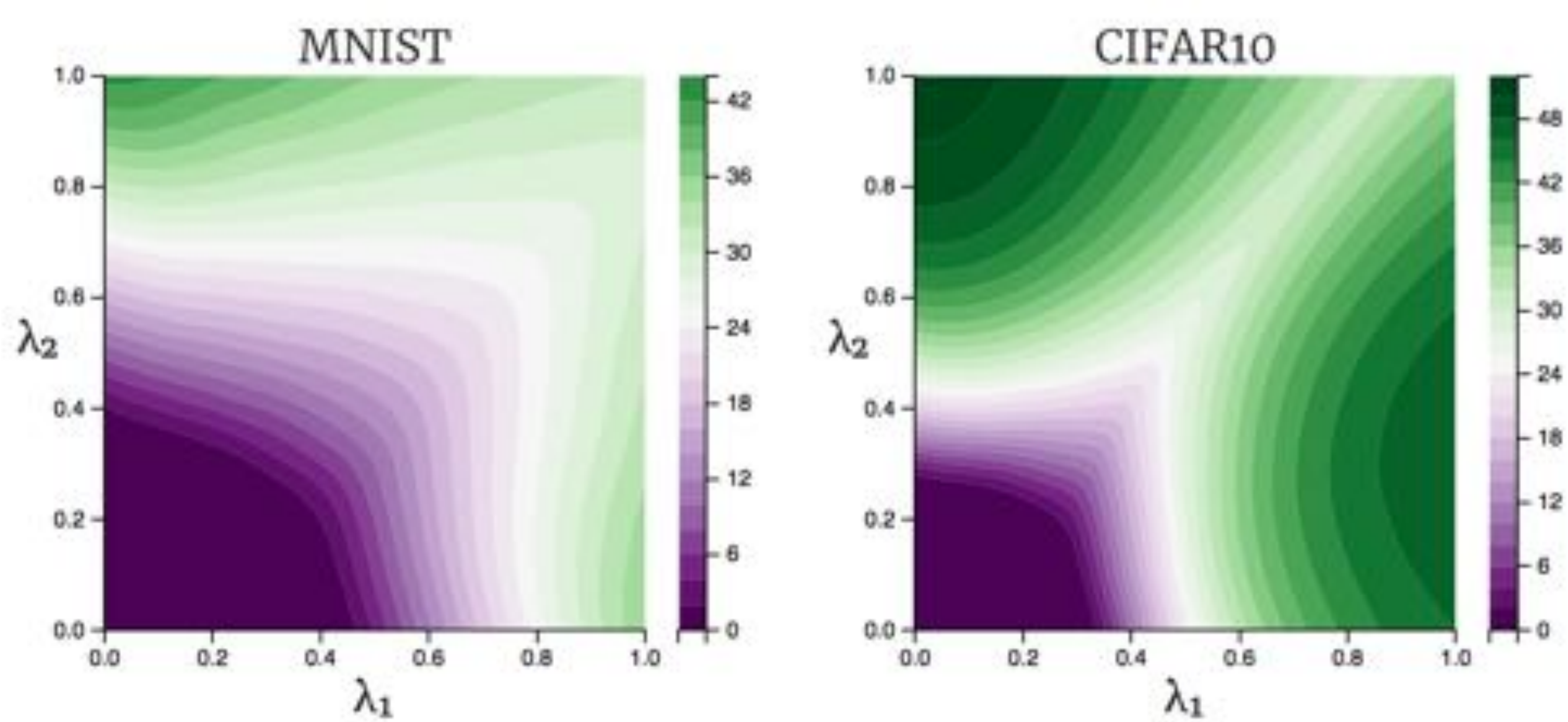
Actually leads to **robust models** (with some care)

