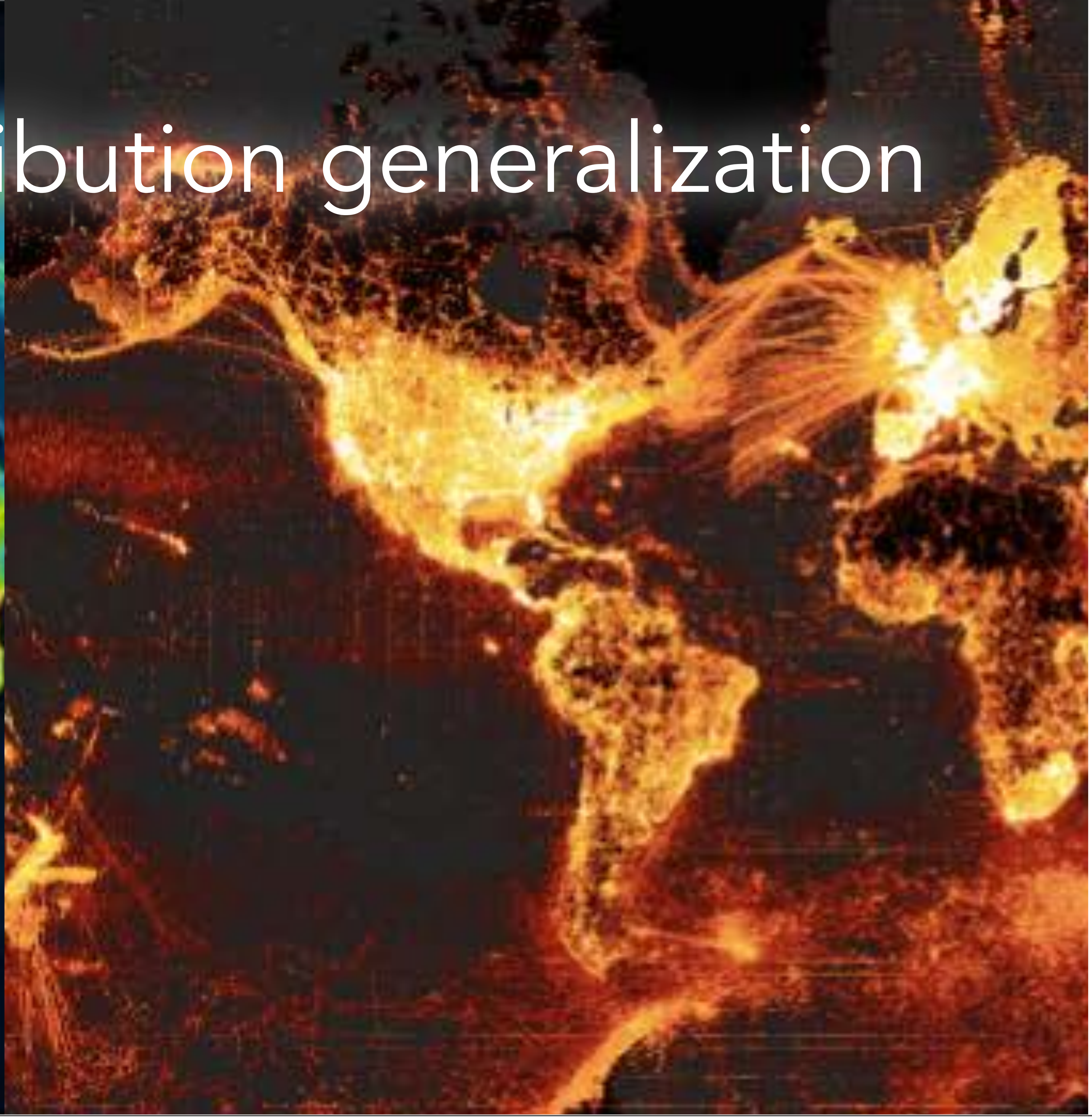
Lecture 17: Out-of-distribution generalization

Speaker: Sara Beery

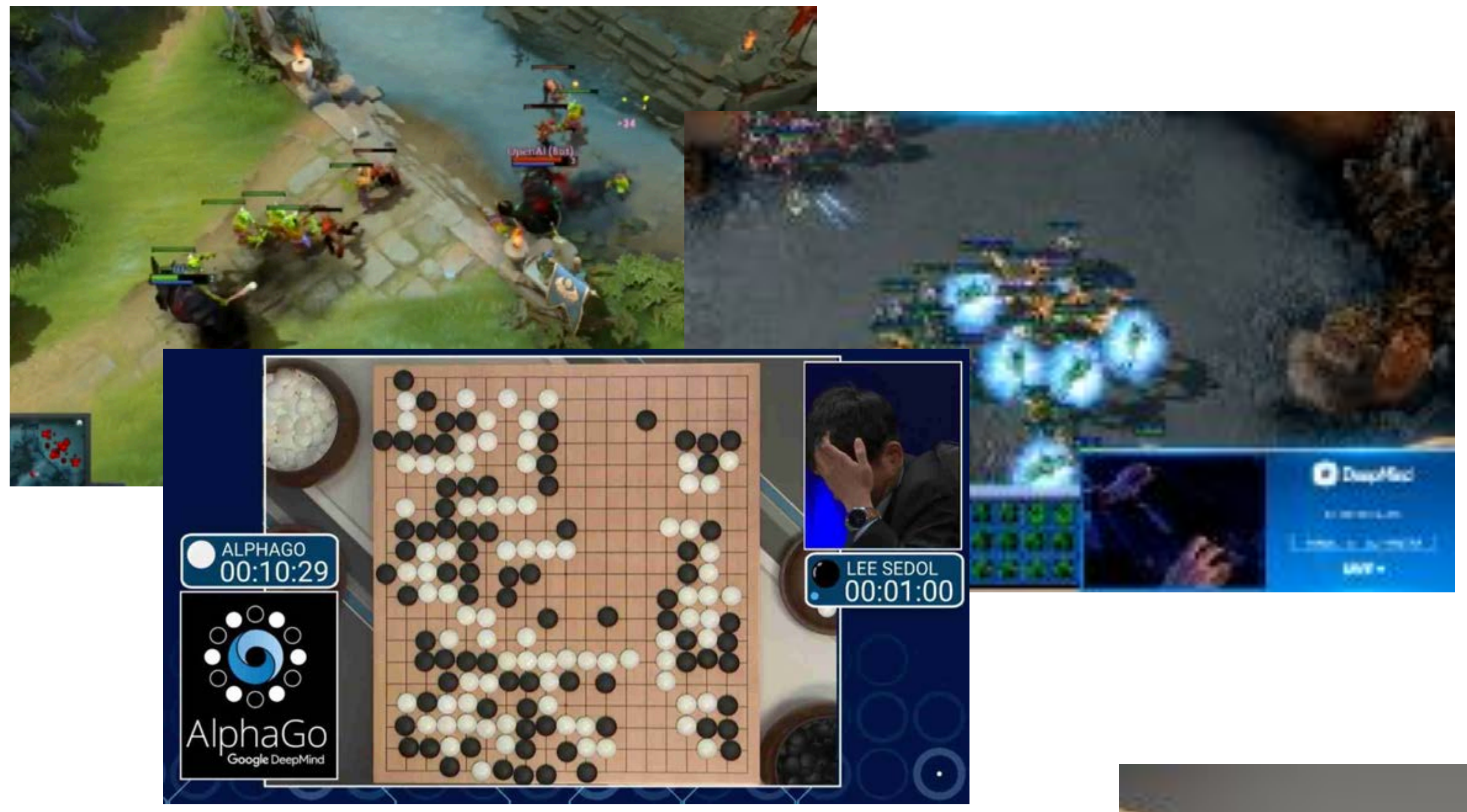


Machine Learning: A Success Story

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.

