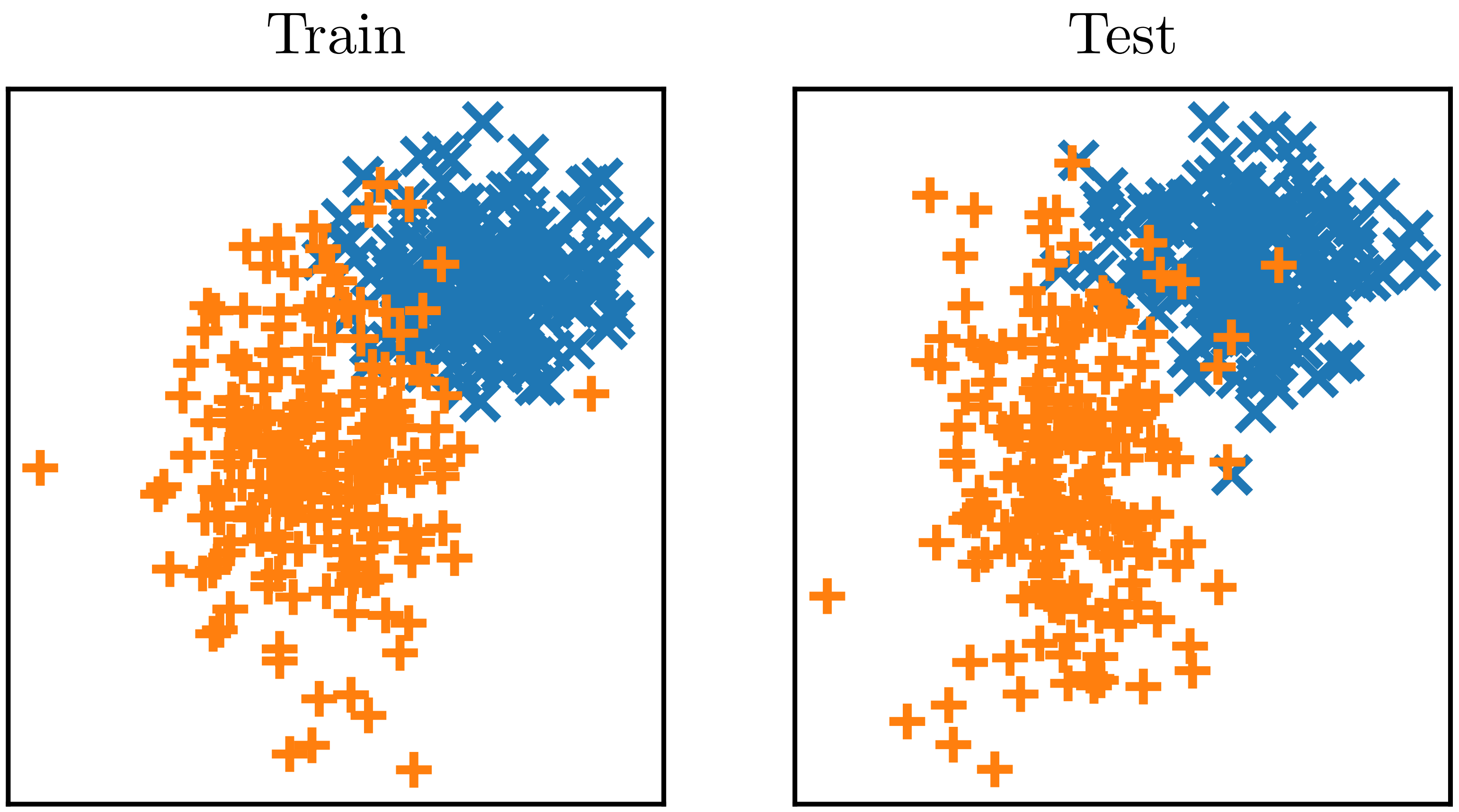
Defense Against the Dark Arts: An overview of adversarial example security research and future research directions

Ian Goodfellow, Staff Research Scientist, Google Brain

1st IEEE Deep Learning and Security Workshop, May 24, 2018

I.I.D. Machine Learning



I: Independent

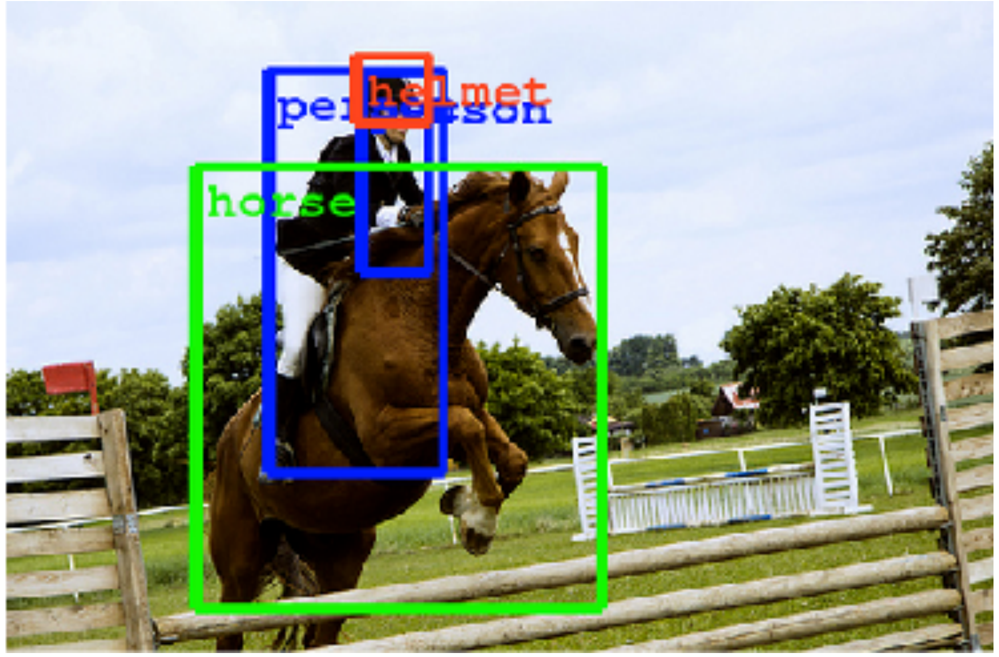
I: Identically

D: Distributed

All train and test examples drawn independently from same distribution

(Goodfellow 2018)

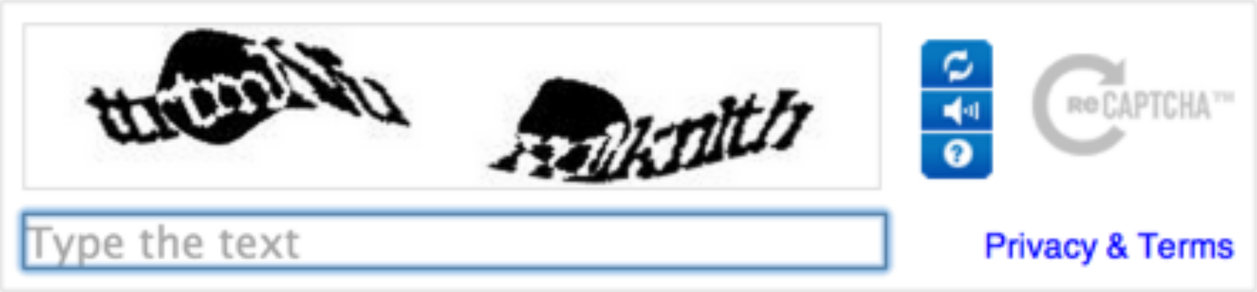
ML reached “human-level performance” on many IID tasks circa 2013



(Szegedy et al, 2014)

...recognizing objects and faces....

(Taigmen et al, 2013)

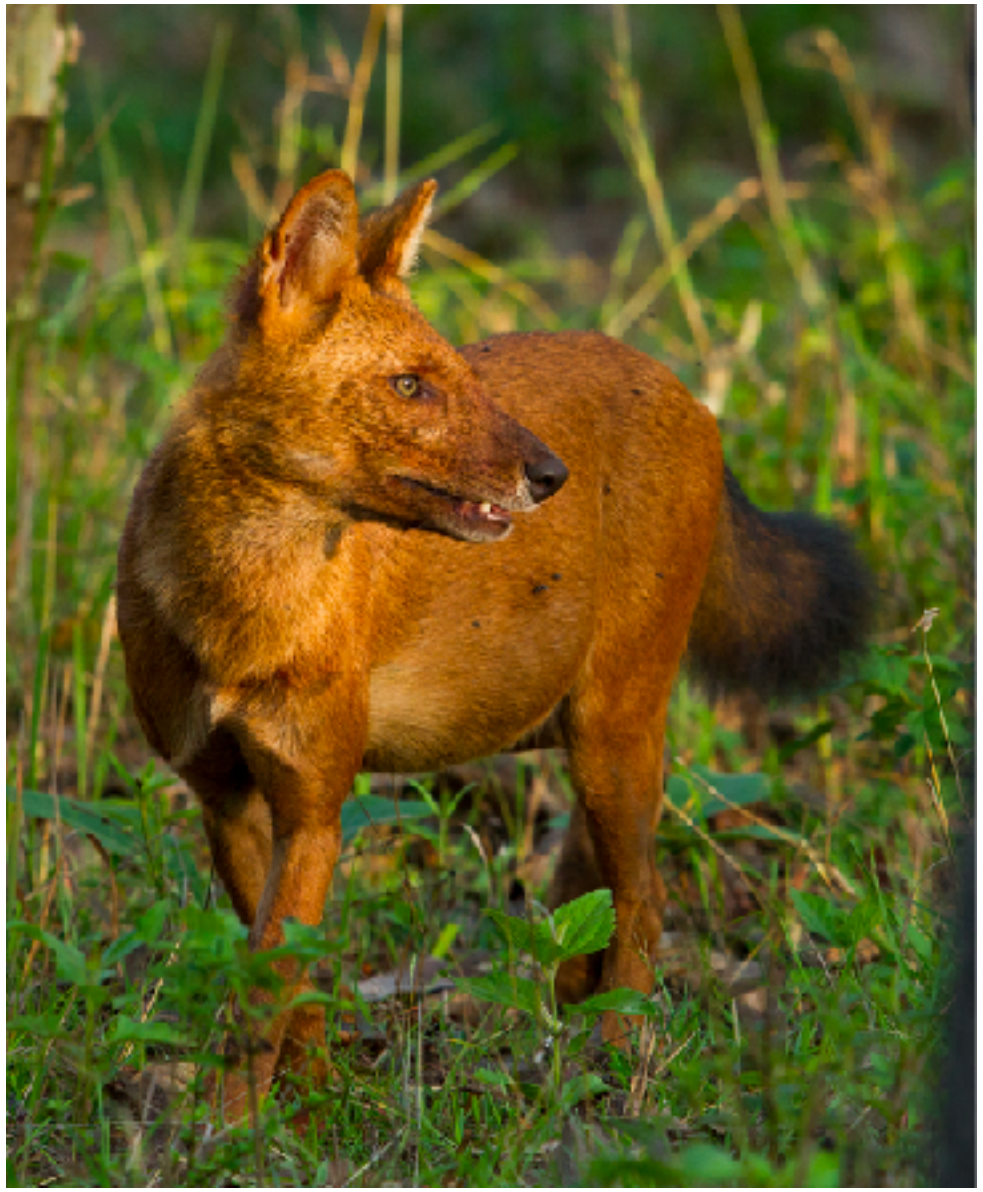


(Goodfellow et al, 2013)

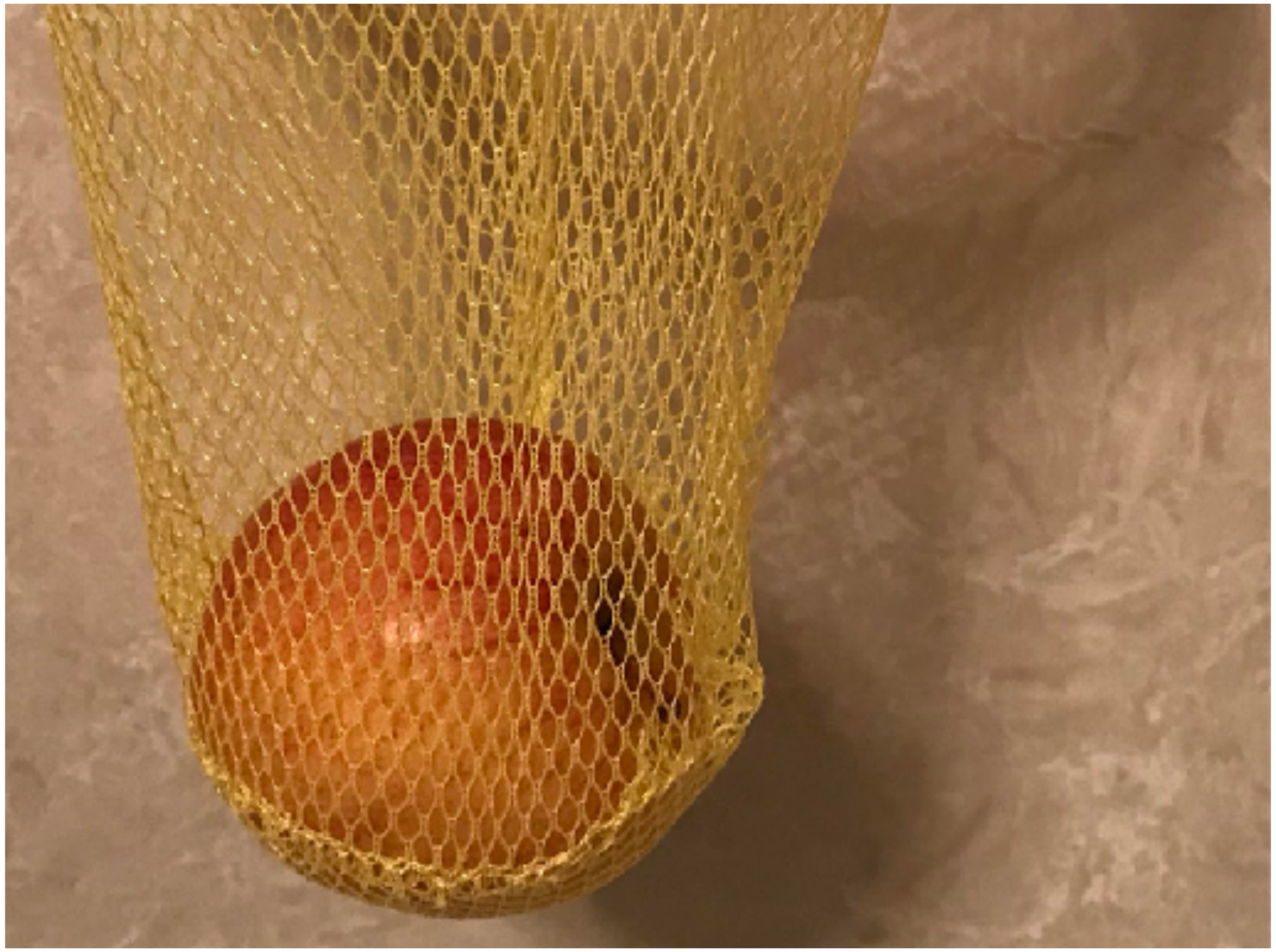
...solving CAPTCHAS and reading addresses...