Key ingredient 1: Reliable attacks

We need to train on (almost) **worst-case inputs**

But: DNN loss is **non-convex**



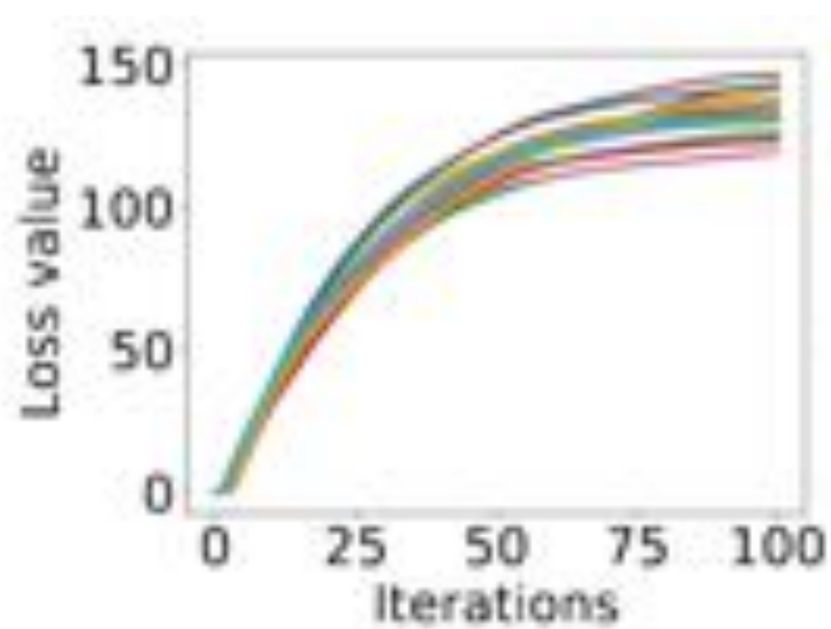


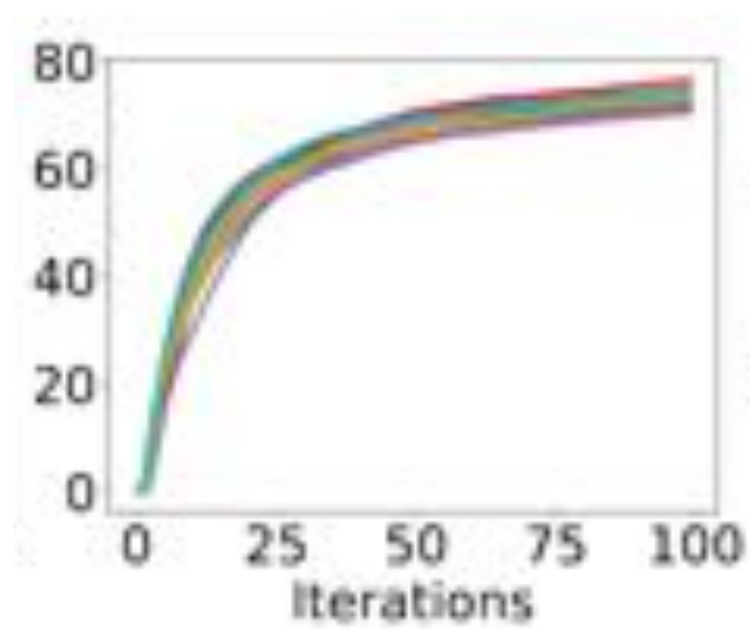
Key ingredient 1: Reliable attacks

We need to train on (almost) **worst-case inputs**

But: DNN loss is **non-convex**

PGD can still find worst-case inputs **reliably**

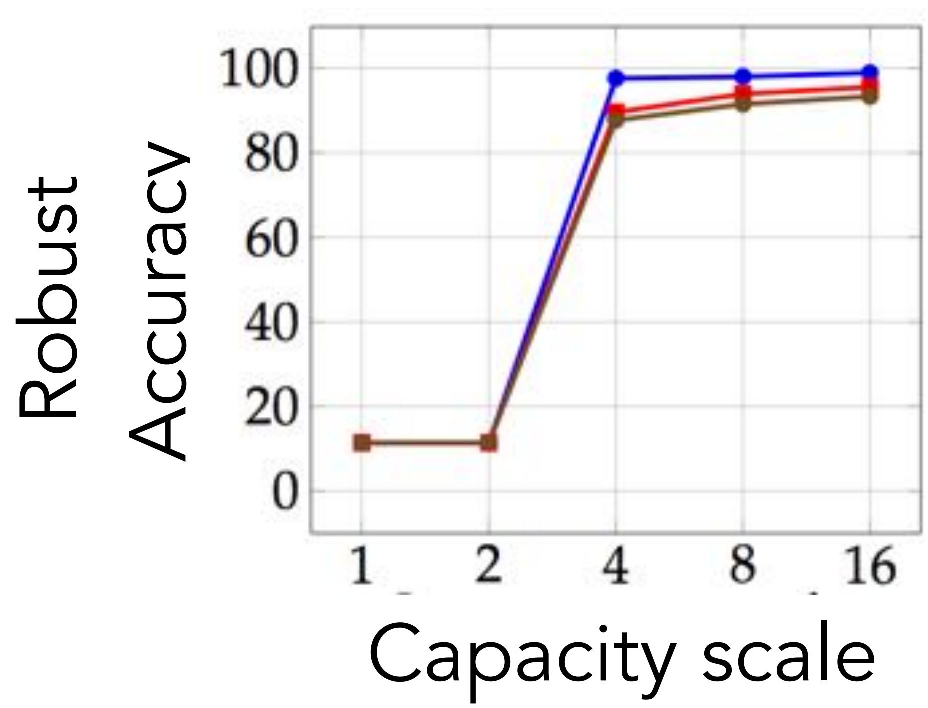




Consistent behavior from random starts

Key ingredient 2: Capacity

Robust models may need to be **more expressive**



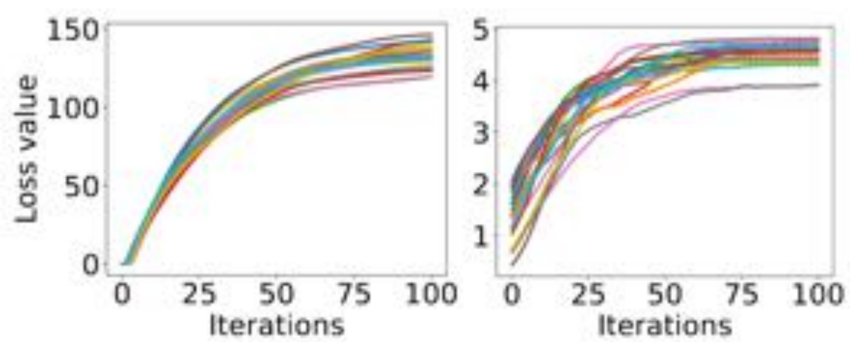
Weak models can fail to train

Higher capacity ⇒ more robust

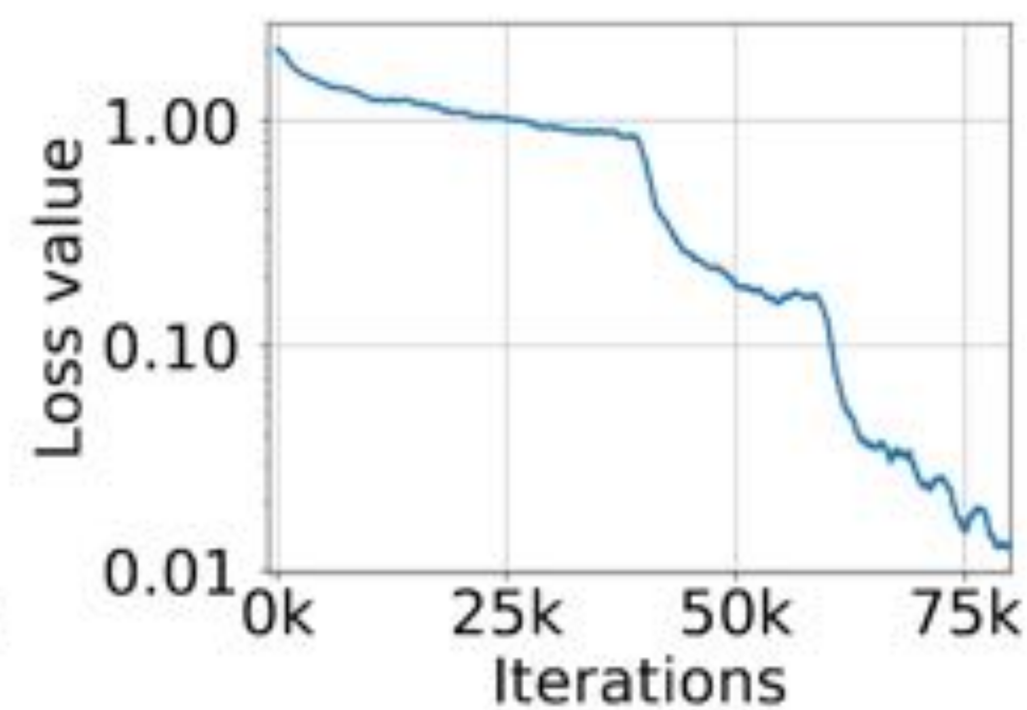
Robust models

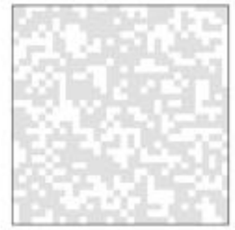
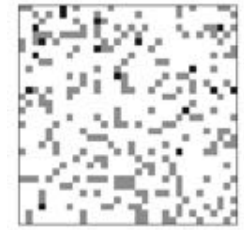
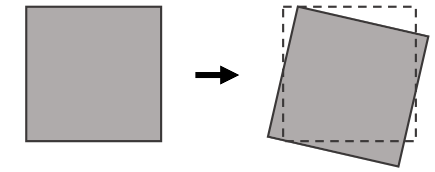
Reliable attacks

Sufficient capacity



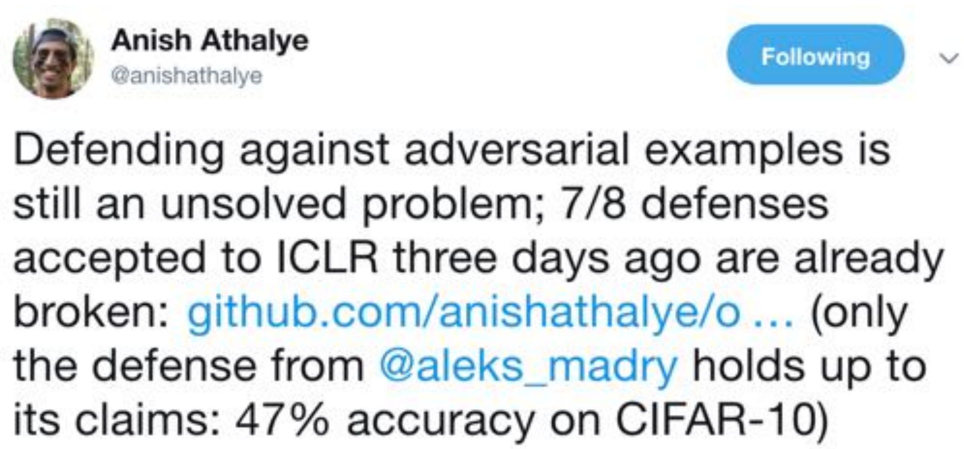
Result: Adversarial loss **decreases steadily**