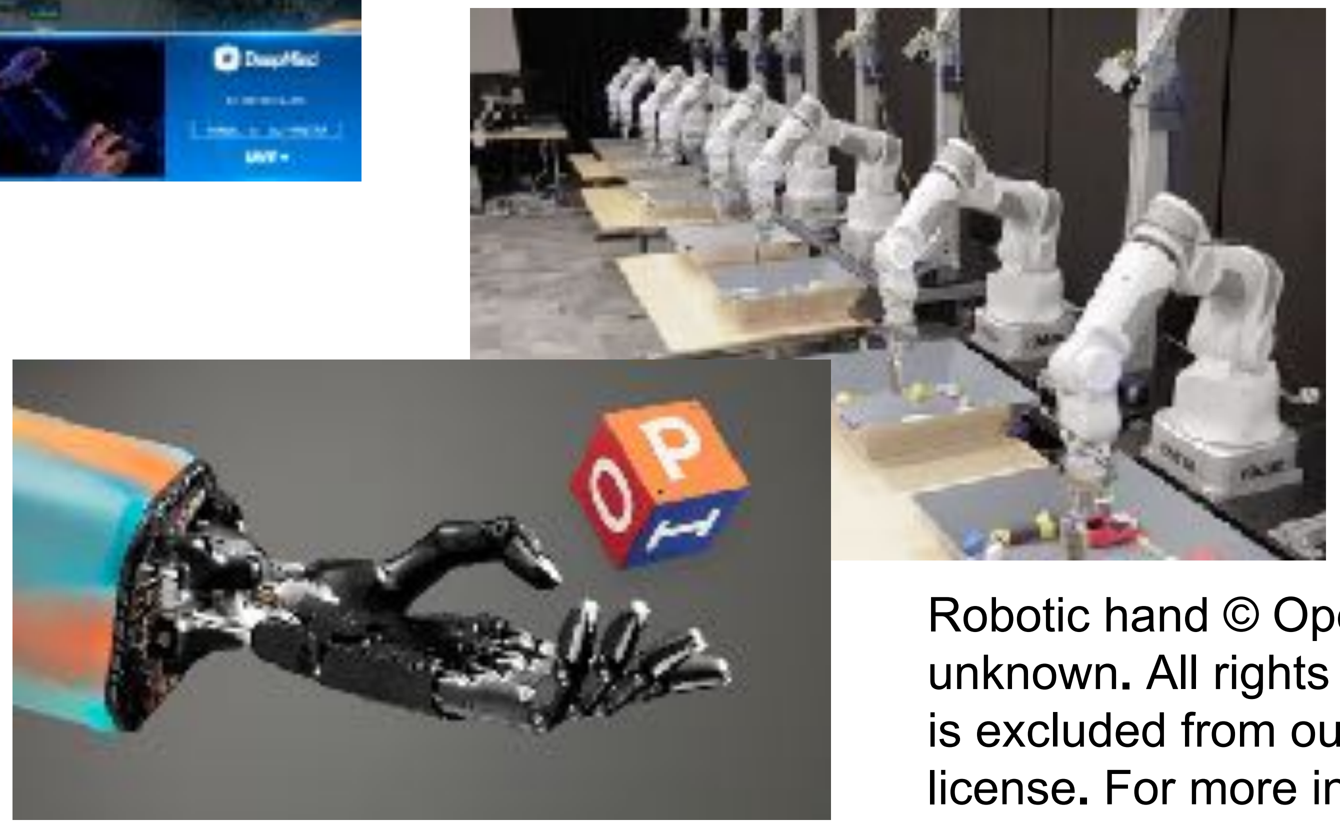
Machine Translation



Strategy Games

Image removed to copyright restrictions.

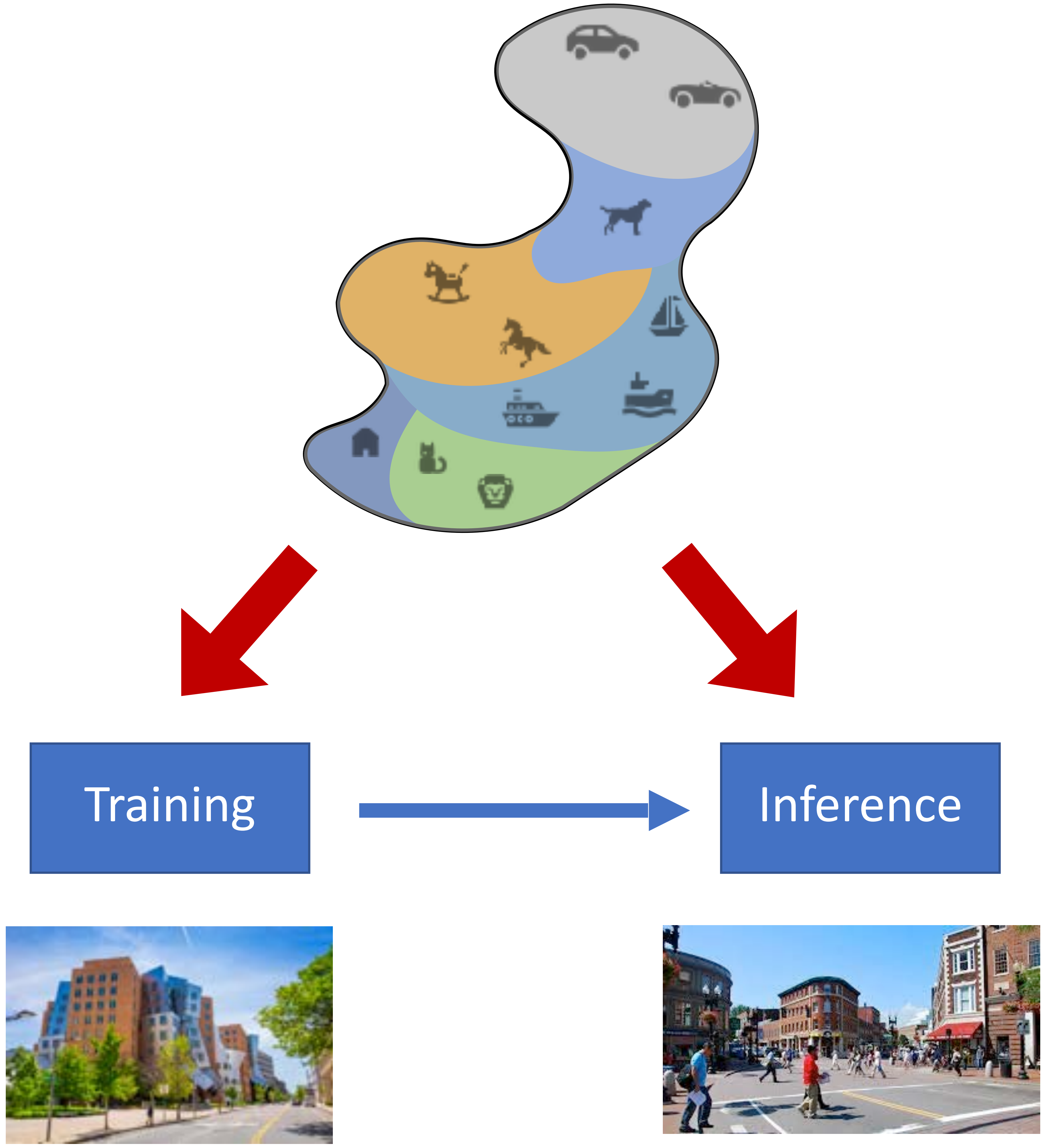
Realistic Image Generation



Robotic Manipulation

Are ML systems really ready for the real world?