(Goodfellow et al, 2013)

Caveats to “human-level” benchmarks



Humans are not very good at some parts of the benchmark

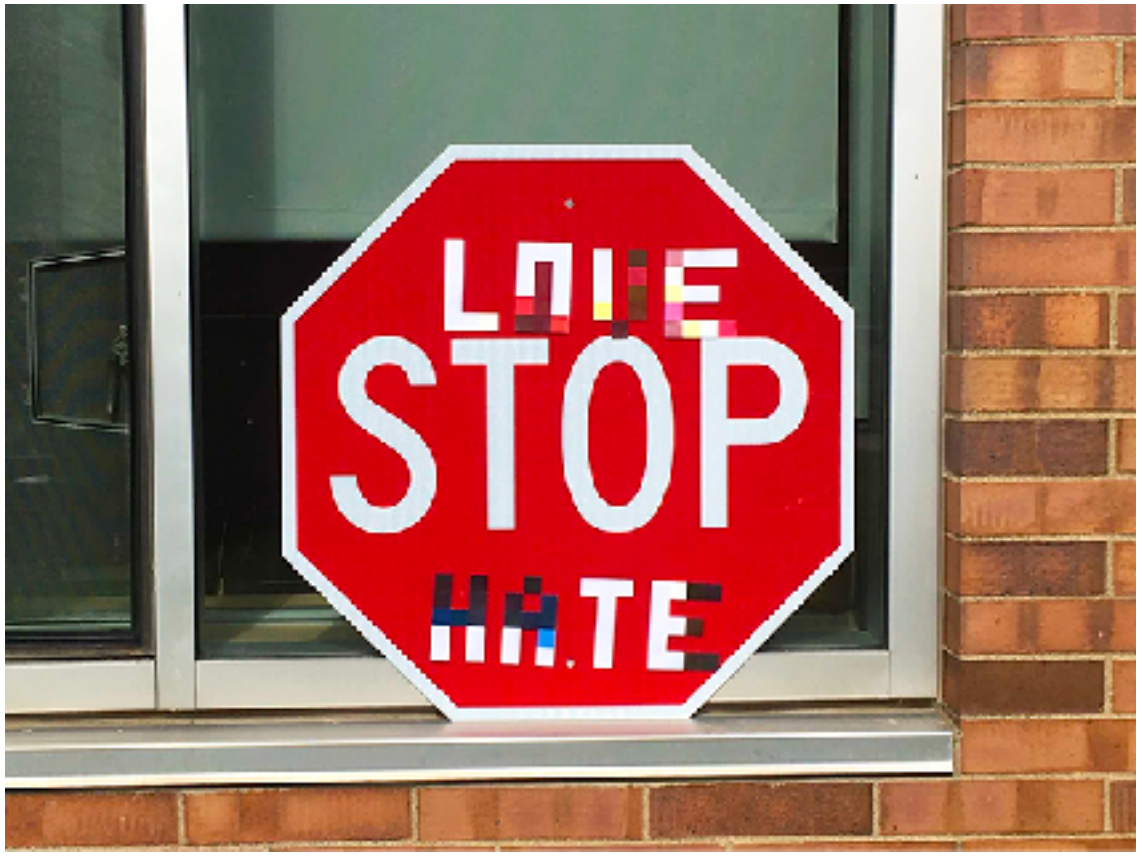


The test data is not very diverse. ML models are fooled by natural but unusual data.

(Goodfellow 2018)

Security Requires Moving Beyond I.I.D.

- Not identical: attackers can use unusual inputs



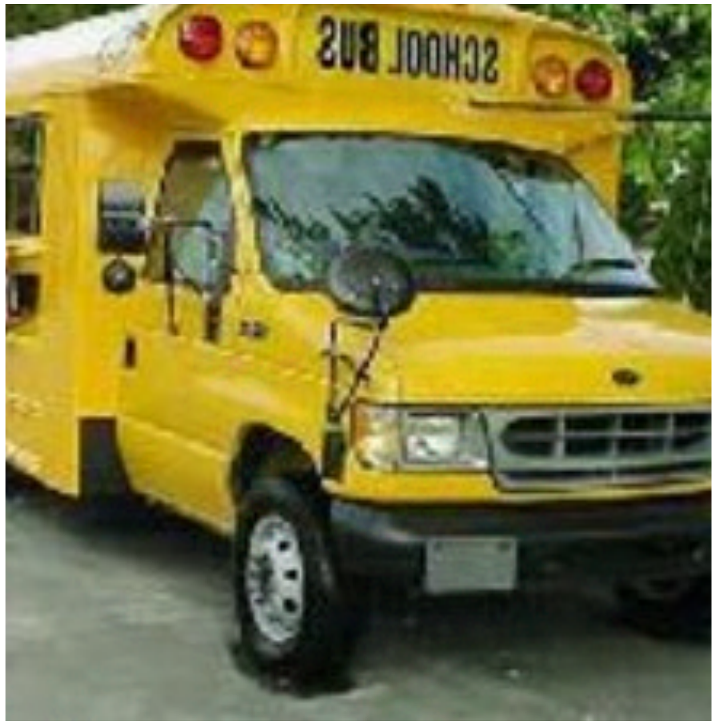
(Eykholt et al, 2017)

- Not independent: attacker can repeatedly send a single mistake (“test set attack”)

(Goodfellow 2018)

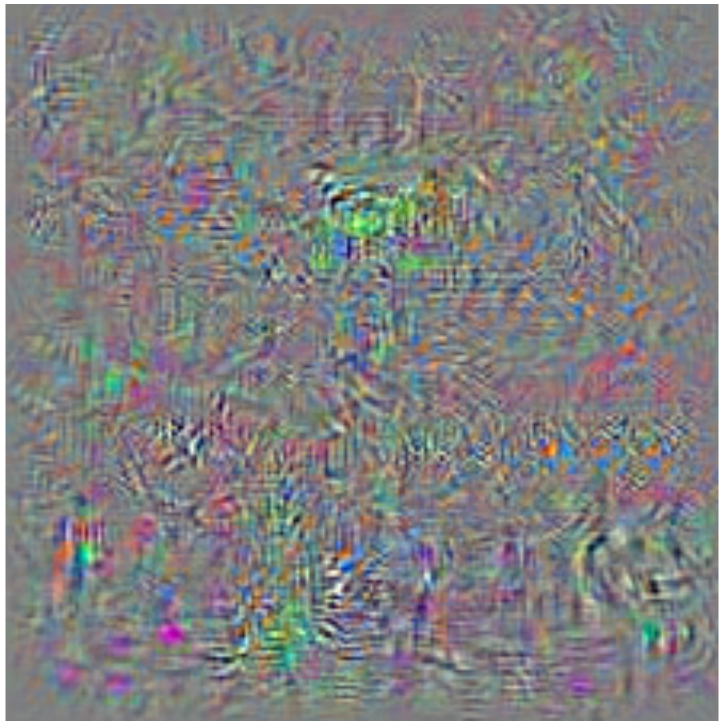
Good models make surprising mistakes in non-IID setting

“Adversarial examples”



Schoolbus

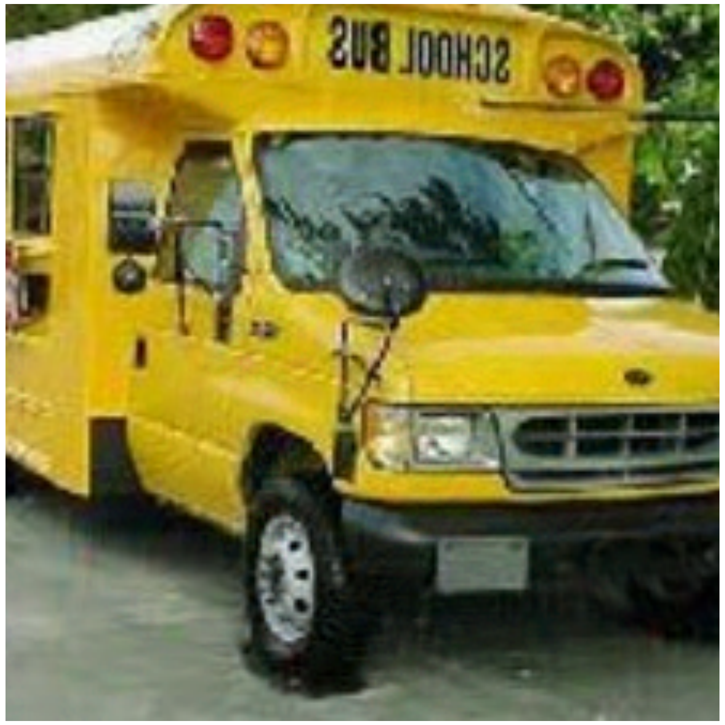
+



Perturbation

(rescaled for visualization)

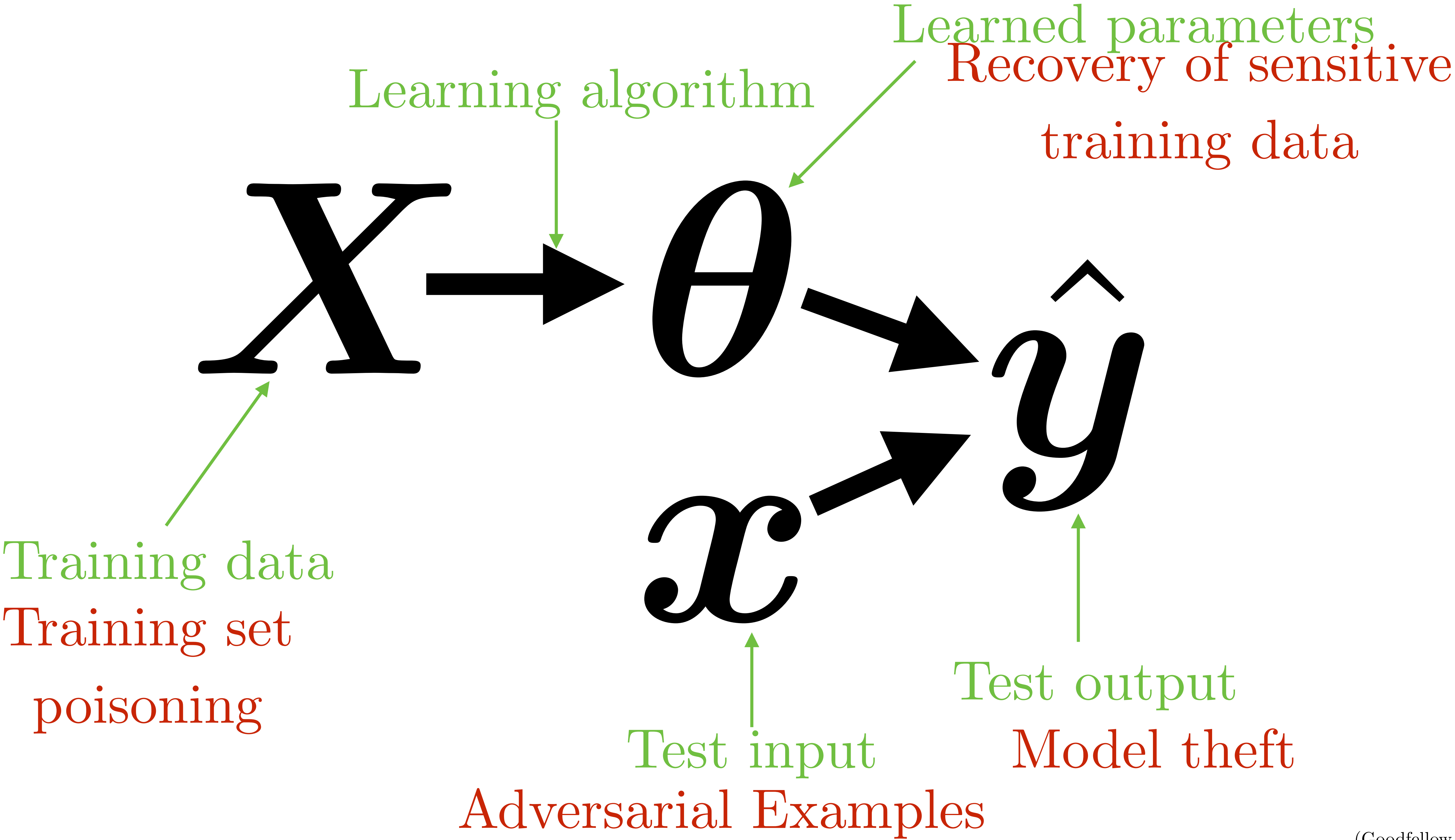
=



Ostrich

(Szegedy et al, 2013)