	ℓ_∞-norm 	ℓ_2-norm 	Rotation+Translation 
MNIST 	ε = 0.3 89%	ε = 2.5 66%	ε = ±3px, ±30° 98%
CIFAR-10 	ε = 8/255 53%	ε = 0.5 70%	ε = ±3px, ±30° 82%
ImageNet 	ε = 4/255 33%	ε = 1 50%	ε = ±3px, ±30° 57%

Evaluating robustness can be hard

Many defenses are broken by **adaptive attacks**


Anish Athalye
@anishathalye
Following

Defending against adversarial examples is still an unsolved problem; 7/8 defenses accepted to ICLR three days ago are already broken: github.com/anishathalye/o … (only the defense from @aleks_madry holds up to its claims: 47% accuracy on CIFAR-10)

[Carlini Wagner 2016] [Carlini Wagner 2017] [Carlini Wagner 2017] [Athalye et al. 2018] [Uesato et al. 2018]

Try multiple adaptive attacks

Release code and models

RobustML
robust-ml.org

Formal robustness verification

Prove robustness on specific examples

Verification

[Tjeng et al. 2019]

MIP solvers

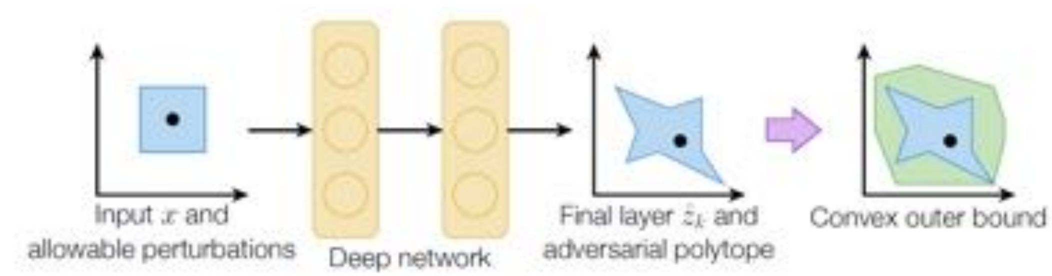
$$
\begin{array}{ll}
 & \min_{x'} d(x', x) \\
\text{subject to} & \text{argmax}_i (f_i(x')) \neq \lambda(x) \\
 & x' \in \mathcal{X}_{valid}
\end{array}
$$

Accurate but intractable

Certification

[Wong Kolter 2018]

Convex relaxation



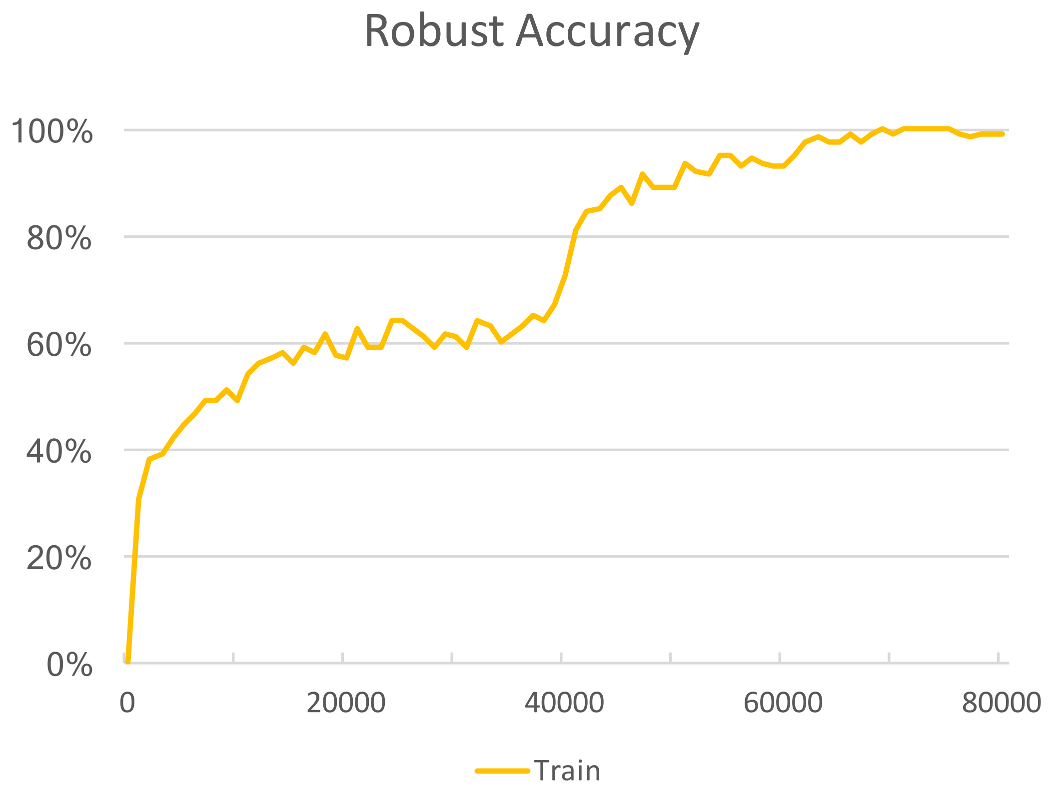
Bounds might be too loose

Accurate and **efficient** verification largely open

Why is robust learning so hard?