Standard ML setting

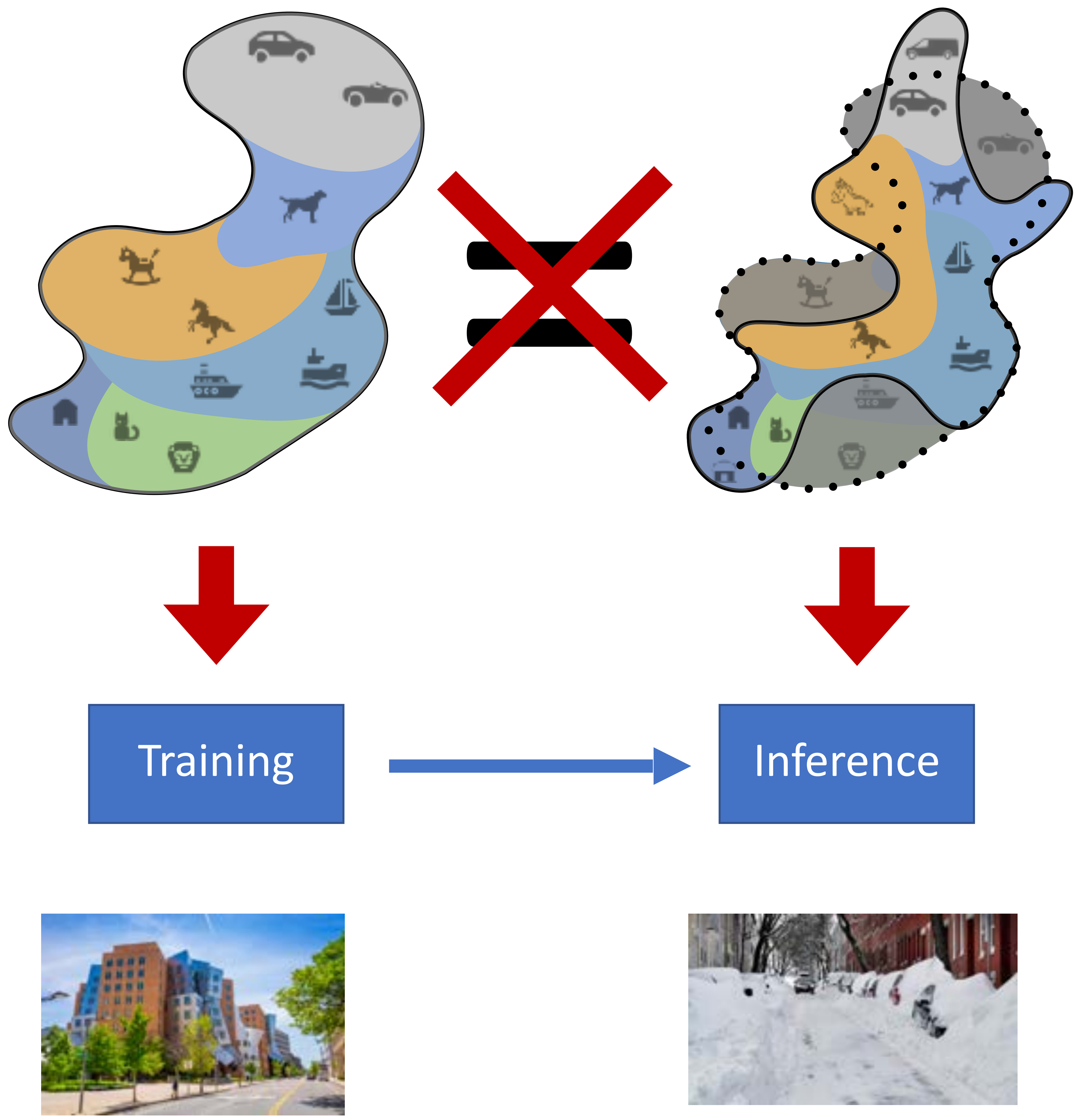


training distribution

=

test distribution

… vs the real world

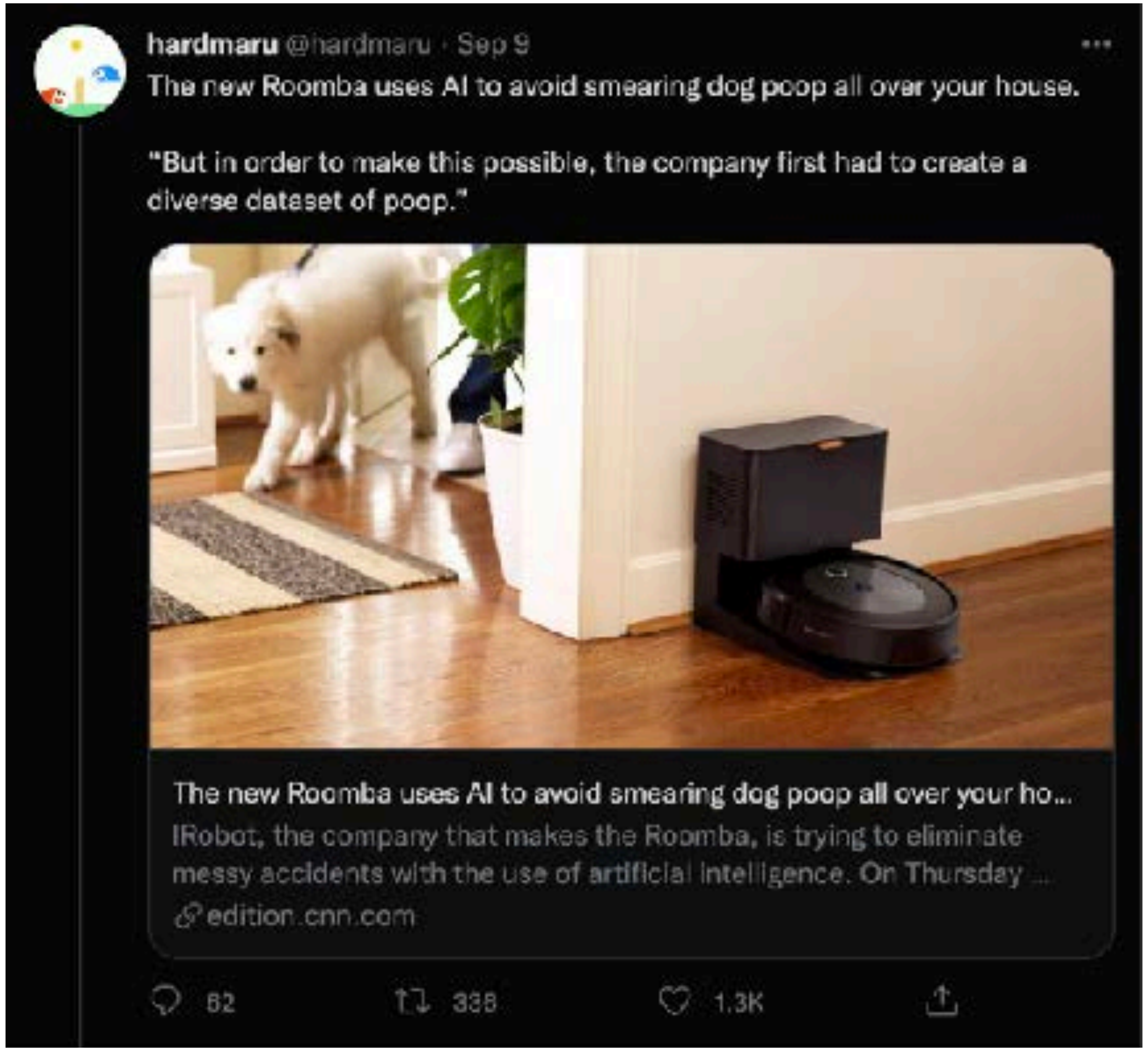


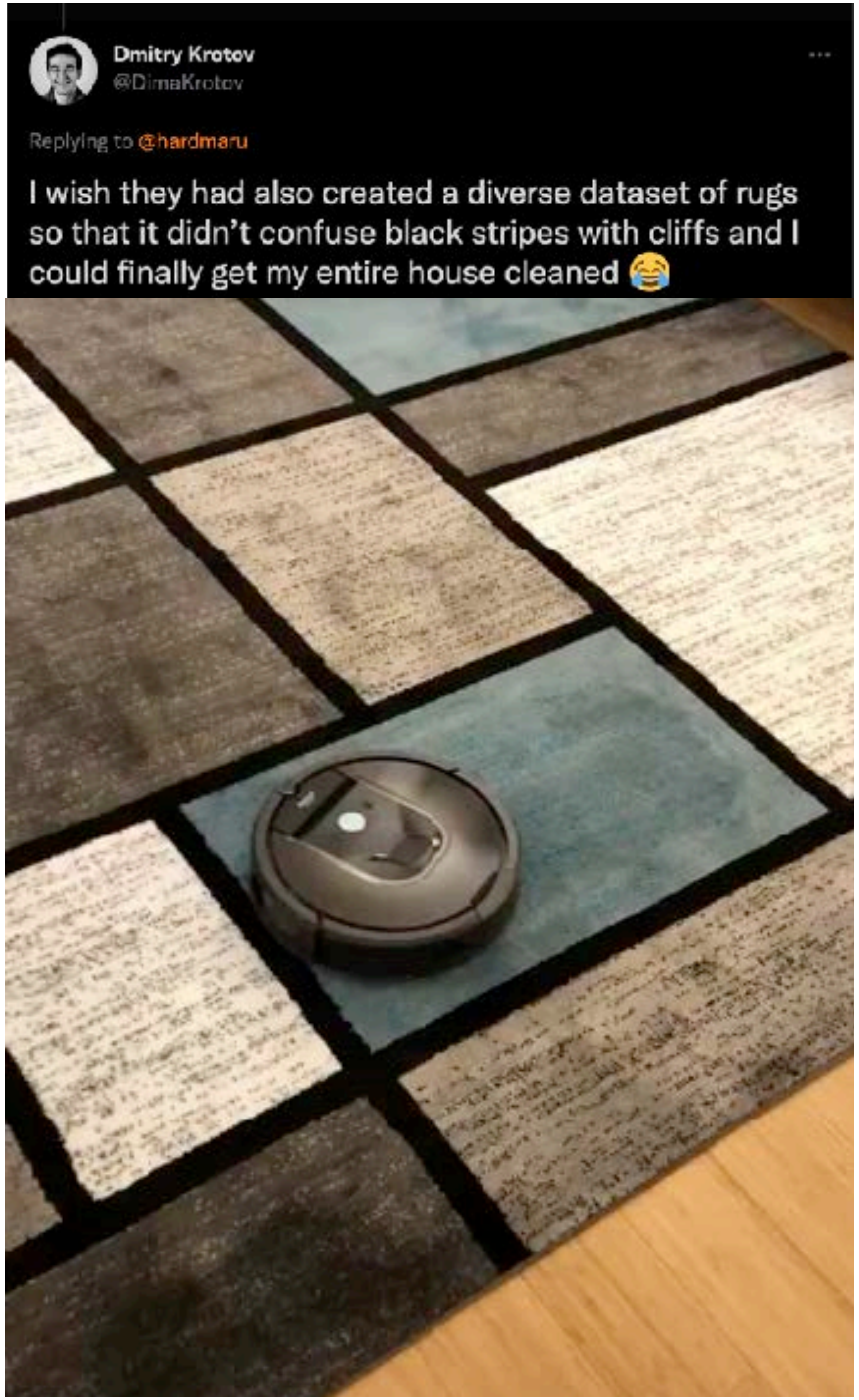
deploy model on data from a different distribution

e.g.:

- perturbed data
- different label distribution
- other shifts (sequence/graph size, weather, country/city, source of measurement,…)

What can go wrong?





Concrete Problems in AI Safety

Dario Amodei* Google Brain

Chris Olah* Google Brain

Jacob Steinhardt Stanford University

Paul Christiano UC Berkeley

John Schulman OpenAI

Dan Mane Google Brain

might serve a benchmarking role similar to that of the bAbI tasks [163], with the eventual goal being to develop a single architecture that can learn to avoid catastrophes in all environments in the suite.