Attacks on the machine learning pipeline



Learning algorithm

Learned parameters

Recovery of sensitive training data

Training data

Training set poisoning

Test input

Adversarial Examples

Test output

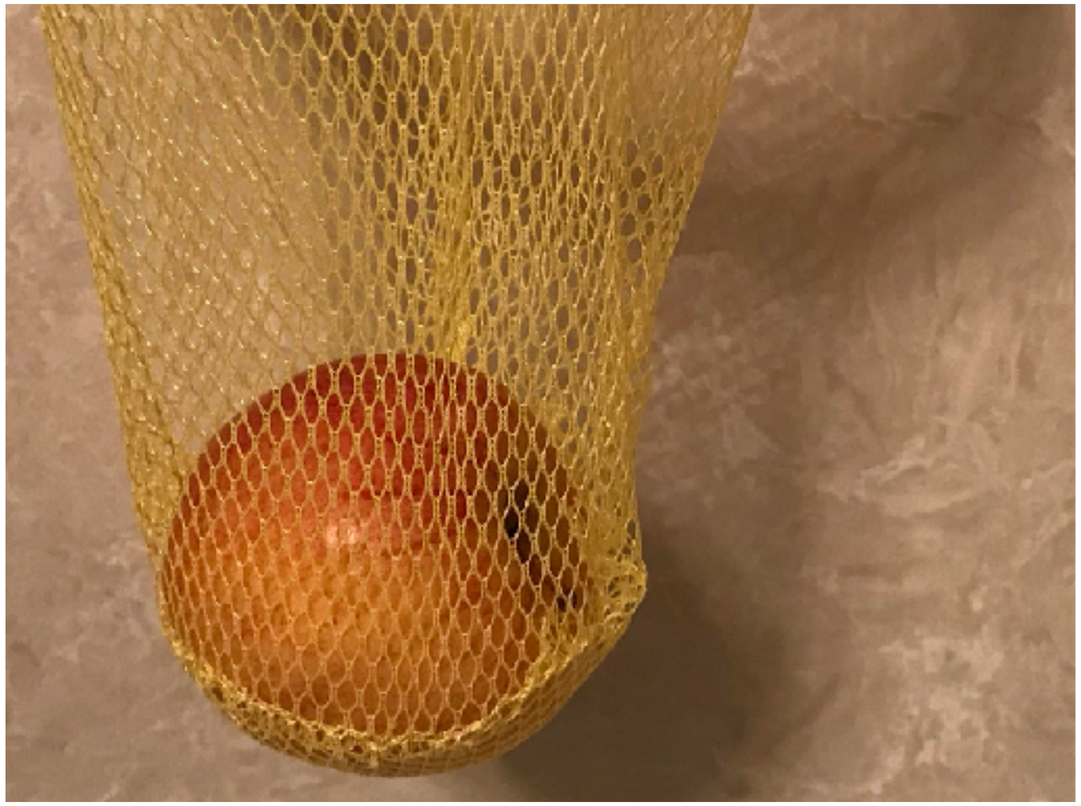
Model theft

(Goodfellow 2018)

Definition

“Adversarial examples are inputs to machine learning models that an attacker has intentionally designed to cause the model to make a mistake”

(Goodfellow et al 2017)



(Goodfellow 2018)

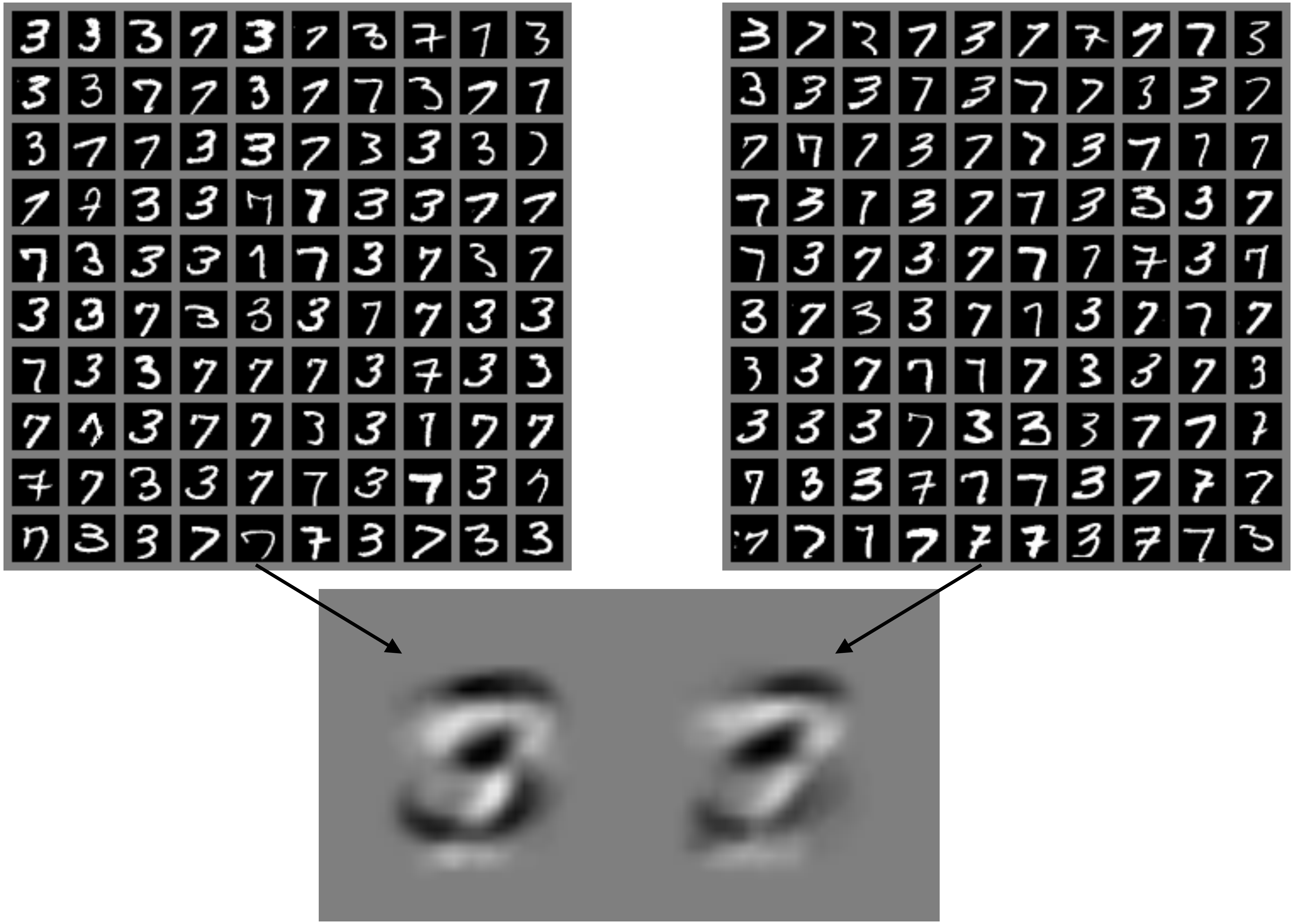
Define a game

- Define an action space for the defender
- Define an action space for an attacker
- Define cost function for defender
- Define cost function for attacker
    - Not necessarily minimax.
    - Targeted vs untargeted

Fifty Shades of Gray Box Attacks

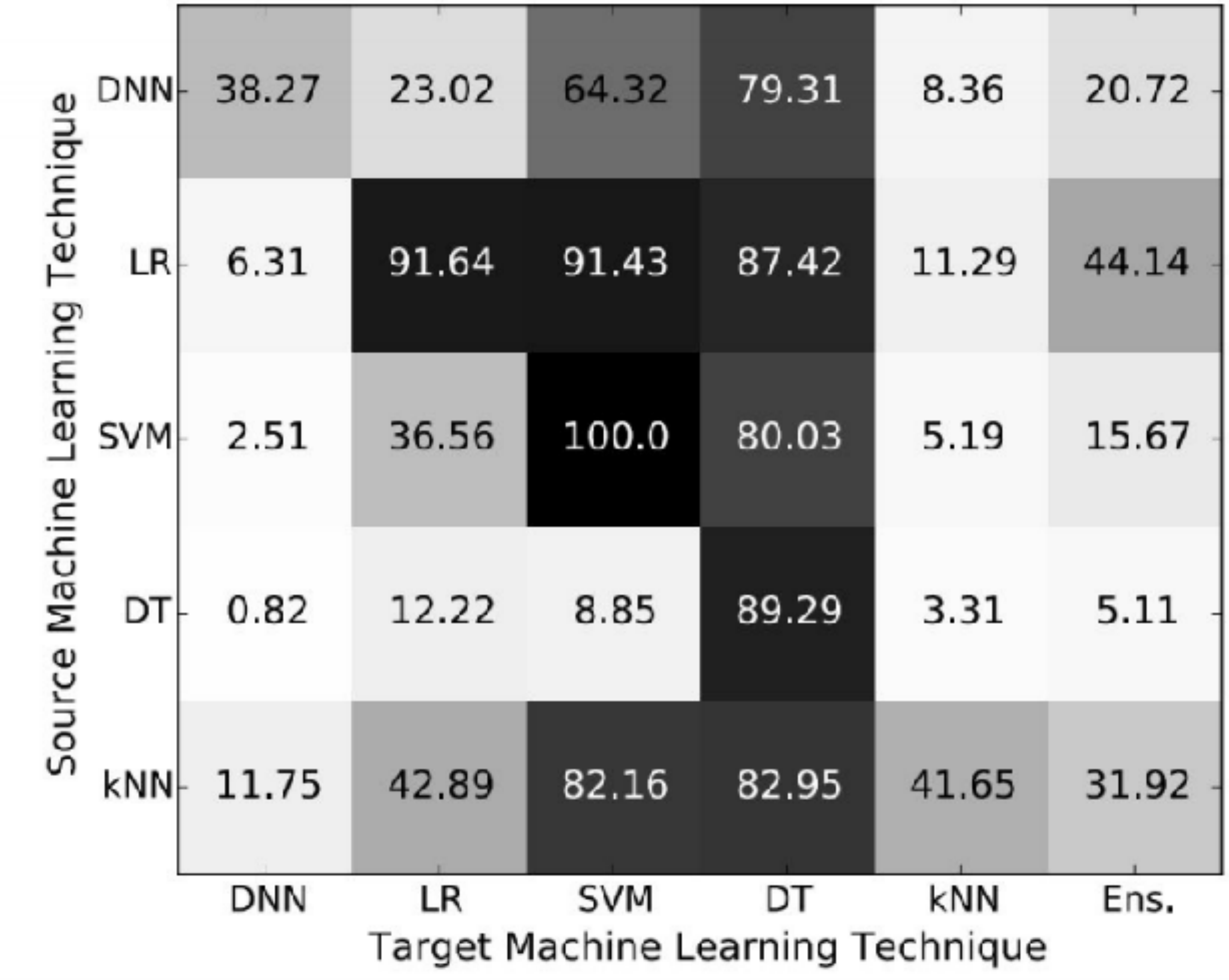
- Does the attacker go first, and the defender reacts?
    - This is easy, just train on the attacks, or design some preprocessing to remove them
- If the defender goes first
    - Does the attacker have full knowledge? This is “white box”
    - Limited knowledge: “black box”
        - Does the attacker know the task the model is solving (input space, output space, defender cost) ?
        - Does the attacker know the machine learning algorithm being used?
        - Details of the algorithm? (Neural net architecture, etc.)
        - Learned parameters of the model?
        - Can the attacker send “probes” to see how the defender processes different test inputs?
            - Does the attacker observe just the output class? Or also the probabilities?