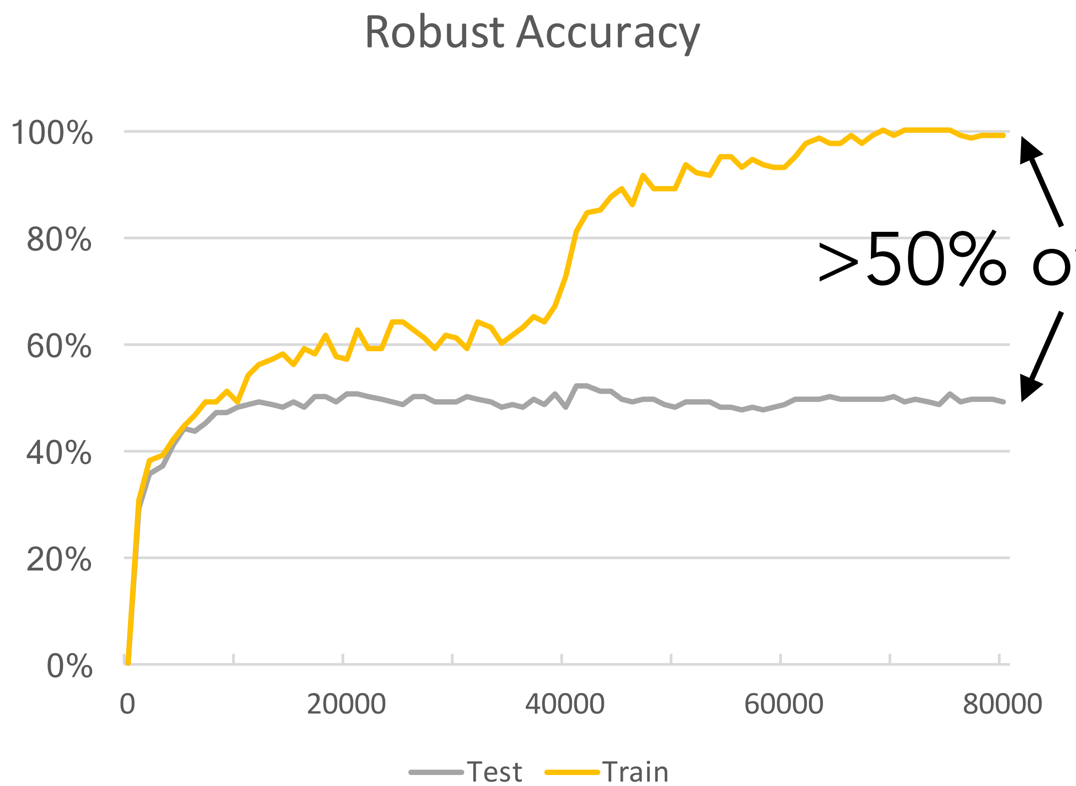
Robust generalization is hard

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



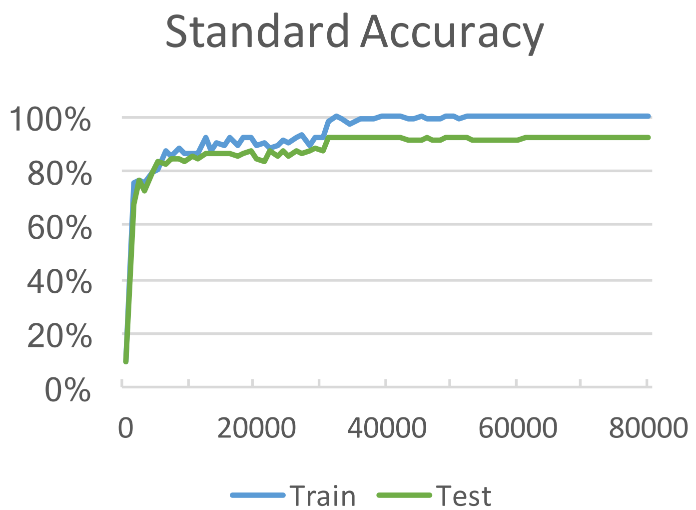
Robust generalization is hard

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



>50% overfitting

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



Doesn't happen "normally"

Is robust learning **fundamentally harder**?

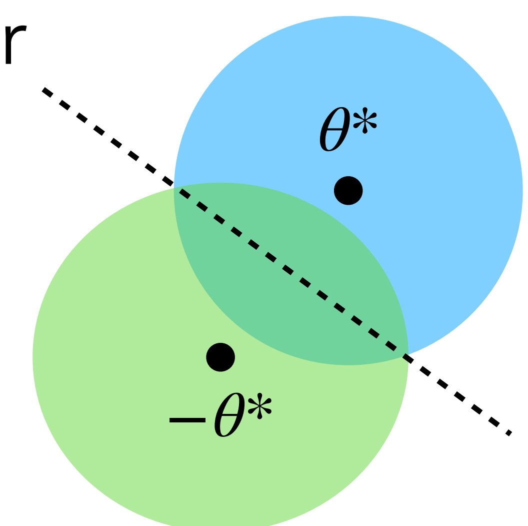
Robust generalization is hard

Theorem: The sample complexity of robust generalization can be significantly larger than that of "standard" generalization.

Specifically: There exists a **d**-dimensional distribution where:

→ A **single sample** is enough to learn a good (standard) classifier

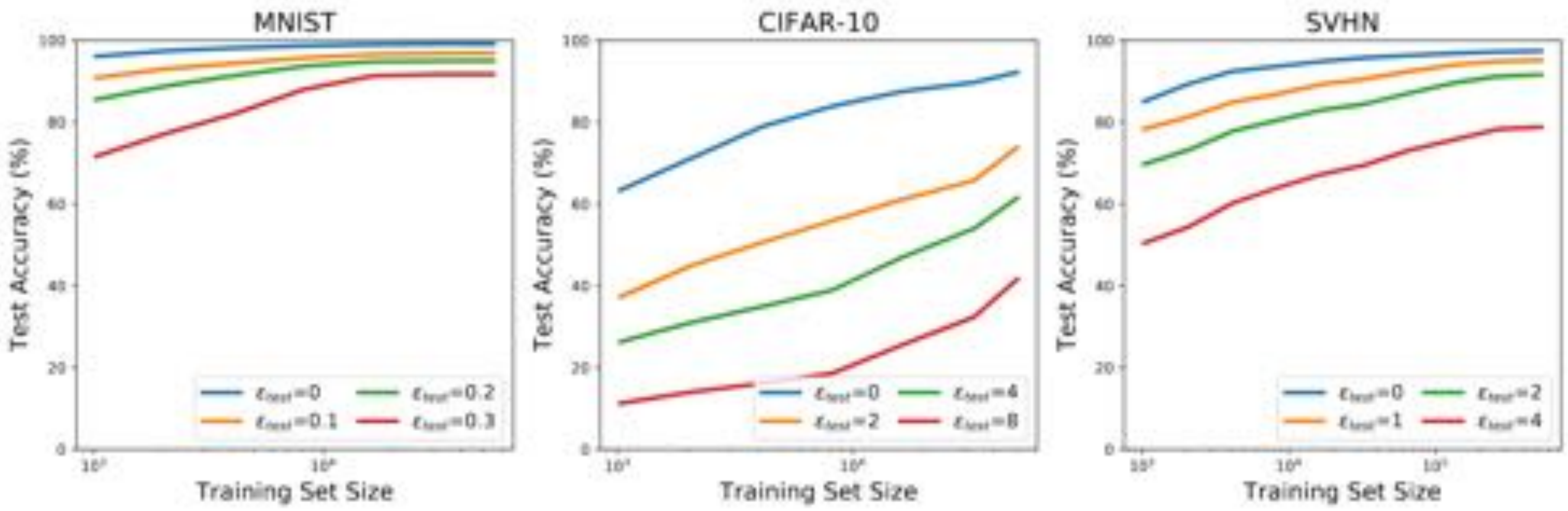
→ **But:** Need at least **$\Omega(\sqrt{d})$** samples for a robust classifier



Robust generalization is hard

Theorem: The sample complexity of robust generalization can be significantly larger than that of "standard" generalization.

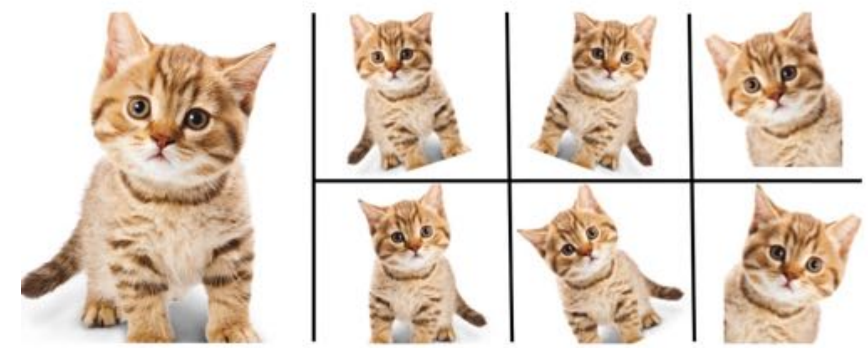
Empirically:



Does robustness improve accuracy?