7 Robustness to Distributional Change

All of us occasionally find ourselves in situations that our previous experience has not adequately prepared us to deal with-for instance, flying an airplane, traveling to a country whose culture is very different from ours, or taking care of children for the first time. Such situations are inherently

Outline for today

- Adversarial examples and training: small perturbations
- Distribution Shifts

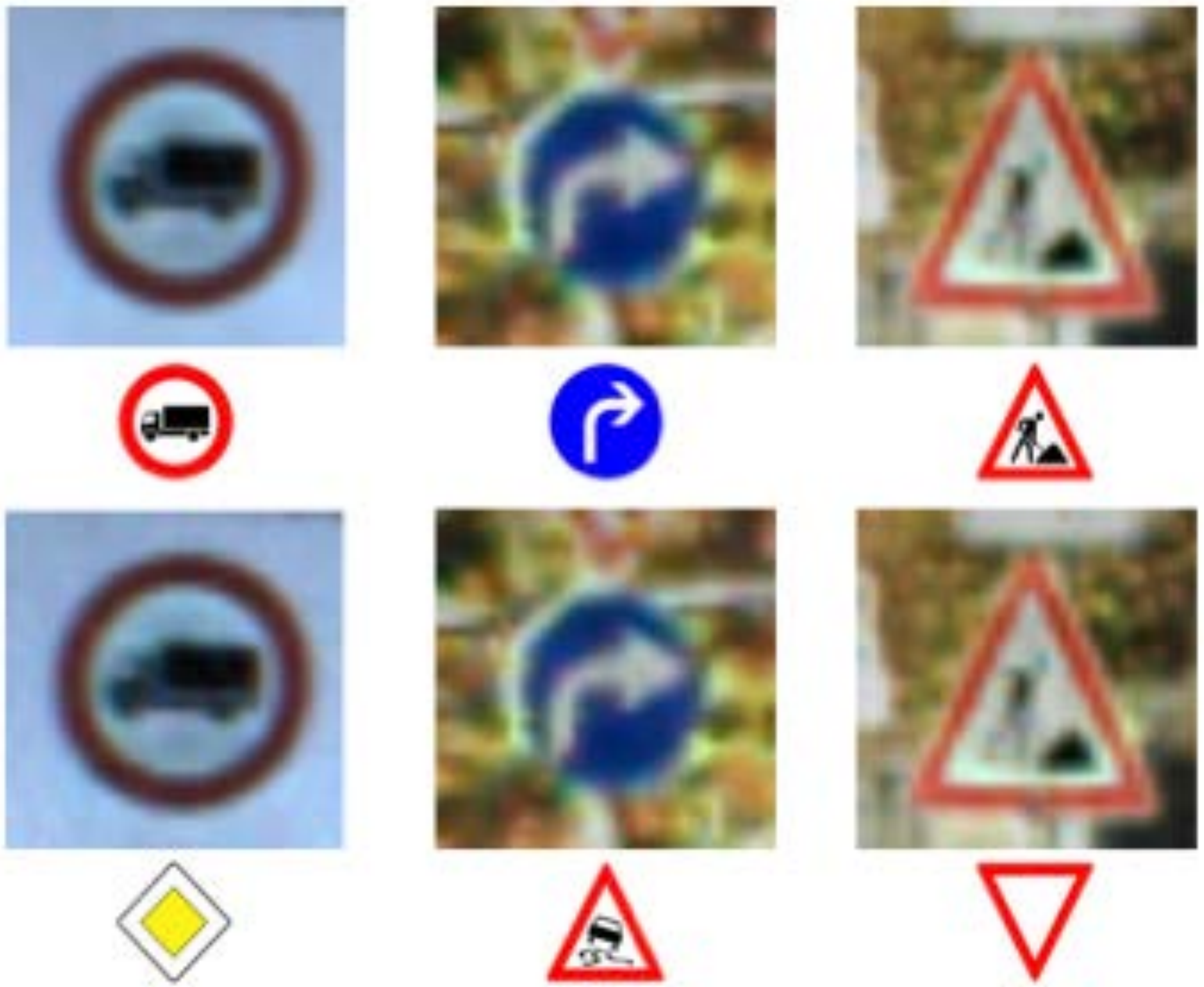
Adversarial examples

Adversarial examples

- ML model predictions are (mostly) accurate but can be brittle

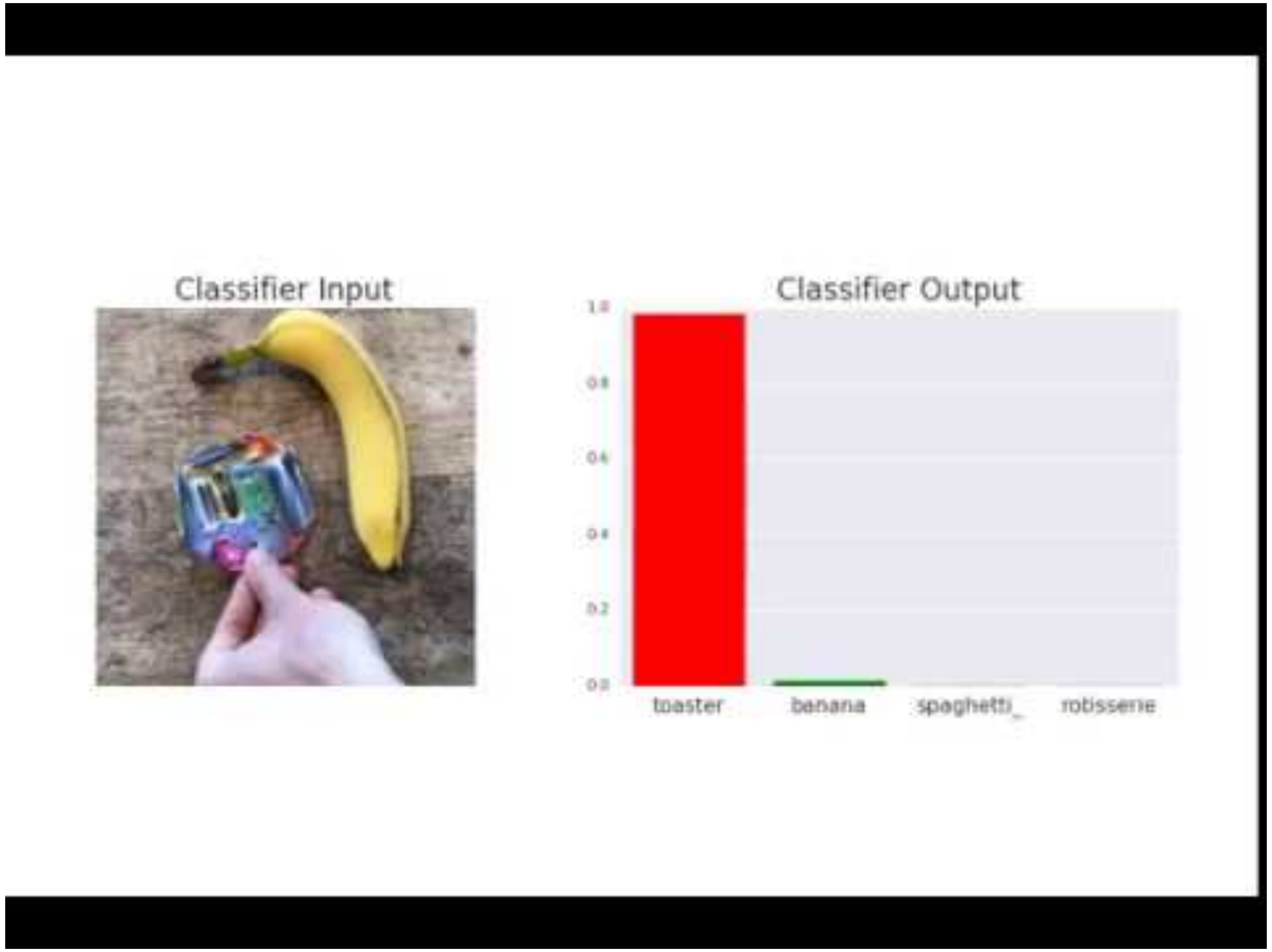
example: Szegedy et al 2013, obtained from https://gradientscience.org/intro_adversarial/