Cross-model, cross-dataset generalization



(Goodfellow 2017)

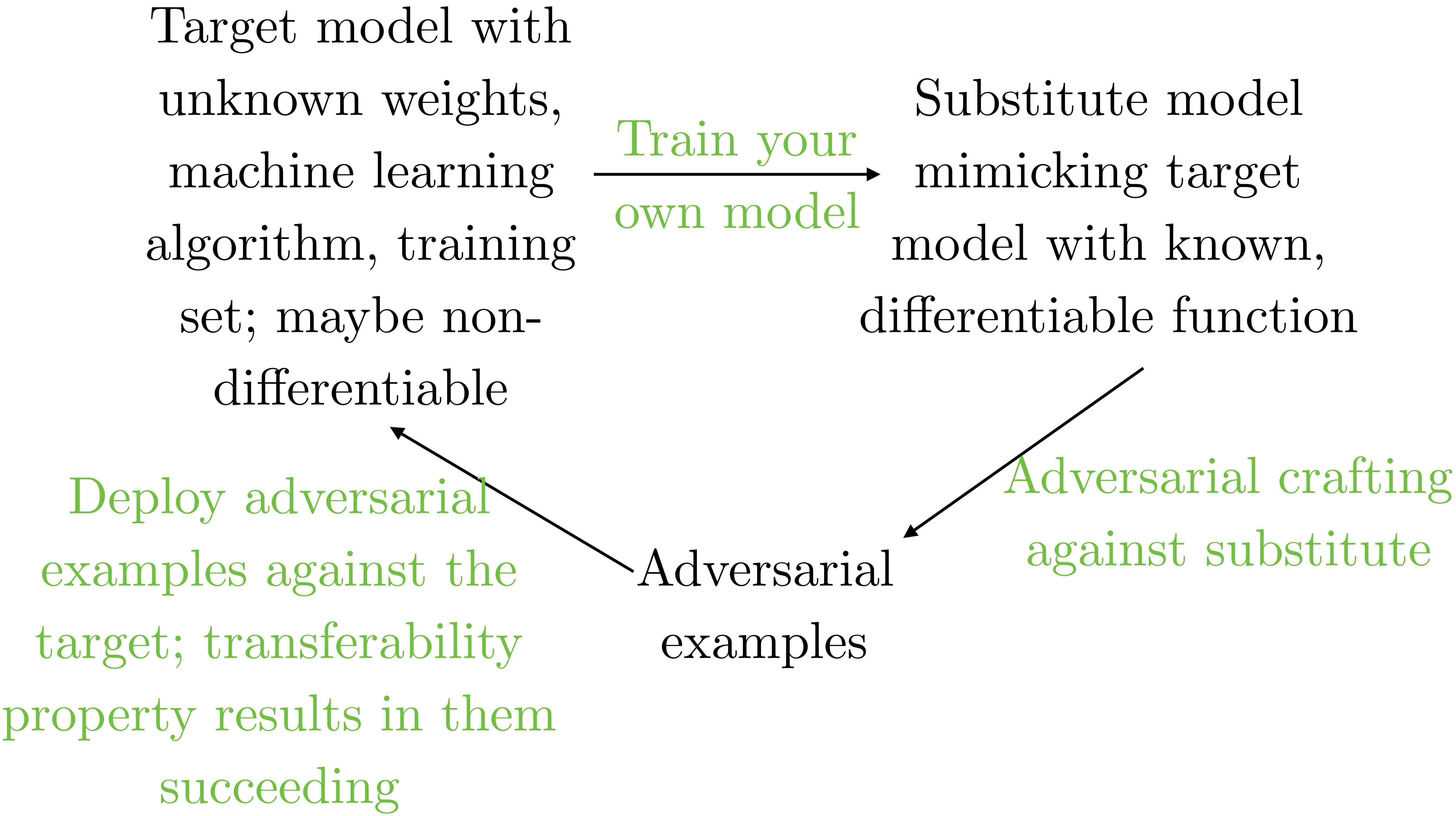
Cross-technique transferability



	DNN	LR	SVM	DT	kNN	Ens.
DNN	38.27	23.02	64.32	79.31	8.36	20.72
LR	6.31	91.64	91.43	87.42	11.29	44.14
SVM	2.51	36.56	100.0	80.03	5.19	15.67
DT	0.82	12.22	8.85	89.29	3.31	5.11
kNN	11.75	42.89	82.16	82.95	41.65	31.92

(Papernot 2016)

Transfer Attack



Target model with unknown weights, machine learning algorithm, training set; maybe non-differentiable

Train your own model

Substitute model mimicking target model with known, differentiable function

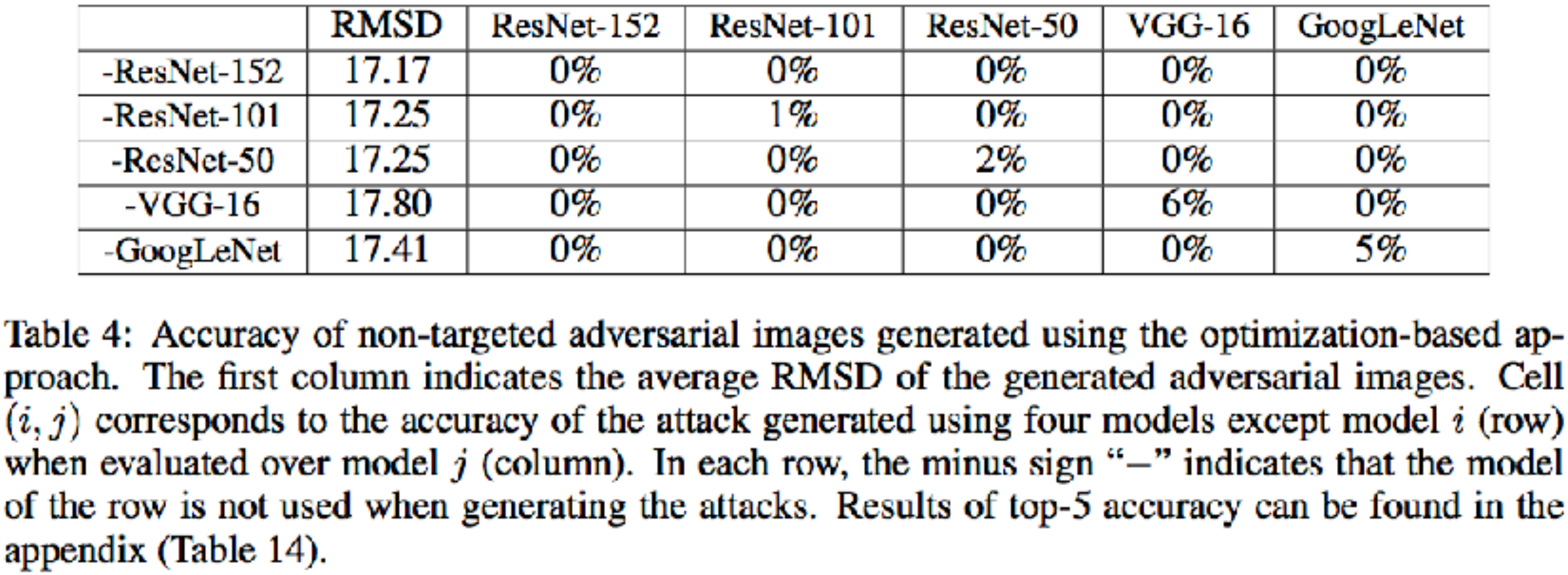
Adversarial crafting against substitute

Adversarial examples

Deploy adversarial examples against the target; transferability property results in them succeeding

(Goodfellow 2018)

Enhancing Transfer With Ensembles



	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%

Table 4: Accuracy of non-targeted adversarial images generated using the optimization-based approach. The first column indicates the average RMSD of the generated adversarial images. Cell (i, j) corresponds to the accuracy of the attack generated using four models except model i (row) when evaluated over model j (column). In each row, the minus sign "−" indicates that the model of the row is not used when generating the attacks. Results of top-5 accuracy can be found in the appendix (Table 14).

(Liu et al, 2016)

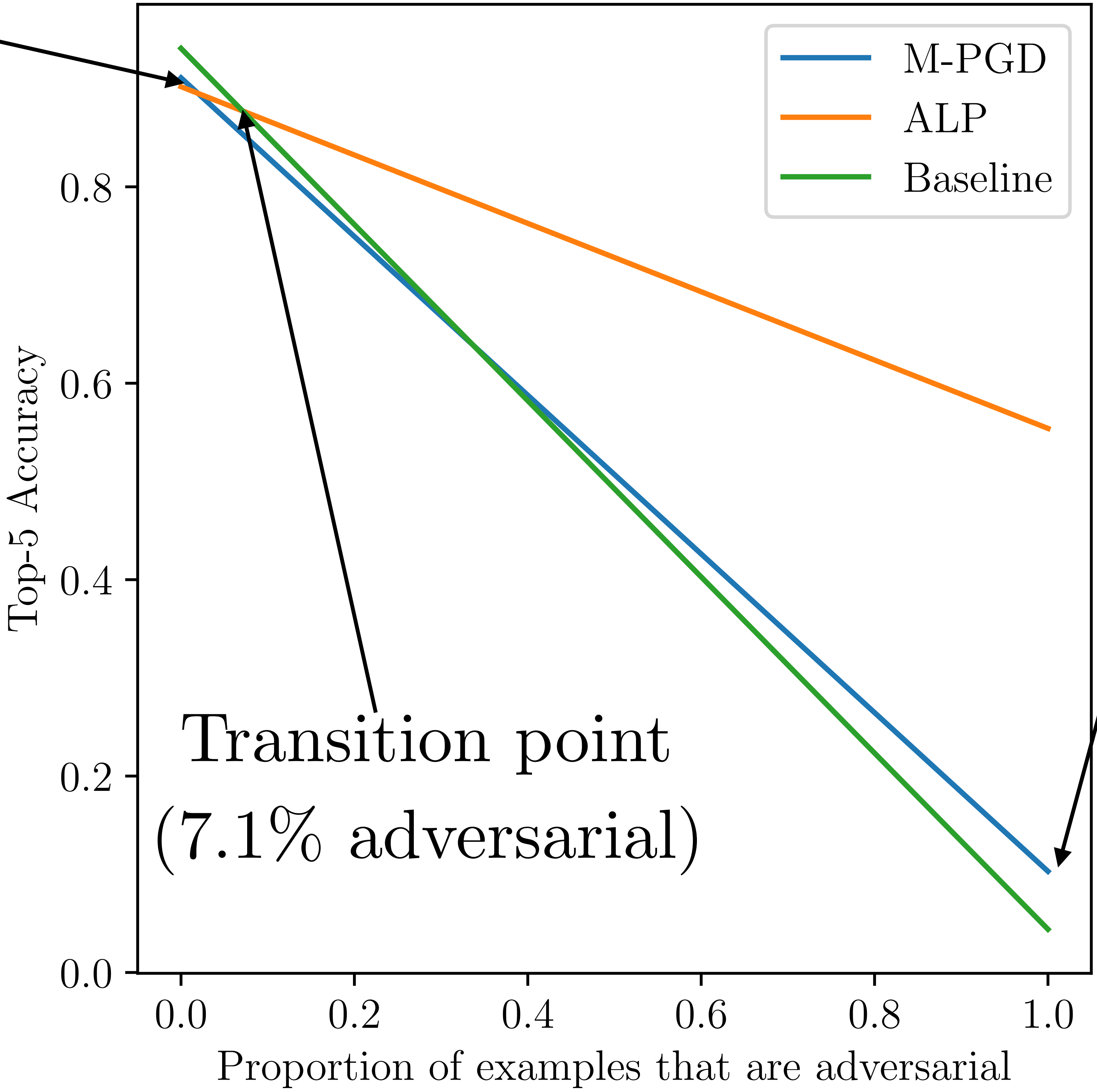
Norm Balls: A Toy Game

- How to benchmark performance on points that are not in the dataset and not labeled?
- Propagate labels from nearby labeled examples
- Attacker action:
    - Given a clean example, add a norm-constrained perturbation to it
- The *drosophila* of adversarial machine learning
- Interesting for *basic research* purposes because of its clarity and difficulty
- Not relevant for most practical purposes: not a *current, applied* security problem
- In my view, this shouldn't be primarily about human perception

Who goes first?

- Attacker goes first:
    - Defender trains on the attacks. Usually the defender wins.
    - Not much more interesting than standard dataset augmentation
- Defender goes first:
    - Attacker is *adaptive* / *reactive*
    - Extremely difficult. Main reason this topic is unsolved.

Tradeoff



Accuracy on clean examples

Transition point (7.1% adversarial)

Accuracy on adversarial examples

(Goodfellow 2018)

Gradient Masking

- Some defenses look like they work because they break gradient-based white box attacks

- But then they don't break black box attacks (e.g., adversarial examples made for other models)

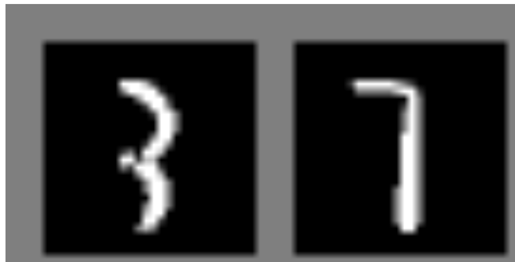
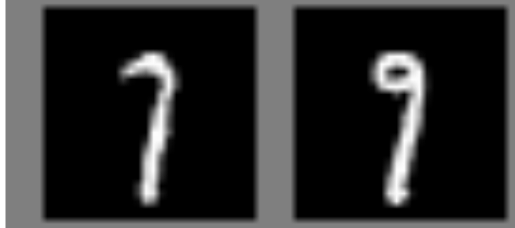
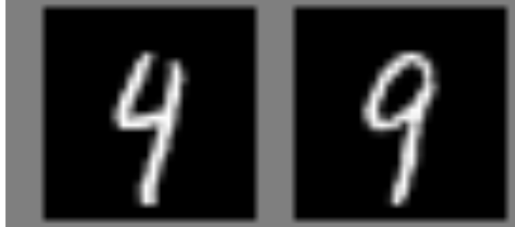
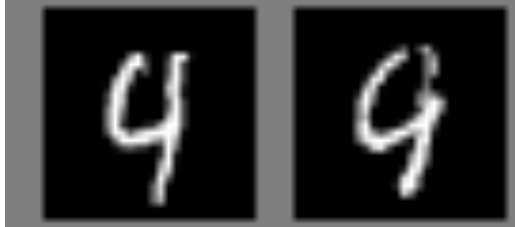
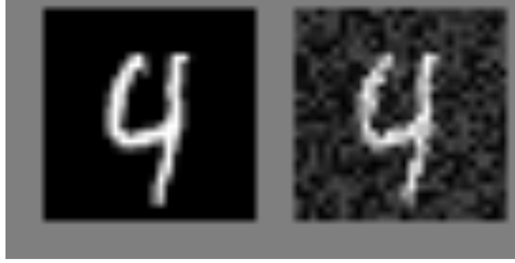
- The defense denies the attacker access to a useful gradient but does not actually make the *decision boundary* secure

- This is called *gradient masking*

(Goodfellow 2018)