Data augmentation: Train on random transformations of the input

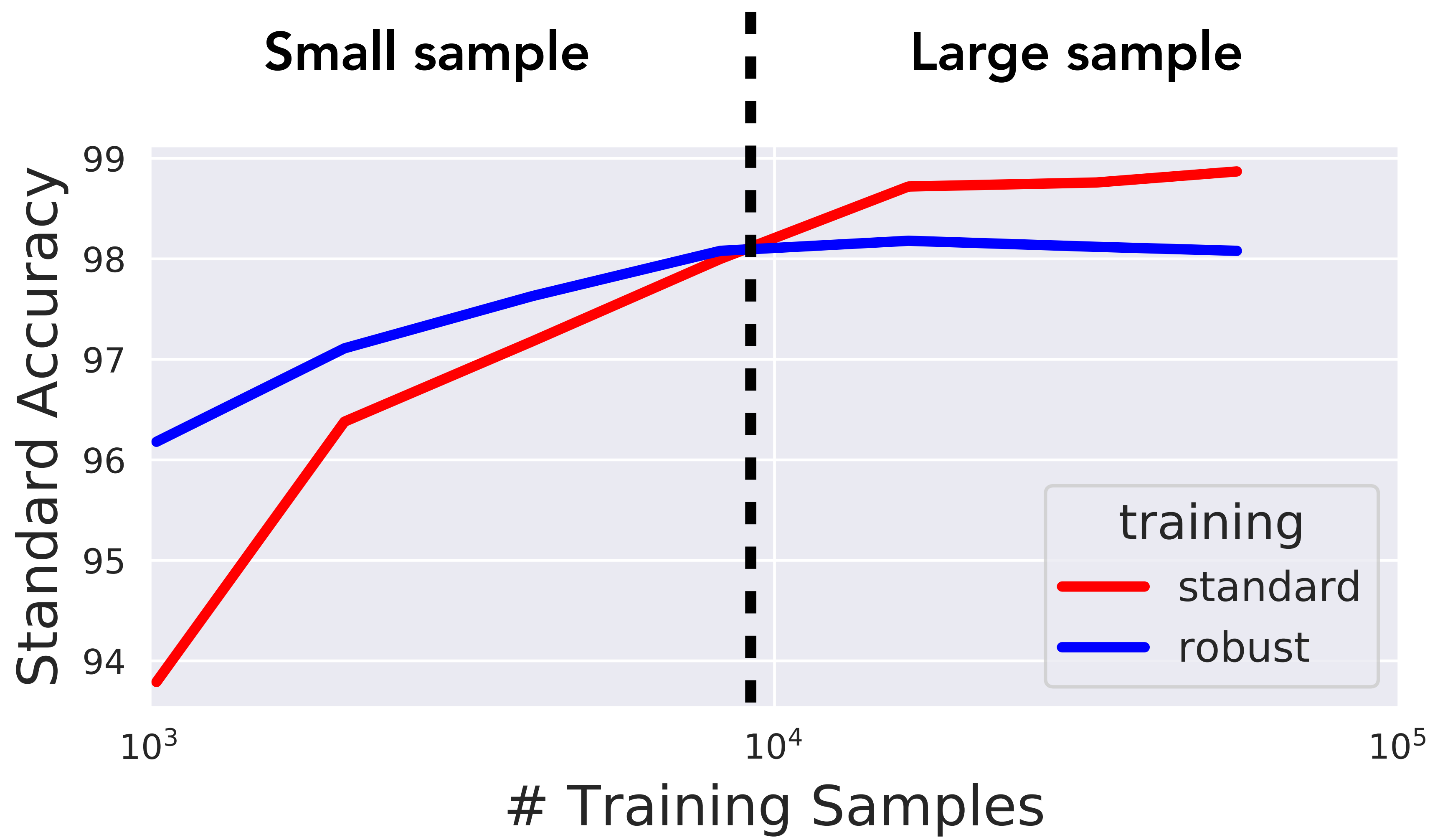
→ Significantly improves test accuracy.



Adversarial training ⇔ Augment with the "most helpful" example

Does adversarial training improve **standard accuracy**?

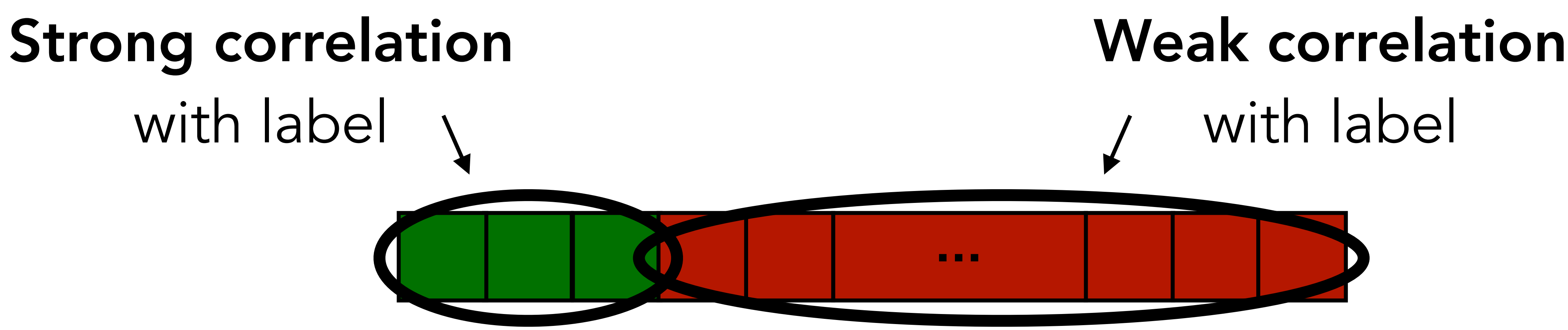
Does robustness improve accuracy?



Why are robust models **less accurate**?

Does robustness improve accuracy?

Theorem: There can exist an inherent trade-off between accuracy and robustness (no "free lunch").



Standard Training: use all the features to maximize accuracy

Adversarial Training: use **only** strong features **(lower accuracy)**

ML vs. “classical” security

Classical security exploits

Attackers use **unintended vulnerabilities** to manipulate system

Spectre: Side-effects of speculative execution

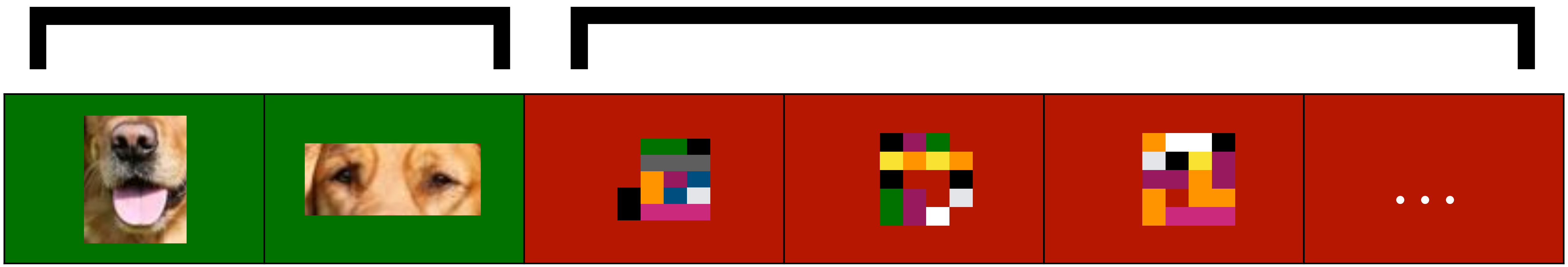
Heartbleed: Missing out-of-bounds read checks

"Correct" software should be unbreakable

ML security exploits

Robust features
Correlated with label
even with adversary

Non-robust features
Correlated with label on average,
but can be manipulated



Adversary manipulates input
features used for classification

Predictive non-robust features

Features small in L_2-norm

Accuracy	CIFAR10	R. ImageNet
Standard	95%	97%
Non-robust features	44%	64%

Other examples of **unintuitive** features

Linear directions
[Jetley et al 2018]