Adversarial examples



Papernot et al 2017, Practical black-box attacks against machine learning

Adversarial stickers



Adversarial stickers

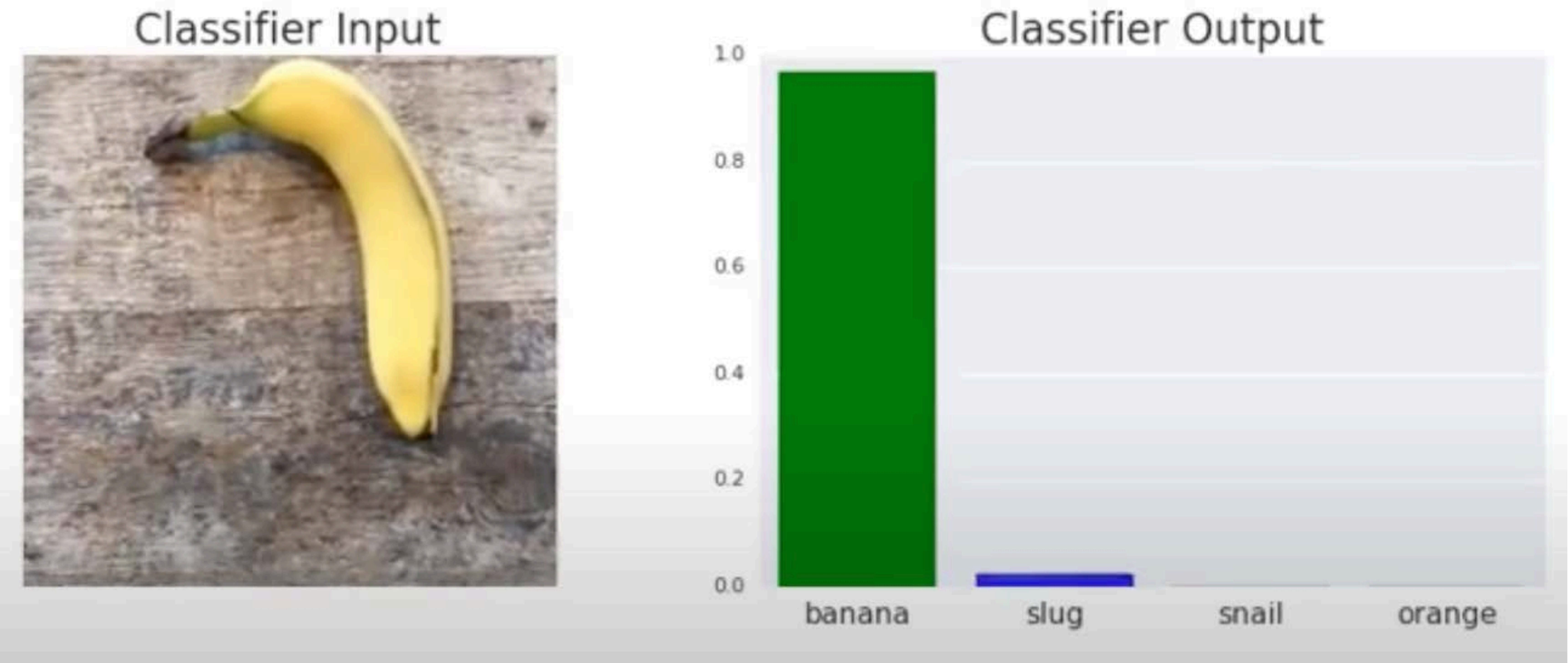


image: Brown et al 2018, Adversarial patch. https://youtu.be/i1sp4X57TL4

Adversarial stickers

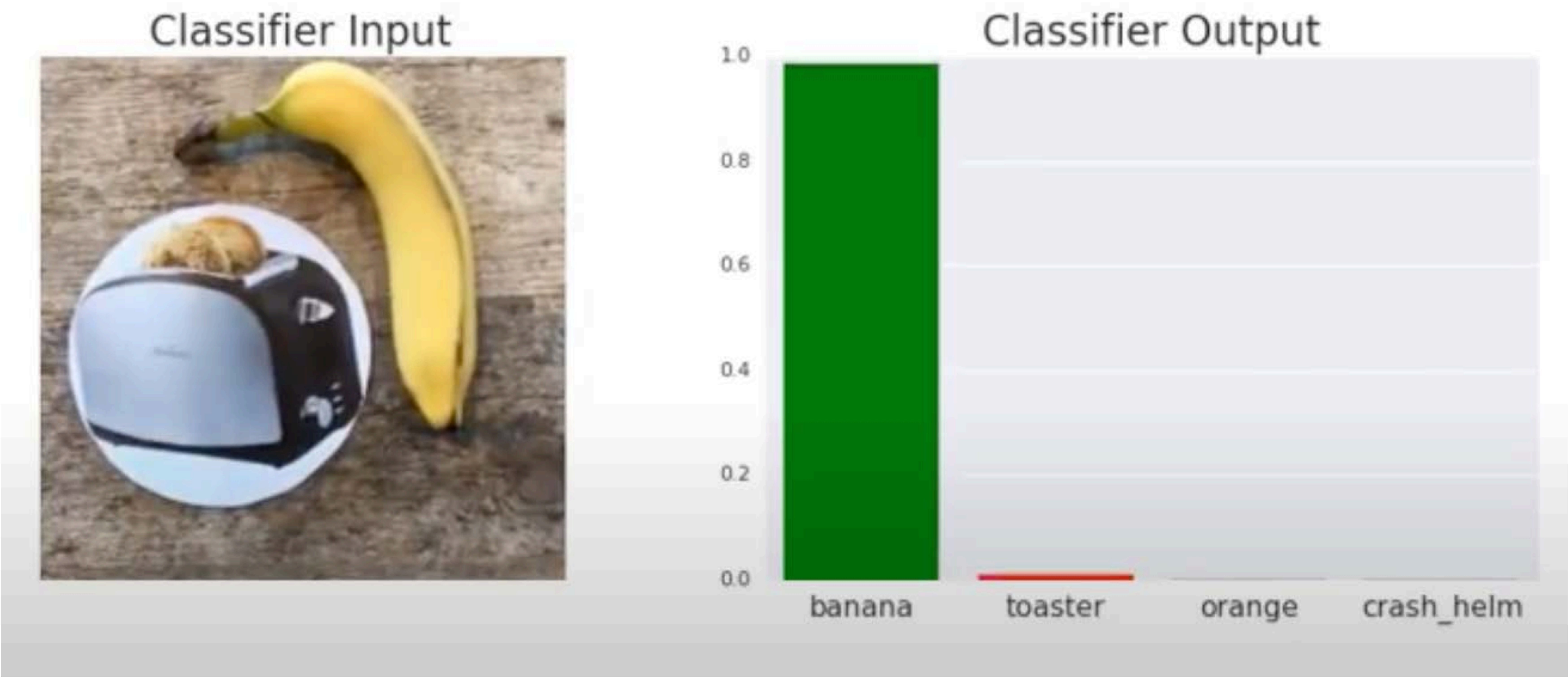


image: Brown et al 2018, Adversarial patch. https://youtu.be/i1sp4X57TL4

Adversarial stickers