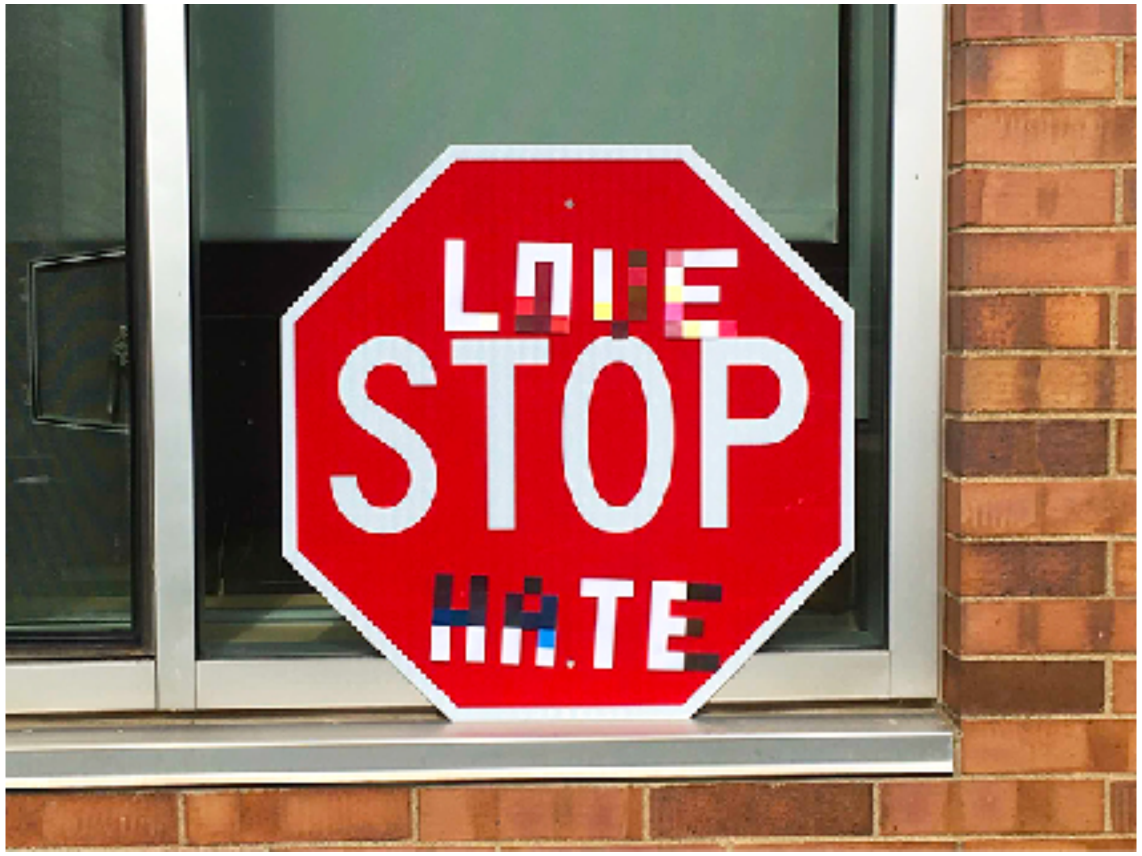
Why not to use L2

Experiments excluding MNIST 1s, many of which look like 7s

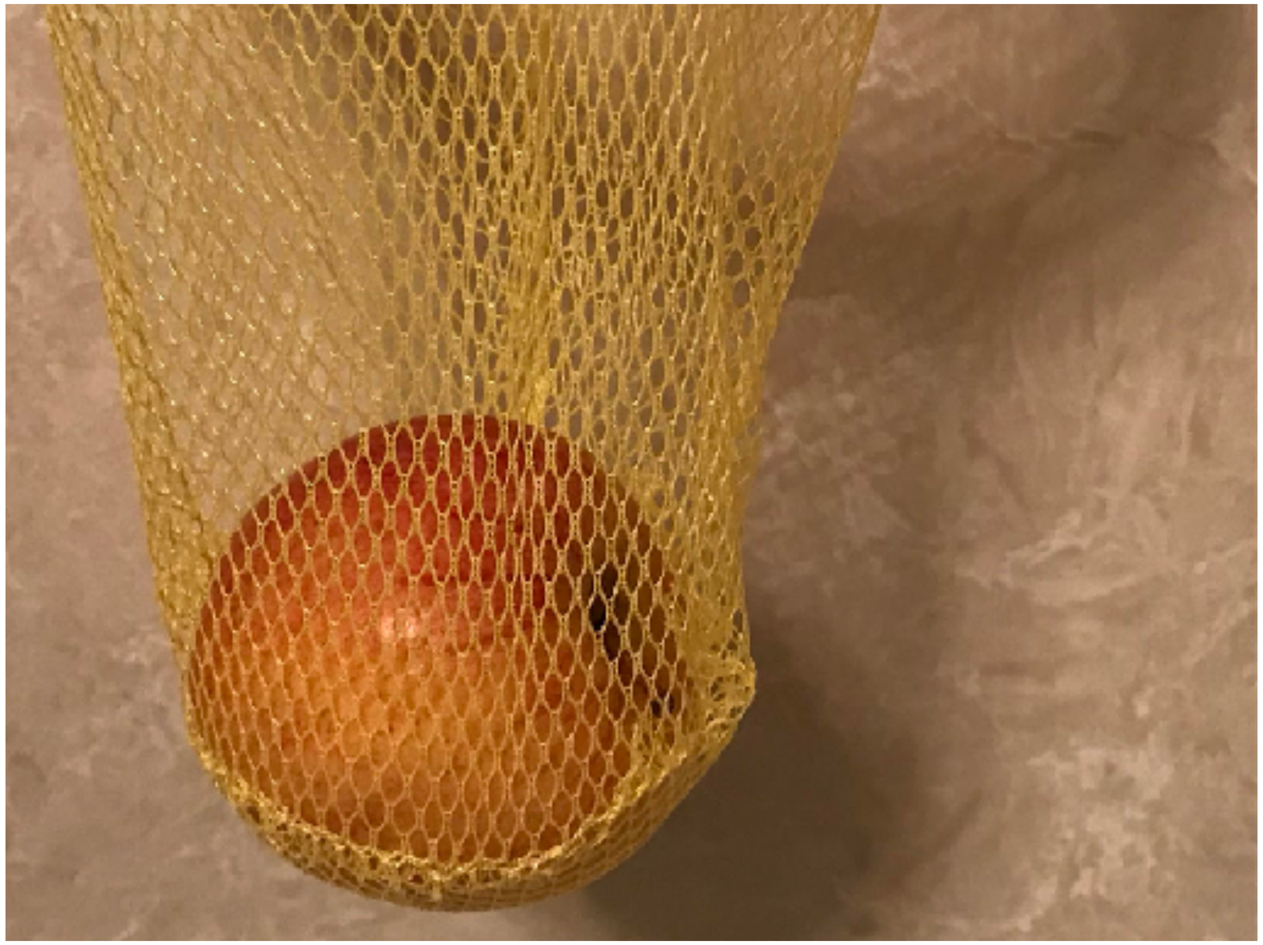
	Pair	Diff	$L0$	$L1$	$L2$	$L\infty$
Nearest $L0$			63	35.0	4.86	1.0
Nearest $L1$			91	19.9	3.21	.996
Nearest $L2$			110	21.7	2.83	1.0
Nearest $L\infty$			121	34.0	3.82	.76
Clipped Random uniform			784	116.0	4.8	.3

(Goodfellow 2018)

Real Attacks Will not be in the Norm Ball

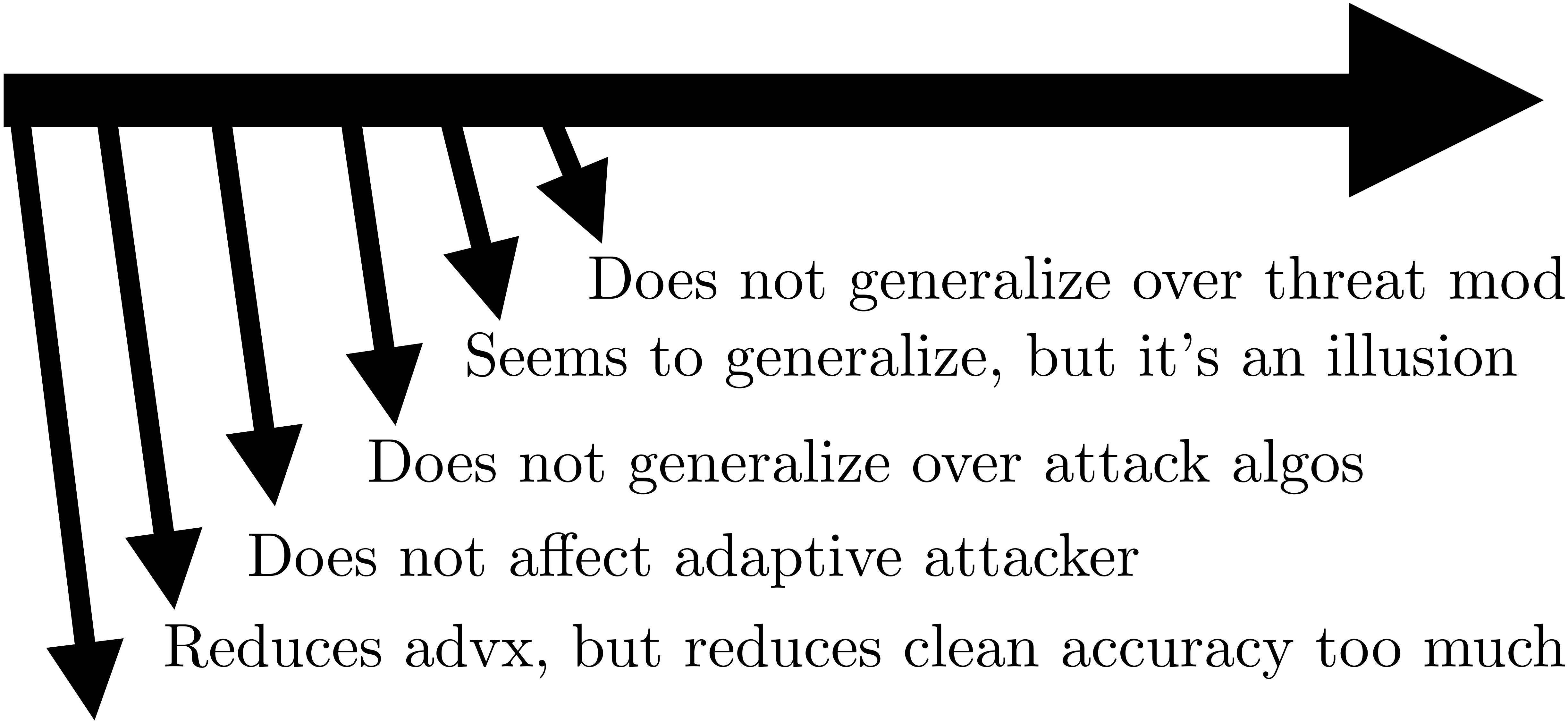


(Eykholt et al, 2017)



(Goodfellow 2018)

Pipeline of Defense Failures



Does not generalize over threat models

Seems to generalize, but it’s an illusion

Does not generalize over attack algos

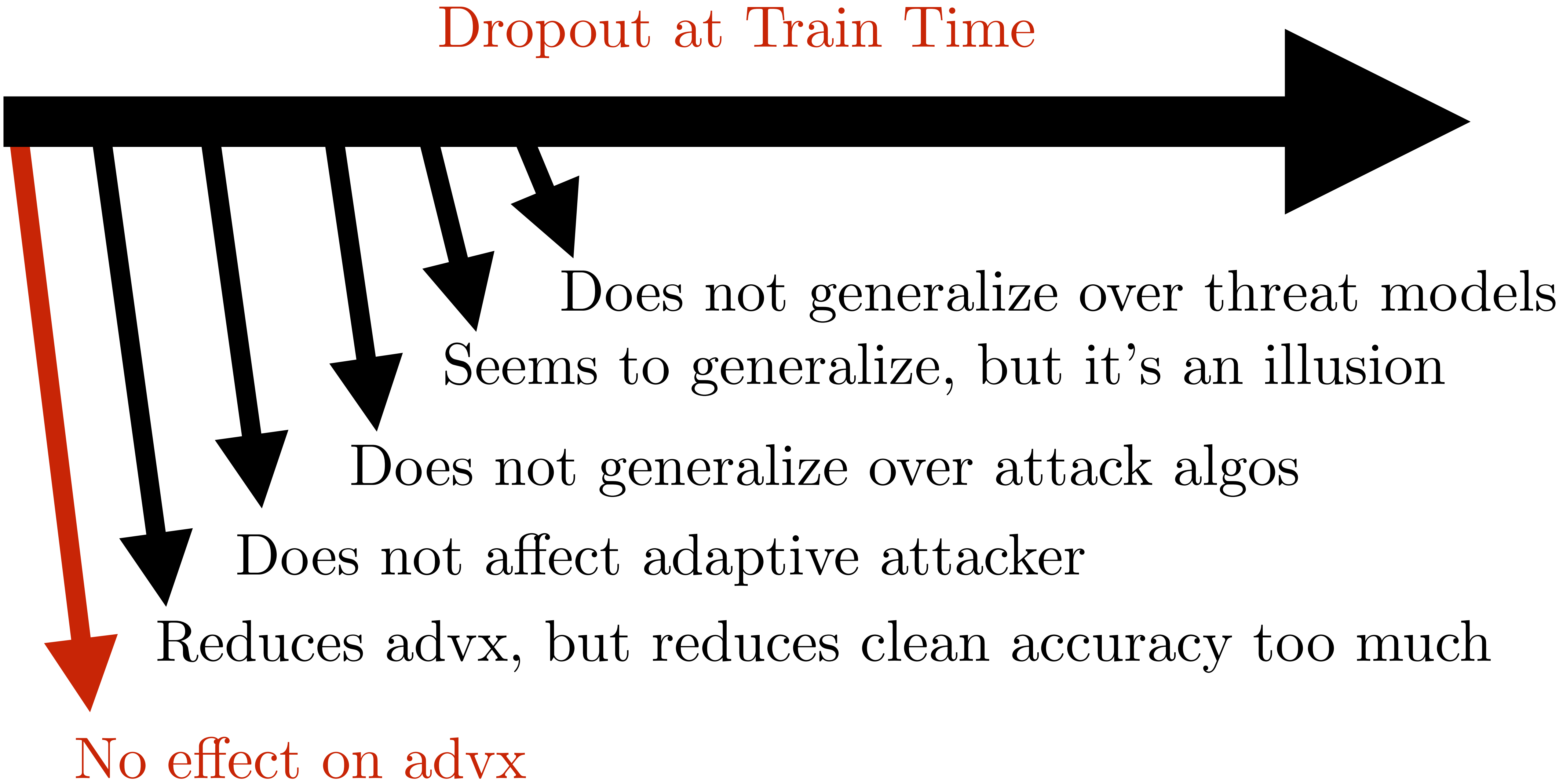
Does not affect adaptive attacker

Reduces advx, but reduces clean accuracy too much

No effect on advx

(Goodfellow 2018)