High-frequency patterns
[Yin et al 2019]

Texture
[Geirhos et al 2019]

Back to adversarial examples

Non-robust features can be **quite predictive**

We train classifiers to **maximize accuracy**:
No wonder they utilize non-robust features

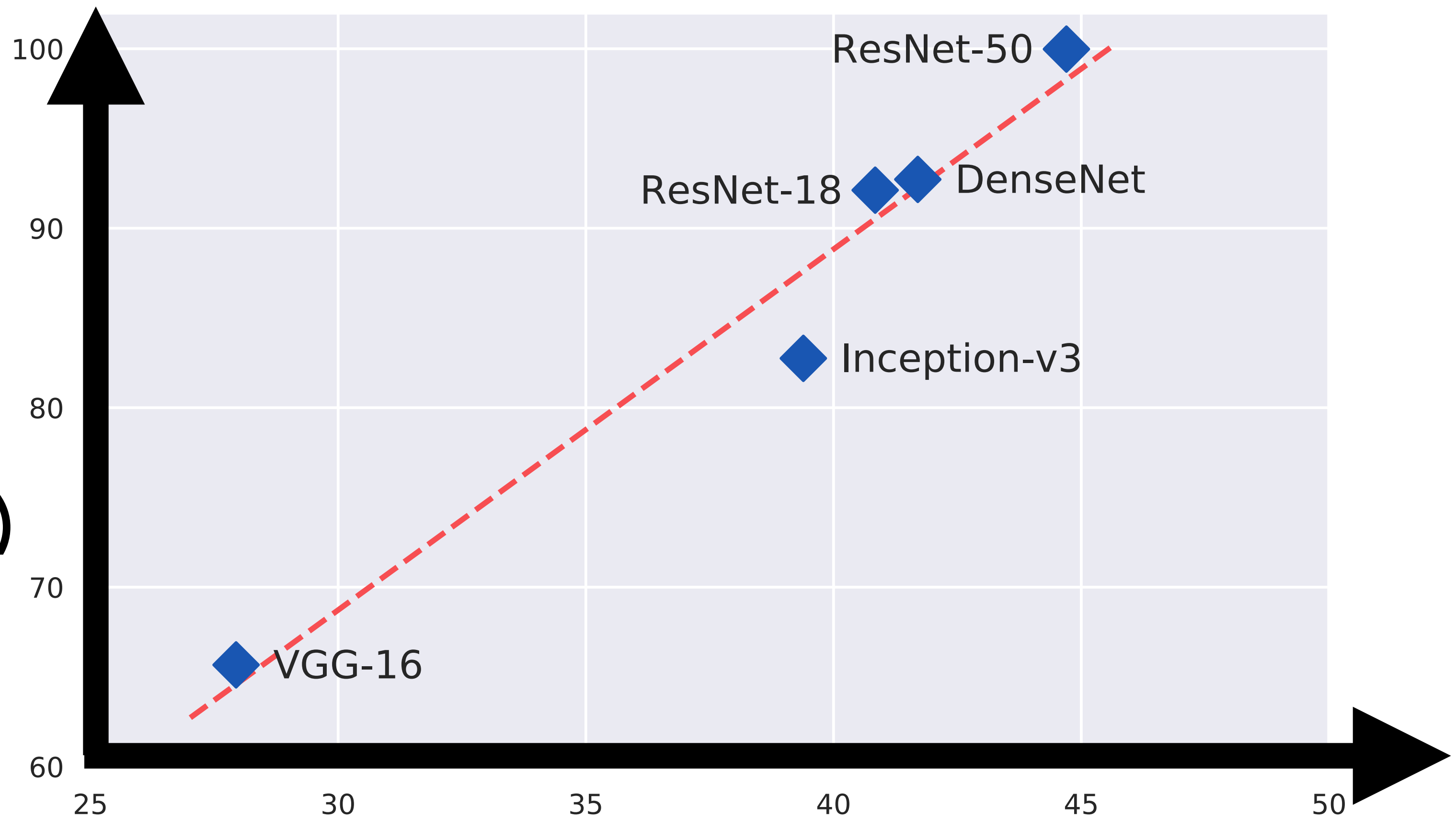
Relying on non-robust features **directly leads**
to adversarial vulnerability

Thus: Adversarial examples are not bugs, they are features

Consequences

Transferability: Models learn similar non-robust features

Adversarial Transferability (ResNet-50→X)



Test accuracy of X trained on non-robust features from ResNet-50

Consequences

Dataset robustification: Removing non-robust features can improve **standard** classifiers