image: Brown et al 2018, Adversarial patch. https://youtu.be/i1sp4X57TL4

Adversarial examples 3D-printed

image: Athalye et al 2018, Synthesizing robust adversarial examples

Adversarial examples 3D-printed

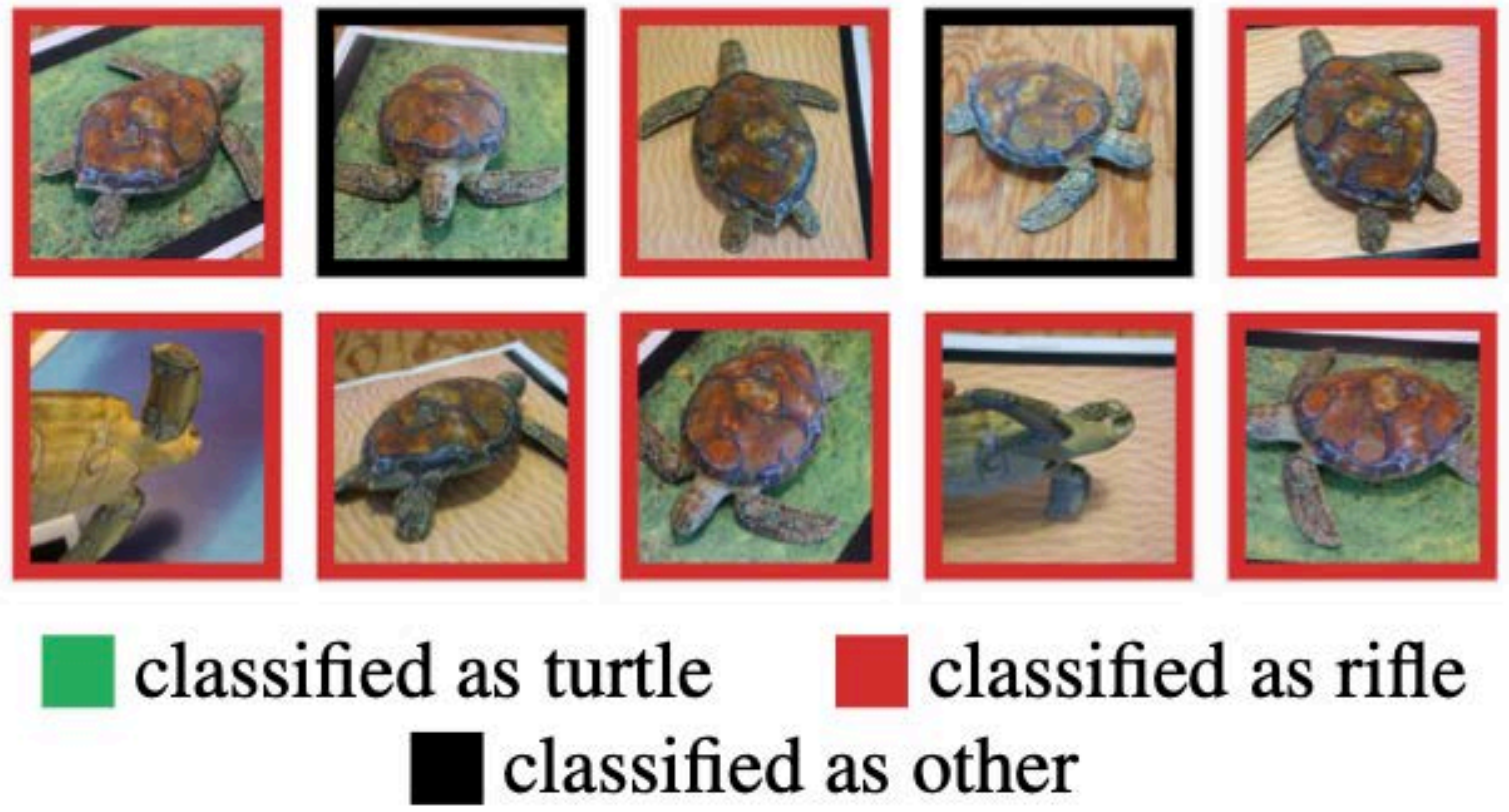
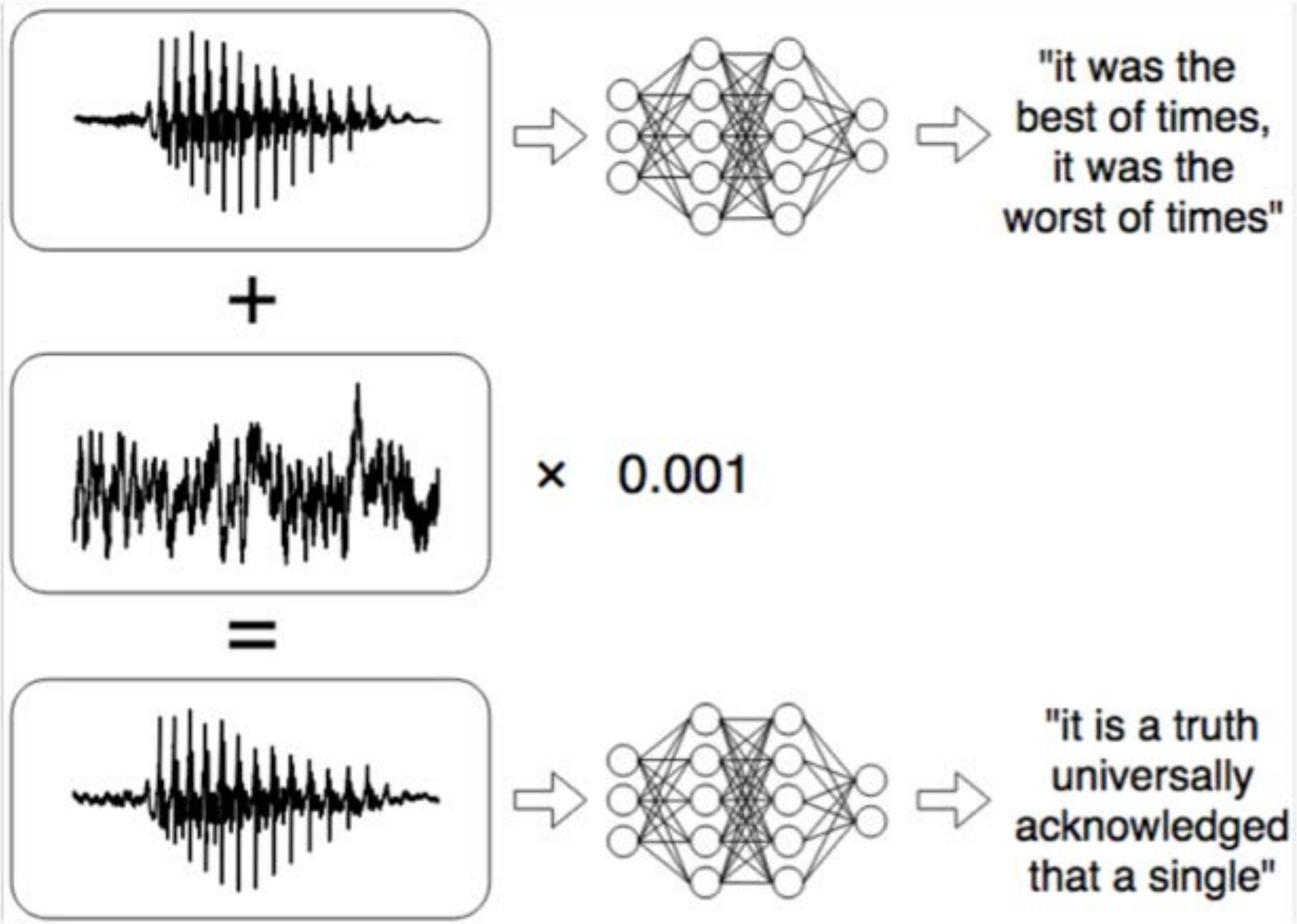


image: Athalye et al 2018, Synthesizing robust adversarial examples

Speech recognition example

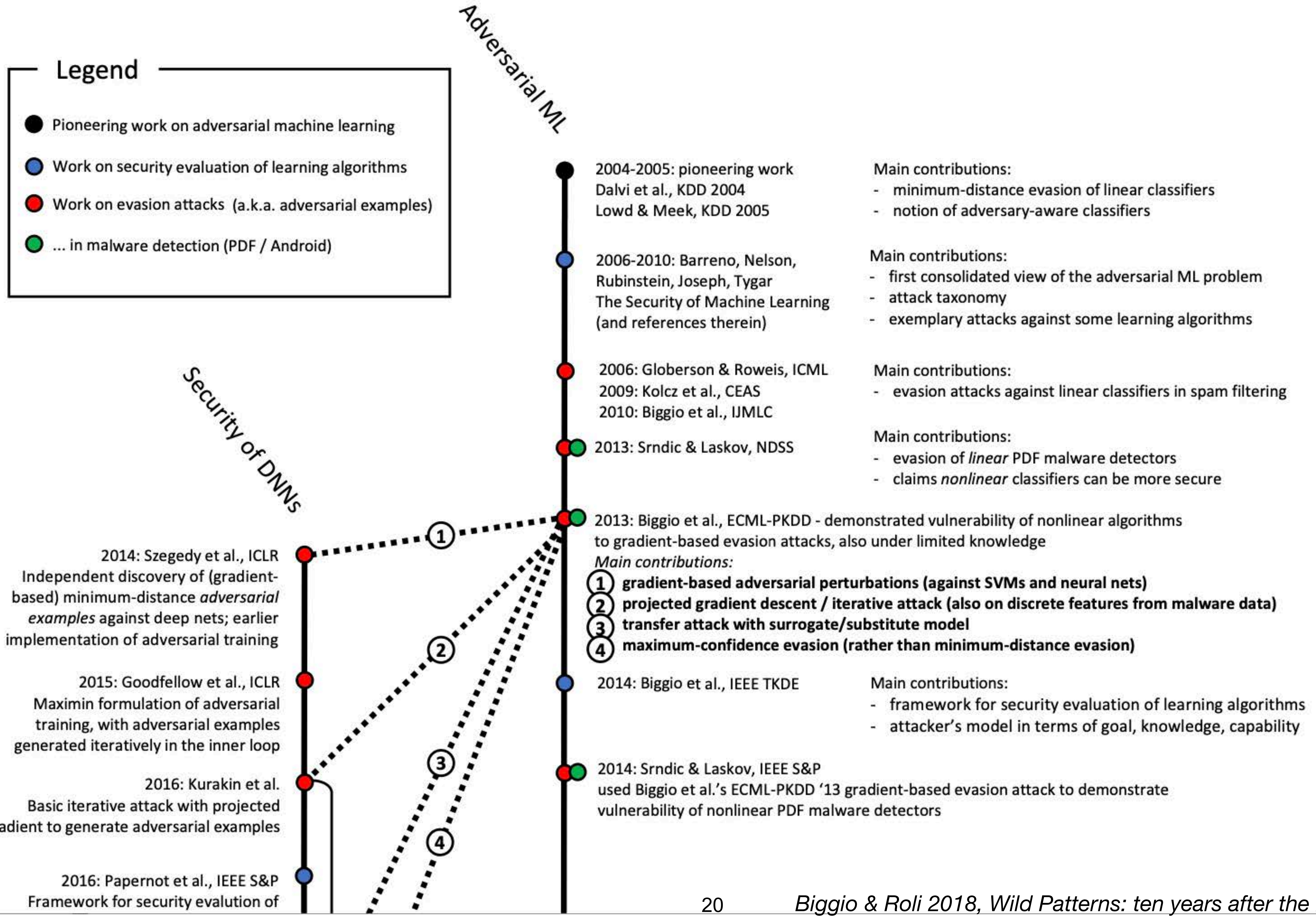


Carlini & Wagner 2018

Hmmmm….