Pipeline of Defense Failures



Dropout at Train Time

Does not generalize over threat models

Seems to generalize, but it's an illusion

Does not generalize over attack algos

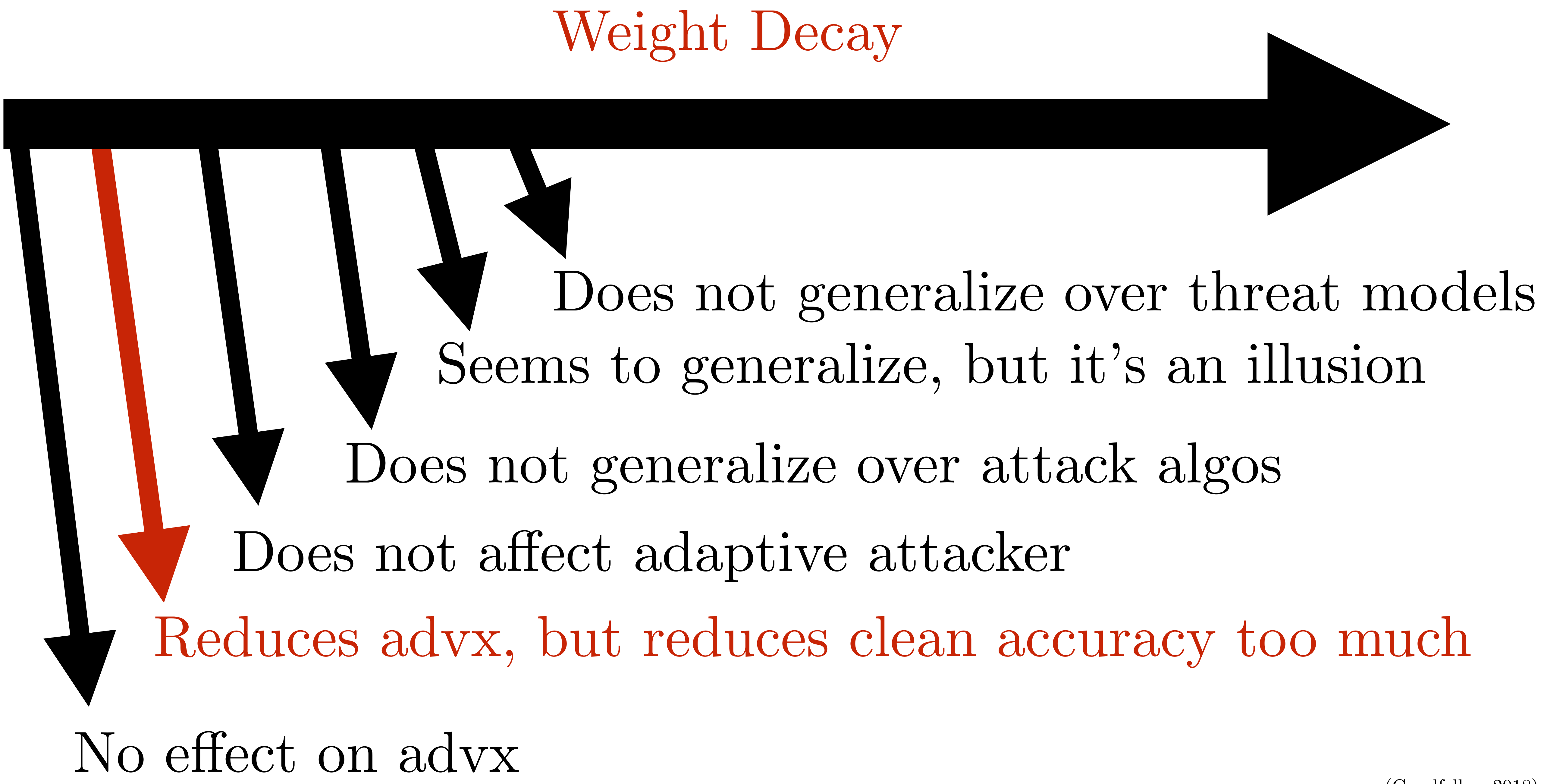
Does not affect adaptive attacker

Reduces advx, but reduces clean accuracy too much

No effect on advx

(Goodfellow 2018)

Pipeline of Defense Failures



Weight Decay

Does not generalize over threat models

Seems to generalize, but it’s an illusion

Does not generalize over attack algos

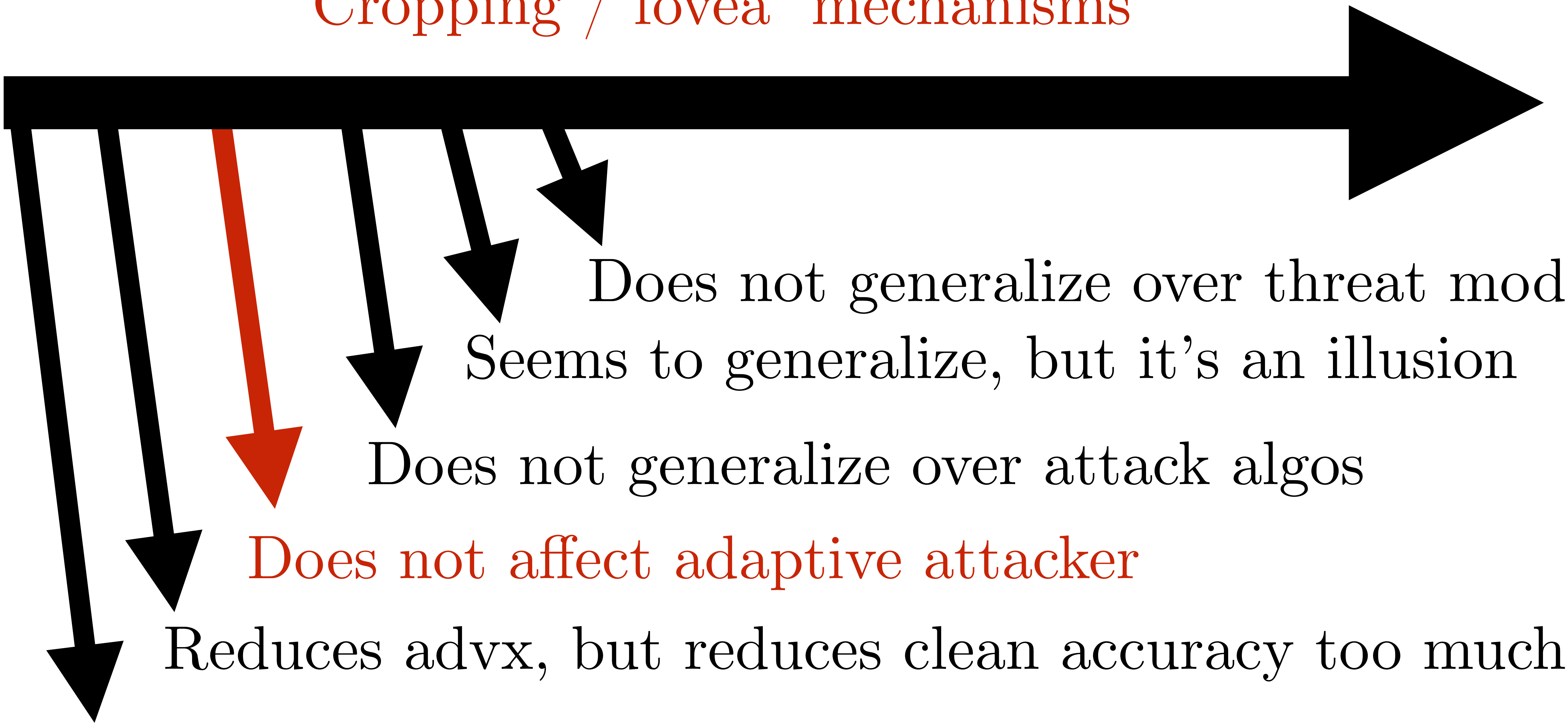
Does not affect adaptive attacker

Reduces advx, but reduces clean accuracy too much

No effect on advx

(Goodfellow 2018)

Pipeline of Defense Failures



Cropping / fovea mechanisms

Does not generalize over threat models

Seems to generalize, but it’s an illusion

Does not generalize over attack algos

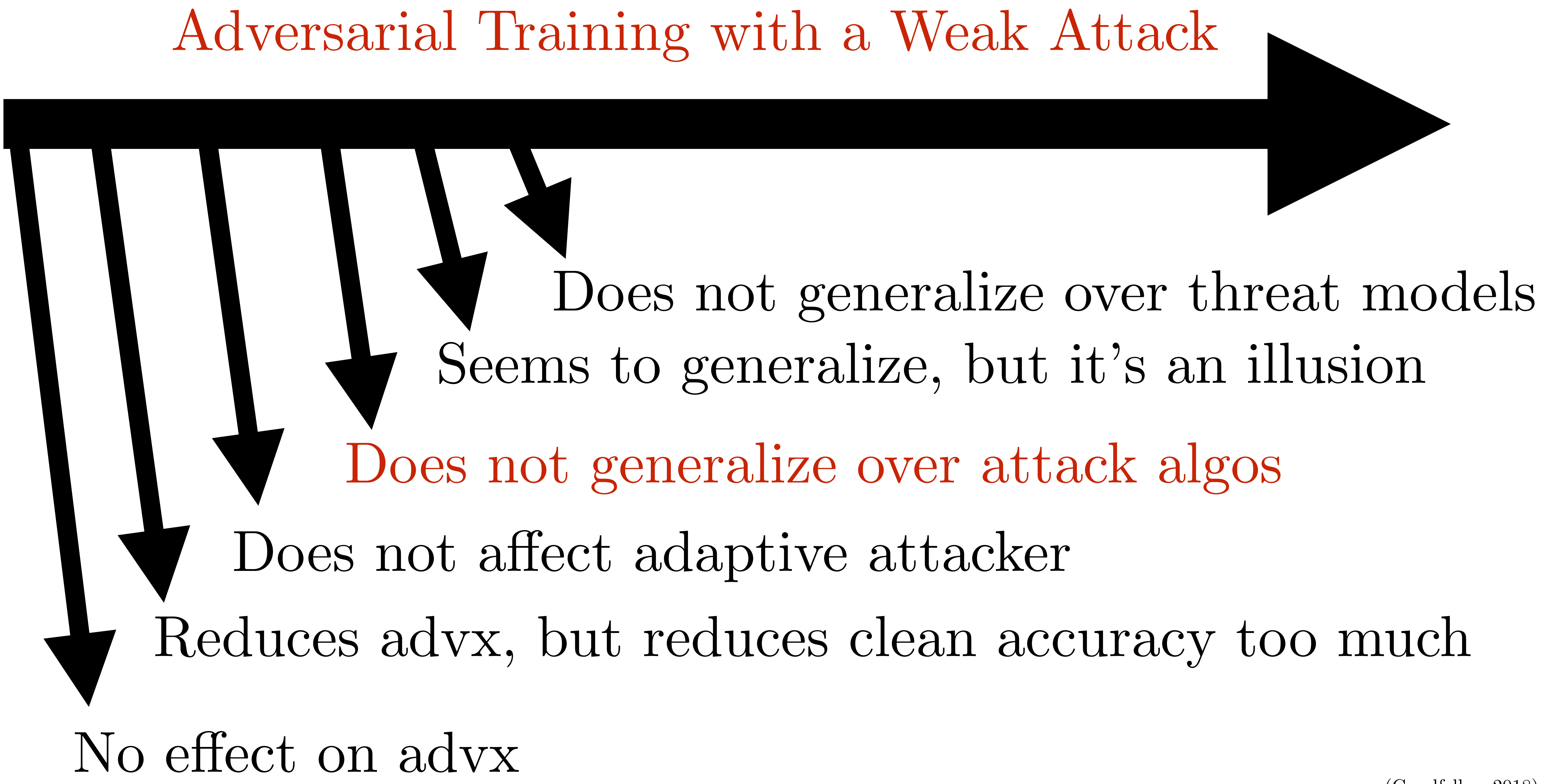
Does not affect adaptive attacker

Reduces advx, but reduces clean accuracy too much

No effect on advx

(Goodfellow 2018)