Training set

frog

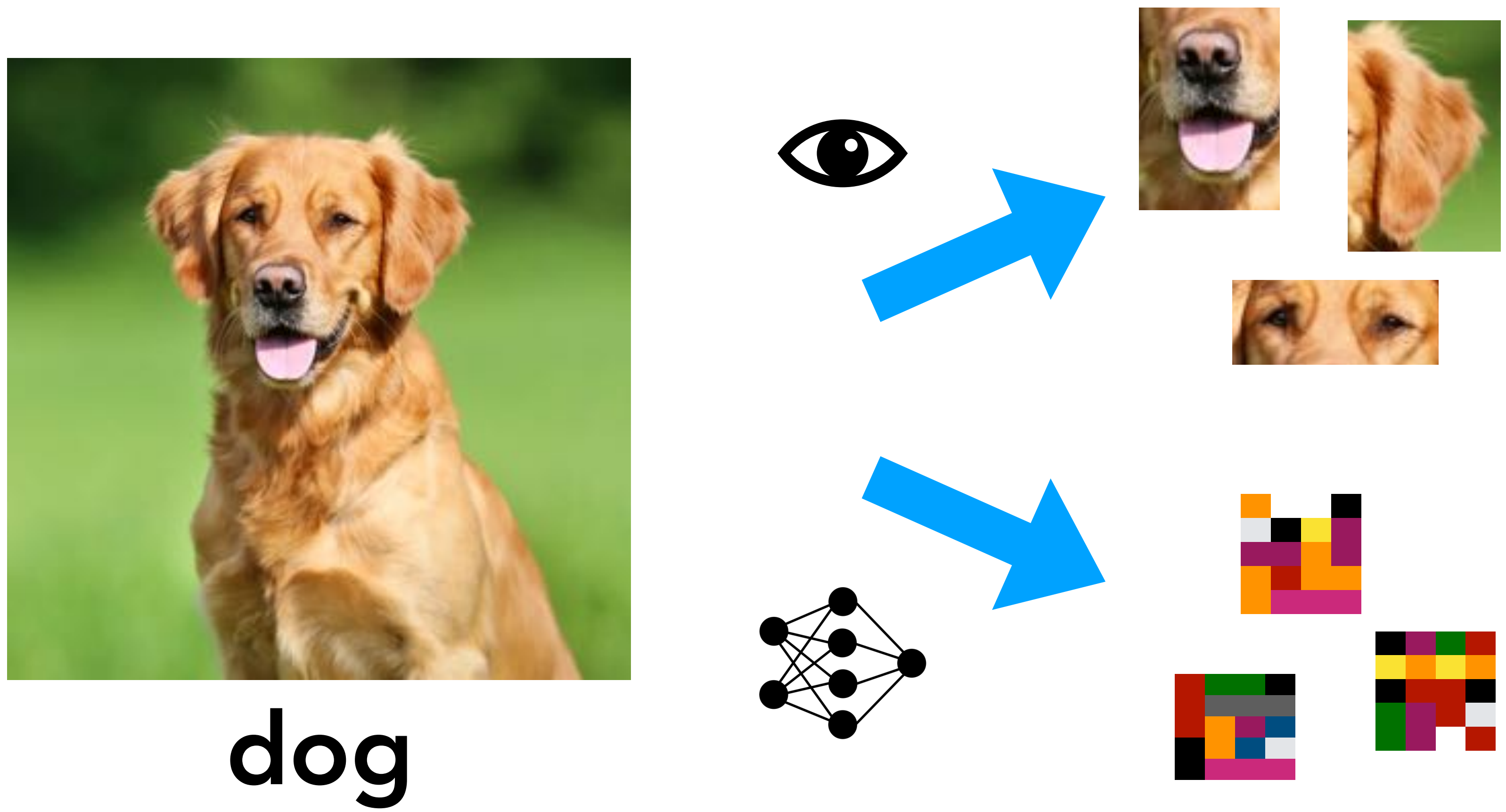
Restrict to features of robust model

New training set

"Robustified" frog

Standard training yields **robust classifiers**

Humans vs ML Models



Equally valid classification methods

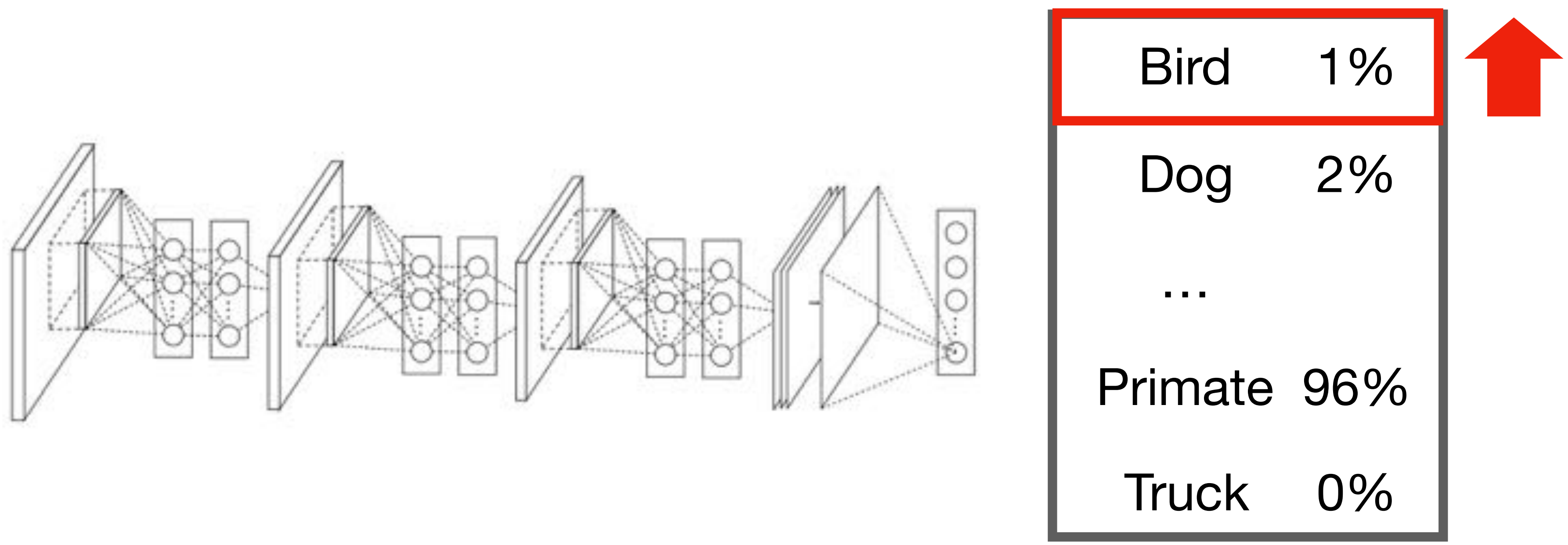
We need to **explicitly enforce robustness**

Robustness beyond security:

Robust models are more human-aligned

Input Manipulation

Key Idea: Manipulate class scores for **robust models**



Class maximization introduces salient features

Downstream applications

Image Generation

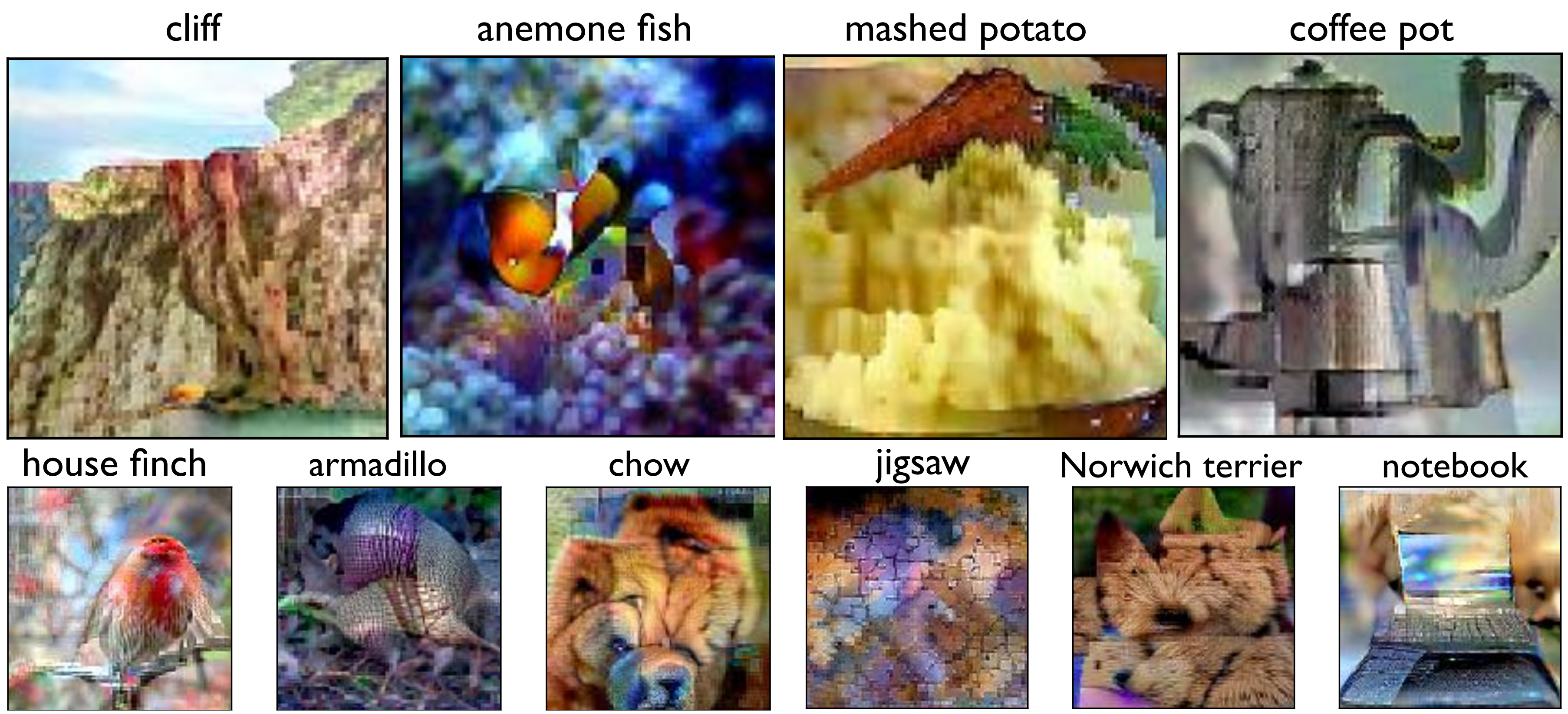
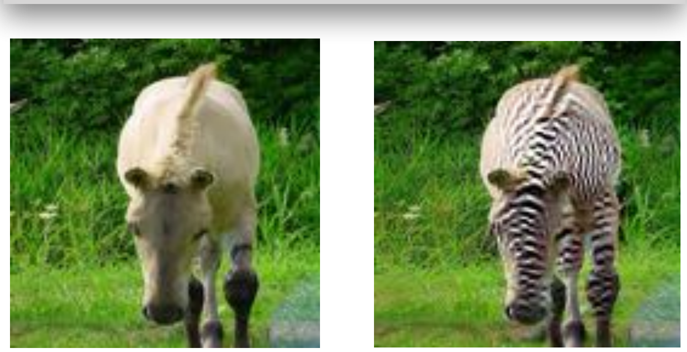
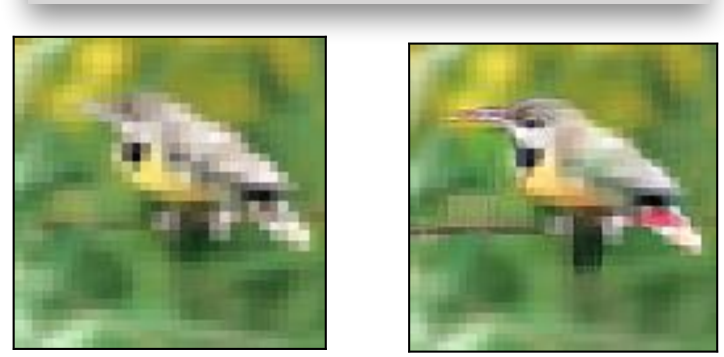


Image Translation



Superresolution

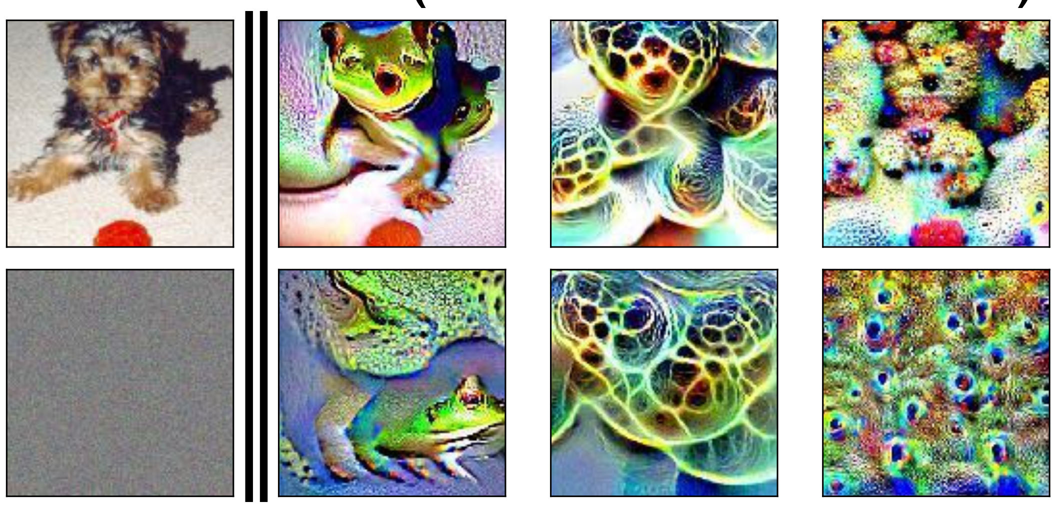


Inpainting

Better representations

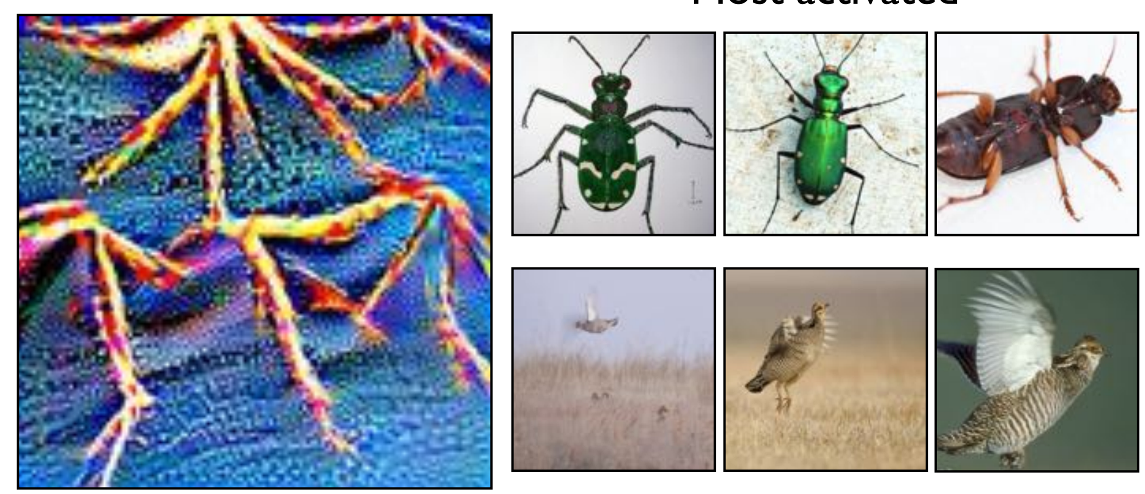
Direct feature visualization

Seed Max(different coordinates)



(insect legs)

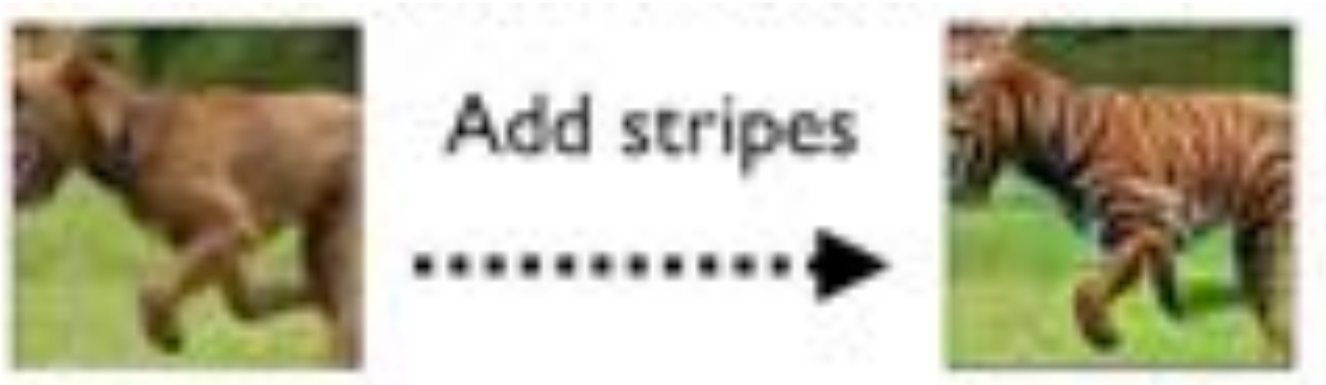
Most activated



Maximized from noise

Least activated

Feature manipulation



Interpolation

Conclusions

Takeaways

ML models are really **brittle**

Brittleness can arise from **non-robust features**

Robust optimization **can lead to robust models**

Robustness as a tool for **human-aligned** models

Future directions

More **robust models**

Different **perturbation sets**

More comprehensive **theoretical models**

Further **exploration of robust models**

`robustness`

`gradsci.org`