Pipeline of Defense Failures



Adversarial Training with a Weak Attack

Does not generalize over threat models

Seems to generalize, but it's an illusion

Does not generalize over attack algos

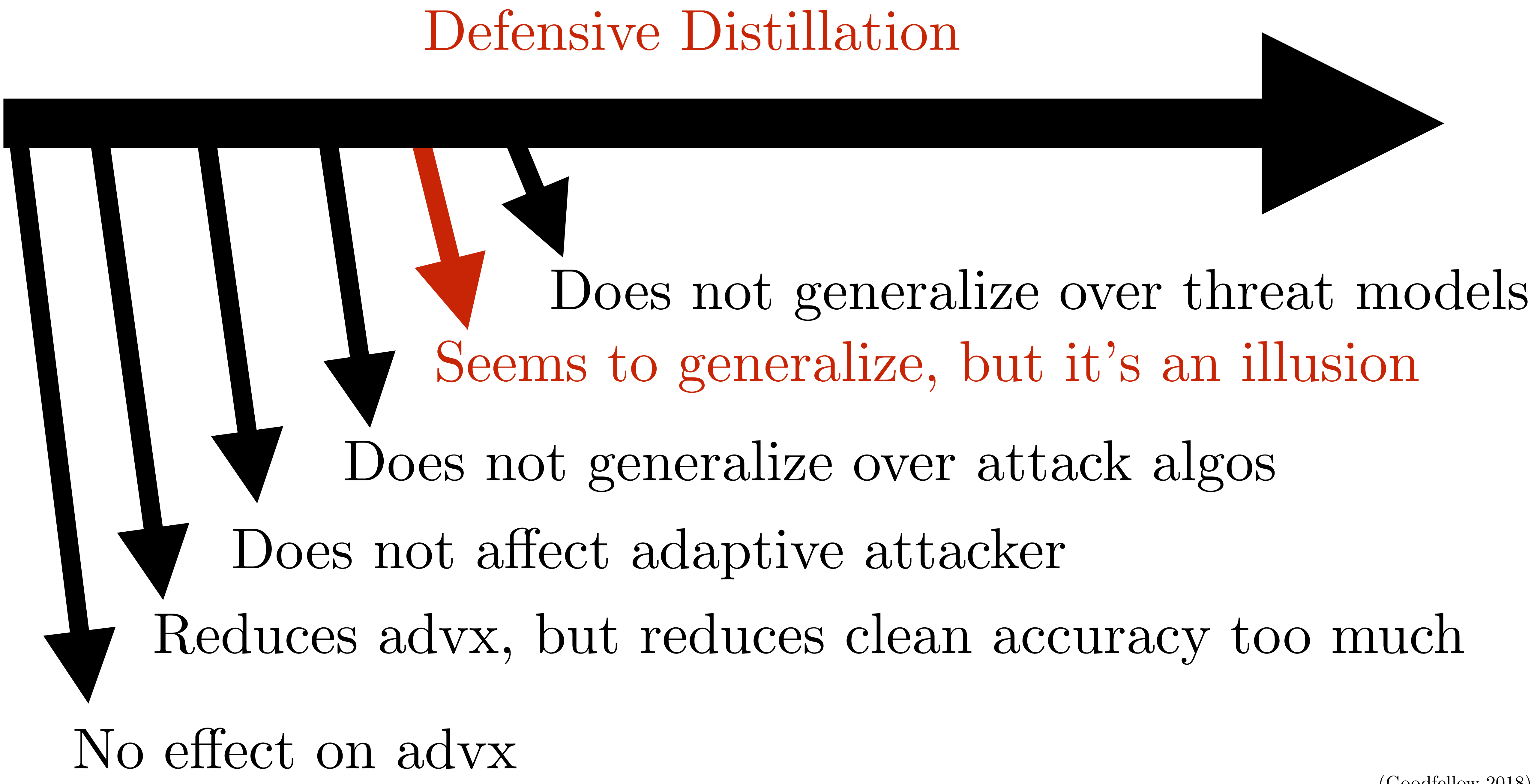
Does not affect adaptive attacker

Reduces advx, but reduces clean accuracy too much

No effect on advx

(Goodfellow 2018)

Pipeline of Defense Failures



Defensive Distillation

Does not generalize over threat models

Seems to generalize, but it’s an illusion

Does not generalize over attack algos

Does not affect adaptive attacker

Reduces advx, but reduces clean accuracy too much

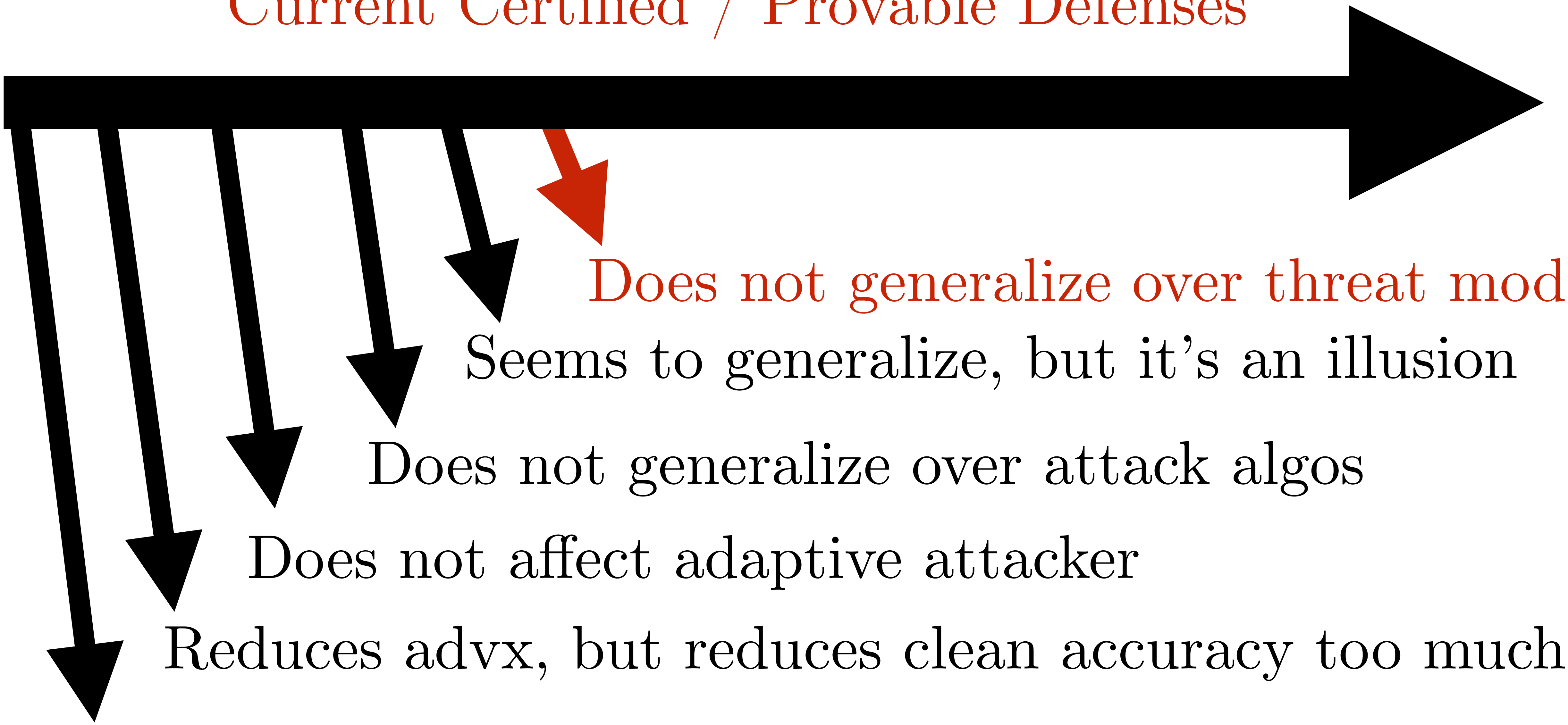
No effect on advx

(Goodfellow 2018)

Pipeline of Defense Failures

Adversarial Training with a Strong Attack

Current Certified / Provable Defenses



Does not generalize over threat models

Seems to generalize, but it’s an illusion

Does not generalize over attack algos

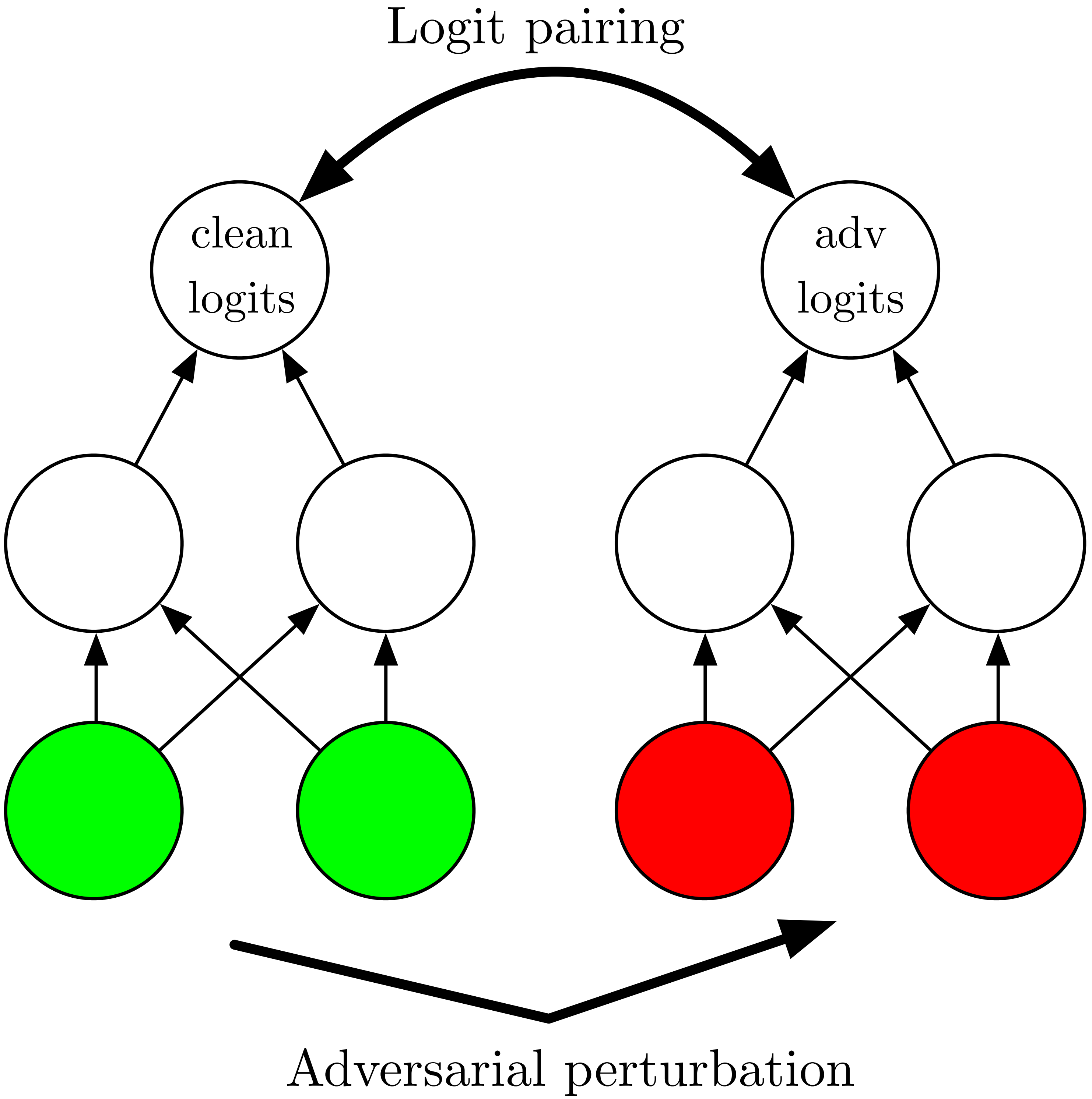
Does not affect adaptive attacker

Reduces advx, but reduces clean accuracy too much

No effect on advx

(Goodfellow 2018)

Adversarial Logit Pairing (ALP)



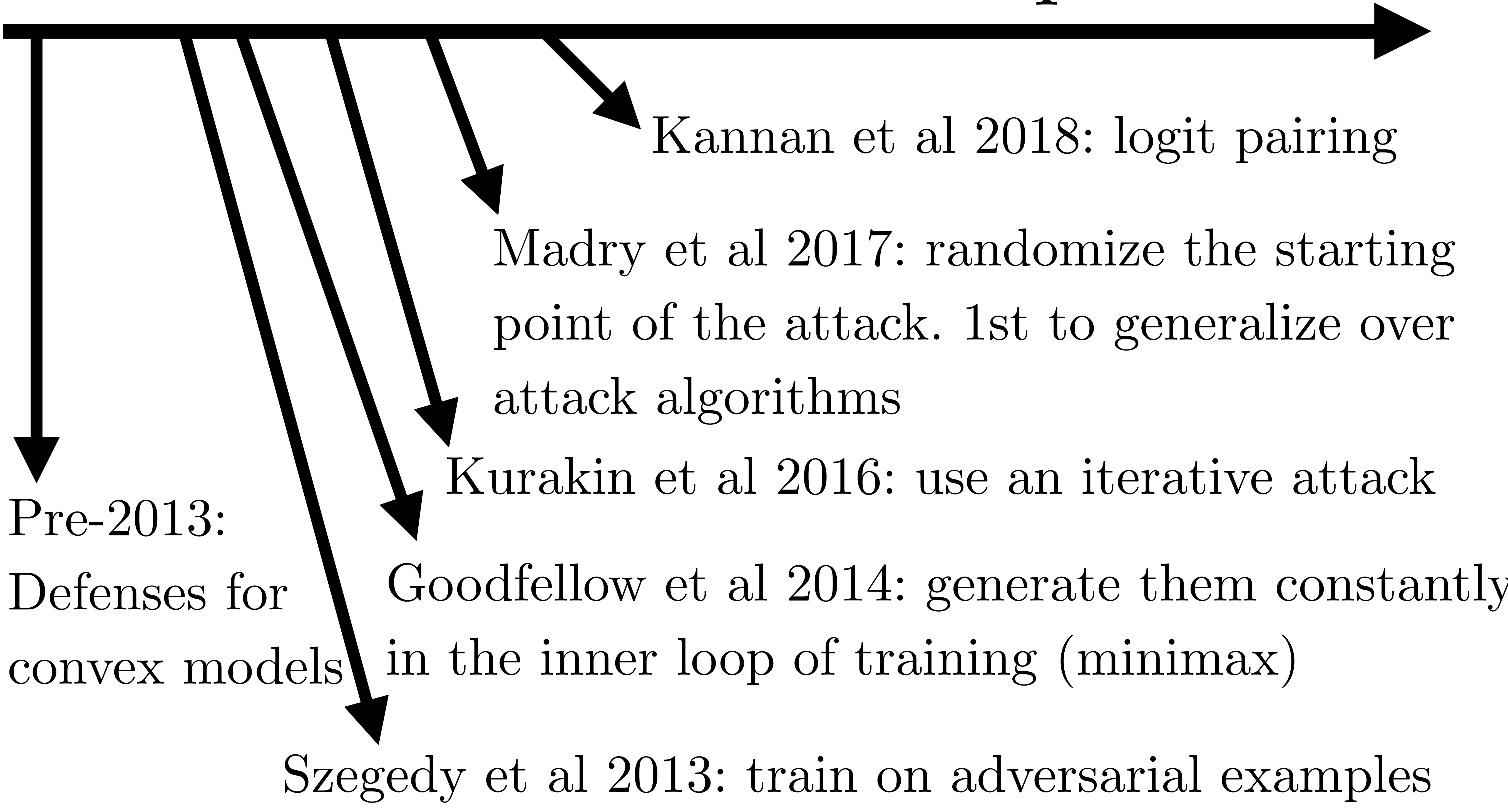
First approach to achieve >50% top-5 accuracy against iterative adversarial examples on ImageNet

Current state of the art

(Kannan et al 2018)

(Goodfellow 2018)

Timeline of Defenses Against Adversarial Examples



Kannan et al 2018: logit pairing

Madry et al 2017: randomize the starting point of the attack. 1st to generalize over attack algorithms

Kurakin et al 2016: use an iterative attack

Pre-2013: Defenses for convex models

Goodfellow et al 2014: generate them constantly in the inner loop of training (minimax)

Szegedy et al 2013: train on adversarial examples

(Goodfellow 2018)

Disappointing outcome of toy game

- My hope: something simple (Bayesian deep nets?) will solve the adversarial example problem, do well on the points we can measure via norm ball label propagation, also do well on points that are hard to measure

- Outcome so far: best results are obtained by directly optimizing the performance measure. Both for empirical and for certified approaches. Defenses do not generalize out of the norm ball.

Future Directions: Indirect Methods

- Do not just optimize the performance measure exactly
- Best methods so far:
  - Logit pairing (non-adversarial)
  - Label smoothing
  - Logit squeezing
- Can we perform a lot better with other methods that are similarly indirect?

(Goodfellow 2018)

Future Directions: Better Attack Models

- Add new attack models other than norm balls
- Study messy real problems in addition to clean toy problems
- Study certification methods that use other proof strategies besides local smoothness
- Study more problems other than vision

(Goodfellow 2018)

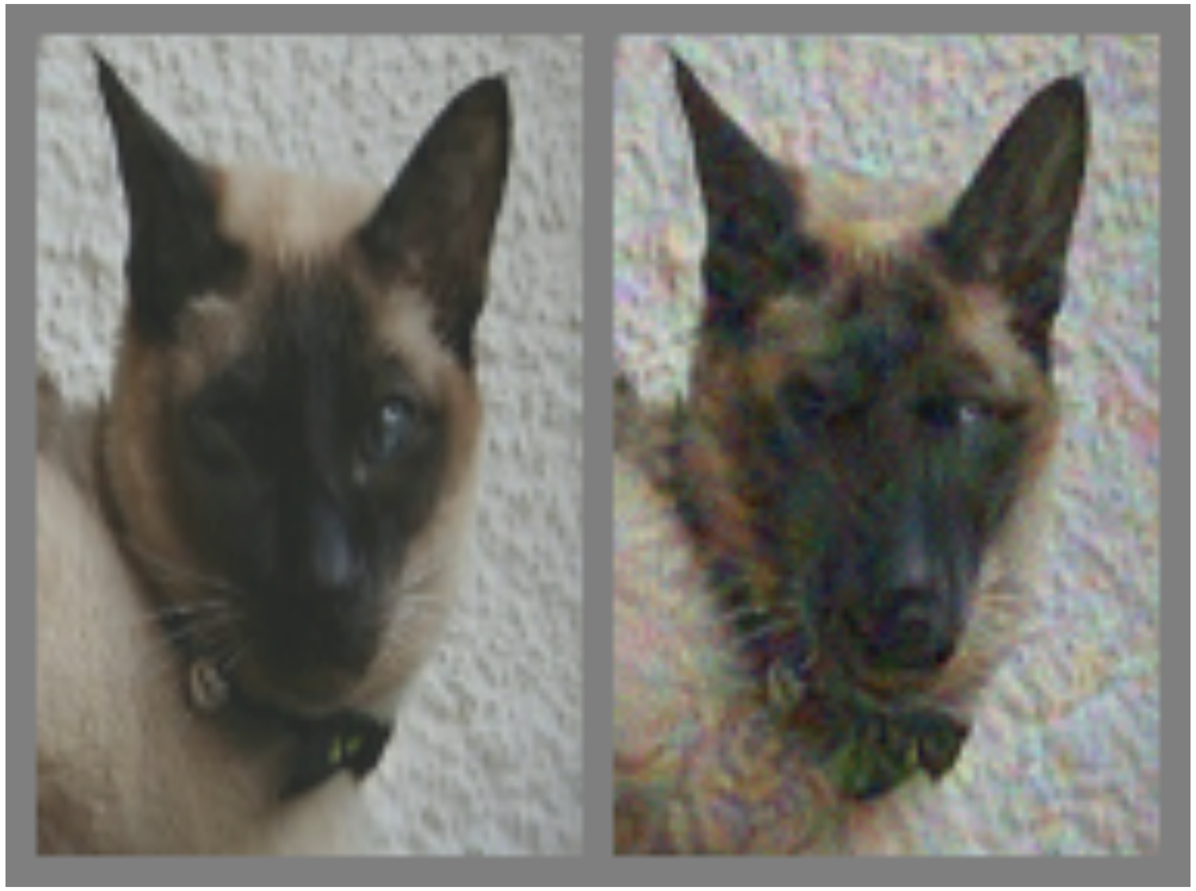
Future Directions: Security Independent from Traditional Supervised Learning

- Until recently, both adversarial example research and traditional supervised learning seemed fully aligned: just make the model better
- They still share this goal
- It is now clear security research must have some independent goals. For two models with the same error volume, for reasons of security we prefer:
    - The model with lower confidence on mistakes
    - The model whose mistakes are harder to find
    - A stochastic model that does not repeatedly make the same mistake on the same input
    - A model whose mistakes are less valuable to the attacker / costly to the defender
    - A model that is harder to reverse engineer with probes
    - A model that is less prone to transfer from related models

Some Non-Security Reasons to Study Adversarial Examples

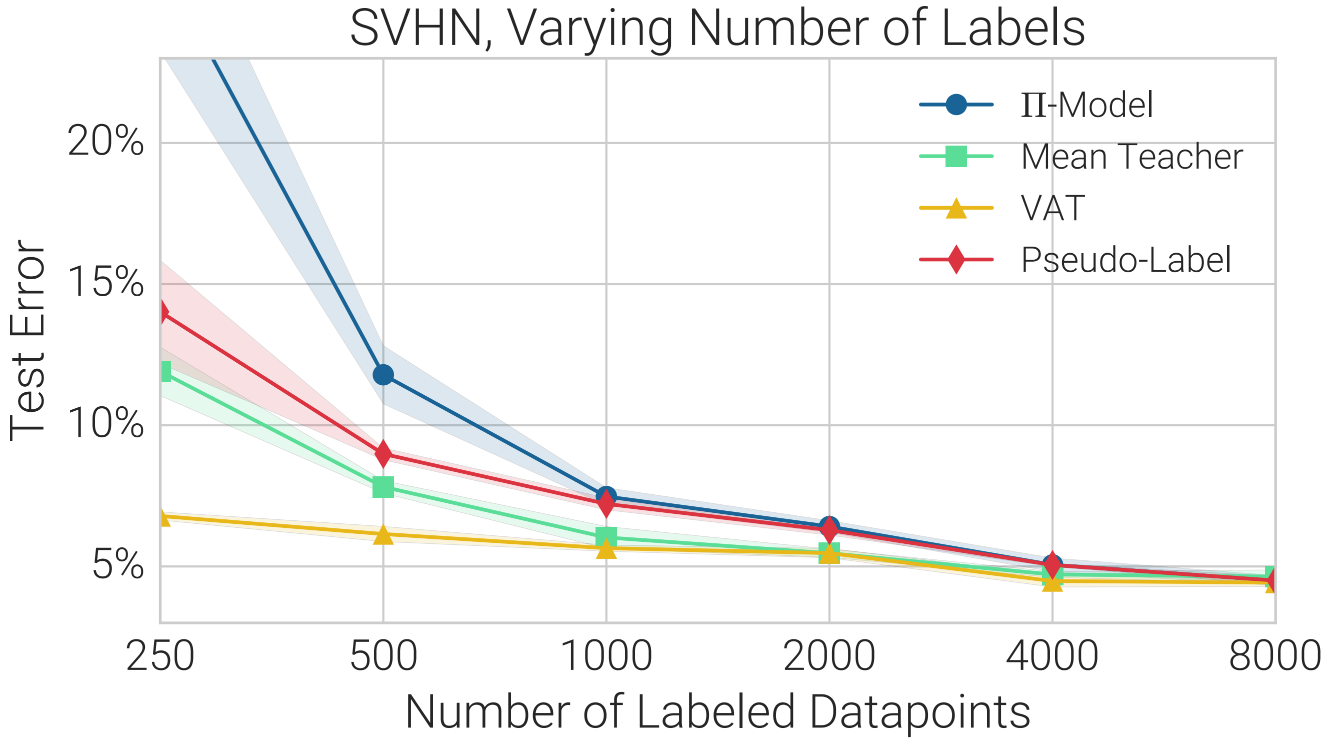
Improve Supervised Learning
(Goodfellow et al 2014)

Understand Human Perception



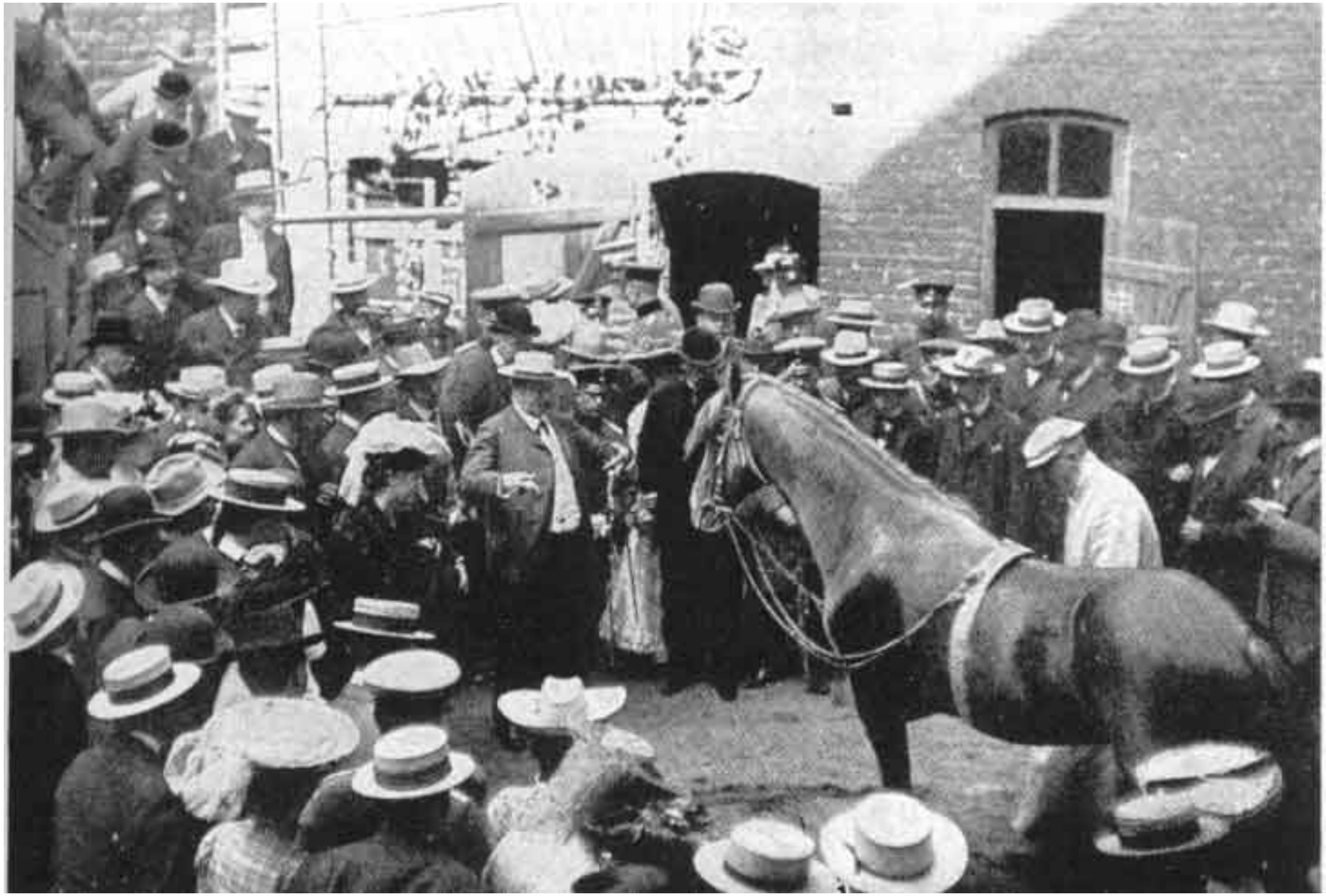
Gamaleldin et al 2018

Improve Semi-Supervised Learning
(Miyato et al 2015)



(Oliver+Odena+Raffel et al, 2018)

Clever Hans



("Clever Hans,
Clever
Algorithms,"
Bob Sturm)

(Goodfellow 2018)

Get involved!

https://github.com/tensorflow/cleverhans

cleverhans