- Are our models completely useless?
- Why does this happen?
- Can one prevent it?

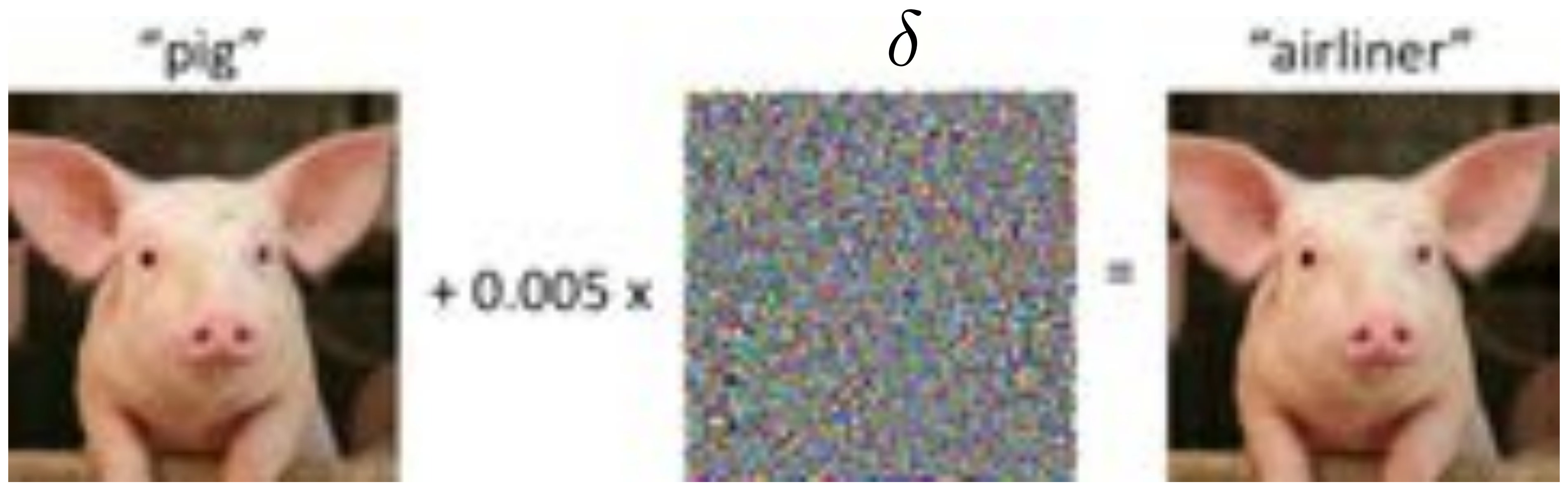
History of adversarial examples / brittleness



Biggio & Roli 2018, Wild Patterns: ten years after the rise of adversarial machine learning

How do you create an adversarial example?

- want: small perturbation that does not change meaning to a human, but to ML model



- model outputs $P_\theta(y \mid \mathbf{x})$ (softmax)

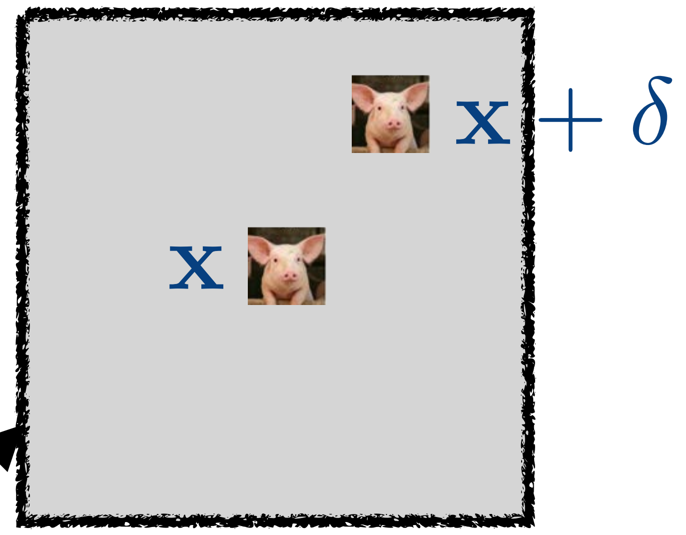
- adversarial example: $\max\limits_{\delta \in \Delta} P_\theta(y_{\text{target}} \mid \mathbf{x} + \delta)$

small perturbation, e.g.

wrong class ("airliner")

input image

$\Delta = \{\delta \in \mathbb{R}^d \mid \|\delta\|_\infty < \epsilon\}$



allowed perturbations

How to find an adversarial example?

ML model

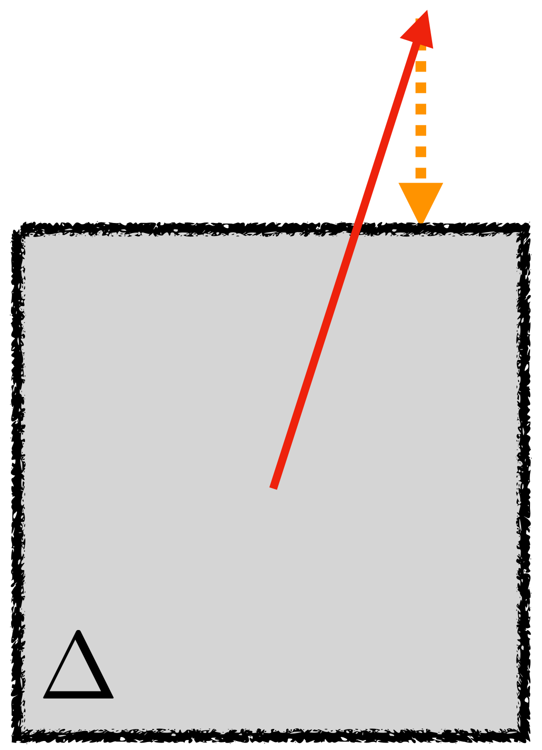
$$\max_{\delta \in \Delta} \boxed{P_\theta}(y_{\text{target}} \mid \mathbf{x} + \delta)$$

small perturbation, e.g.

wrong class (“airliner”)

input image

- e.g. Projected gradient ascent (we update data perturbation δ!):
  1. take a step in the direction of the gradient:
     $\delta^{(t+1)} = \delta^{(t)} + \eta \cdot \nabla_\delta P_\theta(y_{\text{target}} \mid \mathbf{x} + \delta)$
  2. project the result back into the feasible set Δ
  3. repeat steps 1 & 2



How to "defend" against adversarial examples?

Recall:

- Adversarial example versus standard training:

$$\max_{\delta \in \Delta} \text{Loss}\Big(f_\theta(\mathbf{x}+\delta), y\Big)$$

$$\min_{\theta} \text{Loss}\Big(f_\theta(\mathbf{x}), y\Big)$$

How to "defend" against adversarial examples?

- **Standard training**:

  $$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \text{Loss}(f_\theta(\mathbf{x}^{(i)}), y^{(i)})$$

  neural network

  via (stochastic) gradient descent

- **Adversarial training** / robust optimization:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \max_{\delta \in \Delta} \text{Loss}\left(f_\theta(\mathbf{x}^{(i)} + \delta), y^{(i)}\right)$$

"adaptive data augmentation"

Adversarial training with stochastic gradient descent

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \boxed{\max_{\delta \in \Delta}} \text{Loss}\left(f_\theta(\boxed{\mathbf{x}^{(i)} + \delta}), y^{(i)}\right)$$

repeat until convergence:

1. sample a data point $(\mathbf{x}, y)$

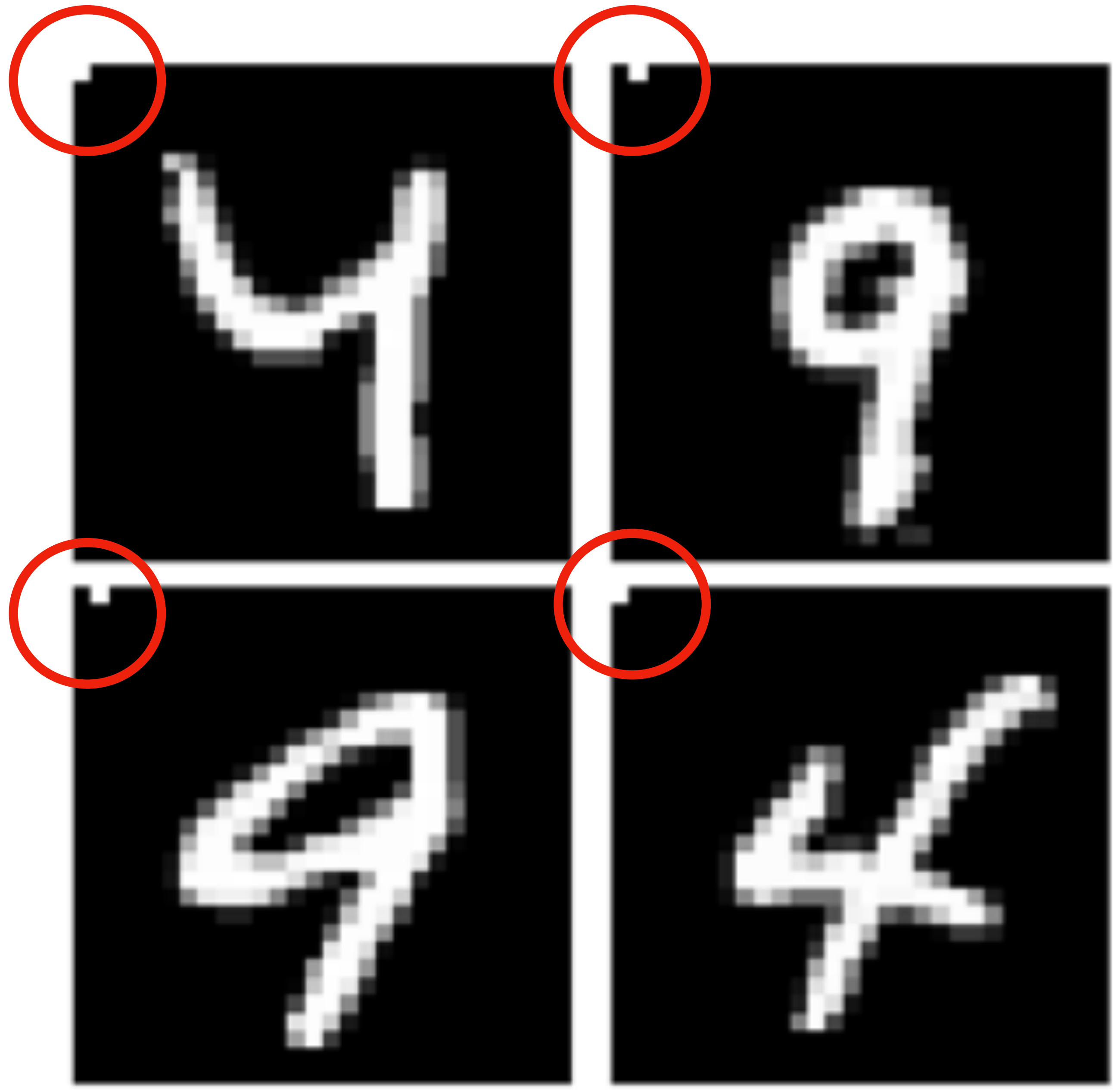
2. compute the optimal adversarial perturbation δ^* *(approximately)*

3. compute the gradient $g = \nabla_\theta \text{Loss}\left(f_\theta(\mathbf{x} + \delta^*), y\right)$

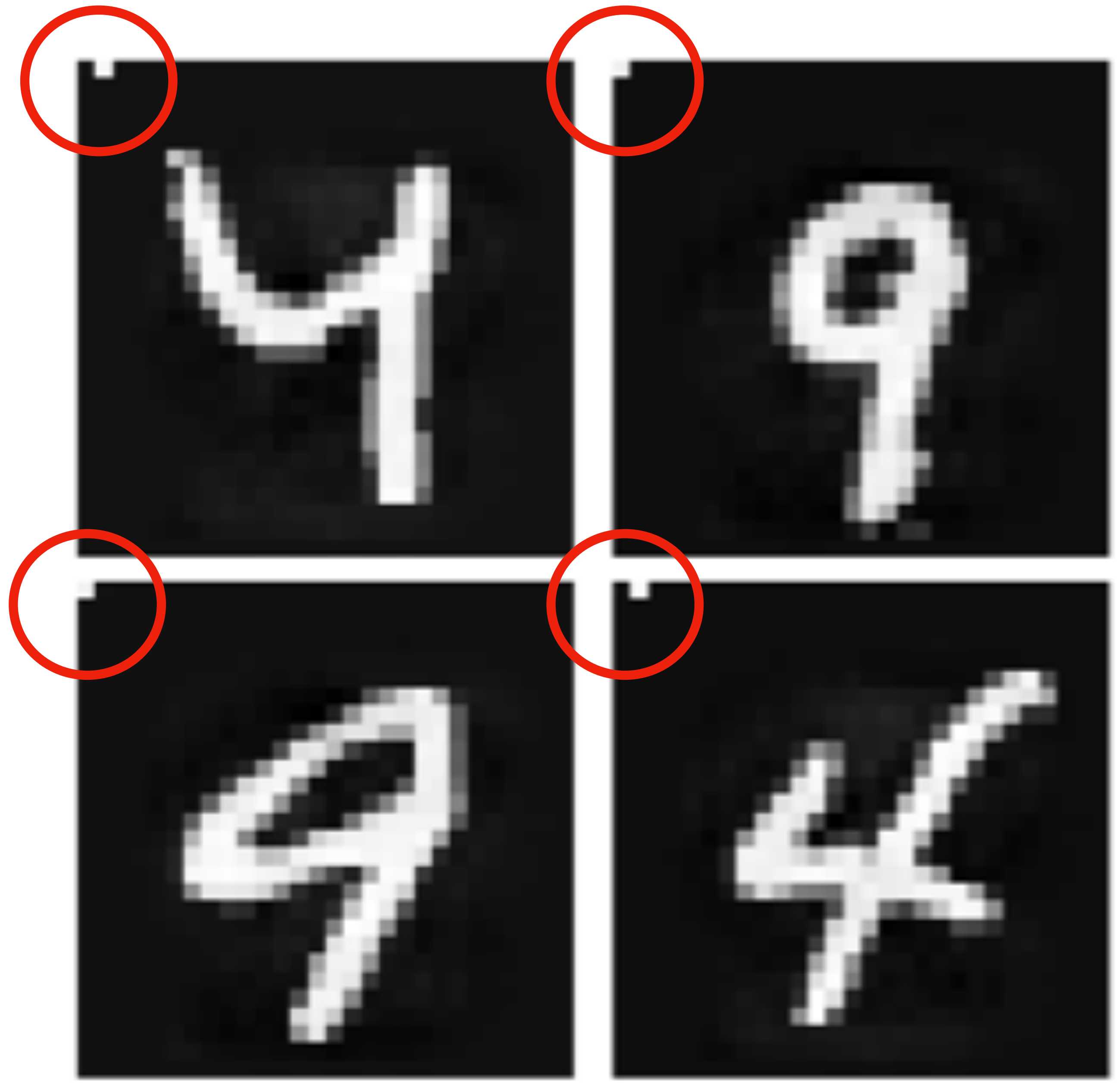
4. update θ with the gradient g

What do adversarial examples tell us?

- something about the input "features" that are critical for the model's decision
- Example:



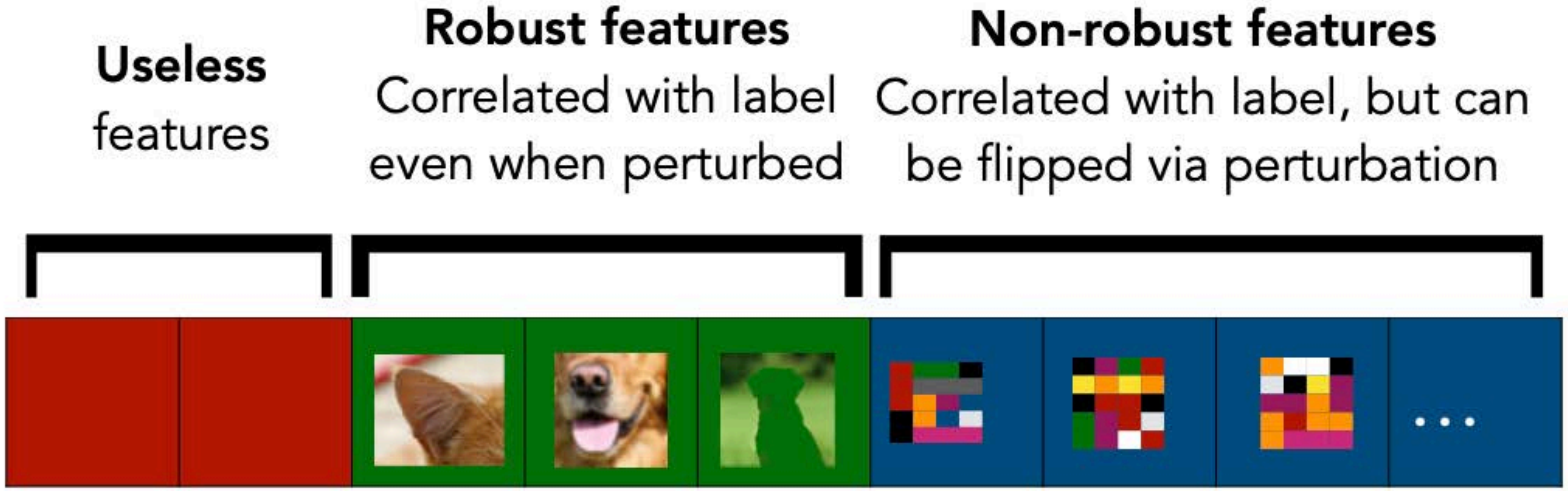
Training data:
classify 4 vs 9



Adversarial
perturbations

images: Hongzhou Lin

Predictive features

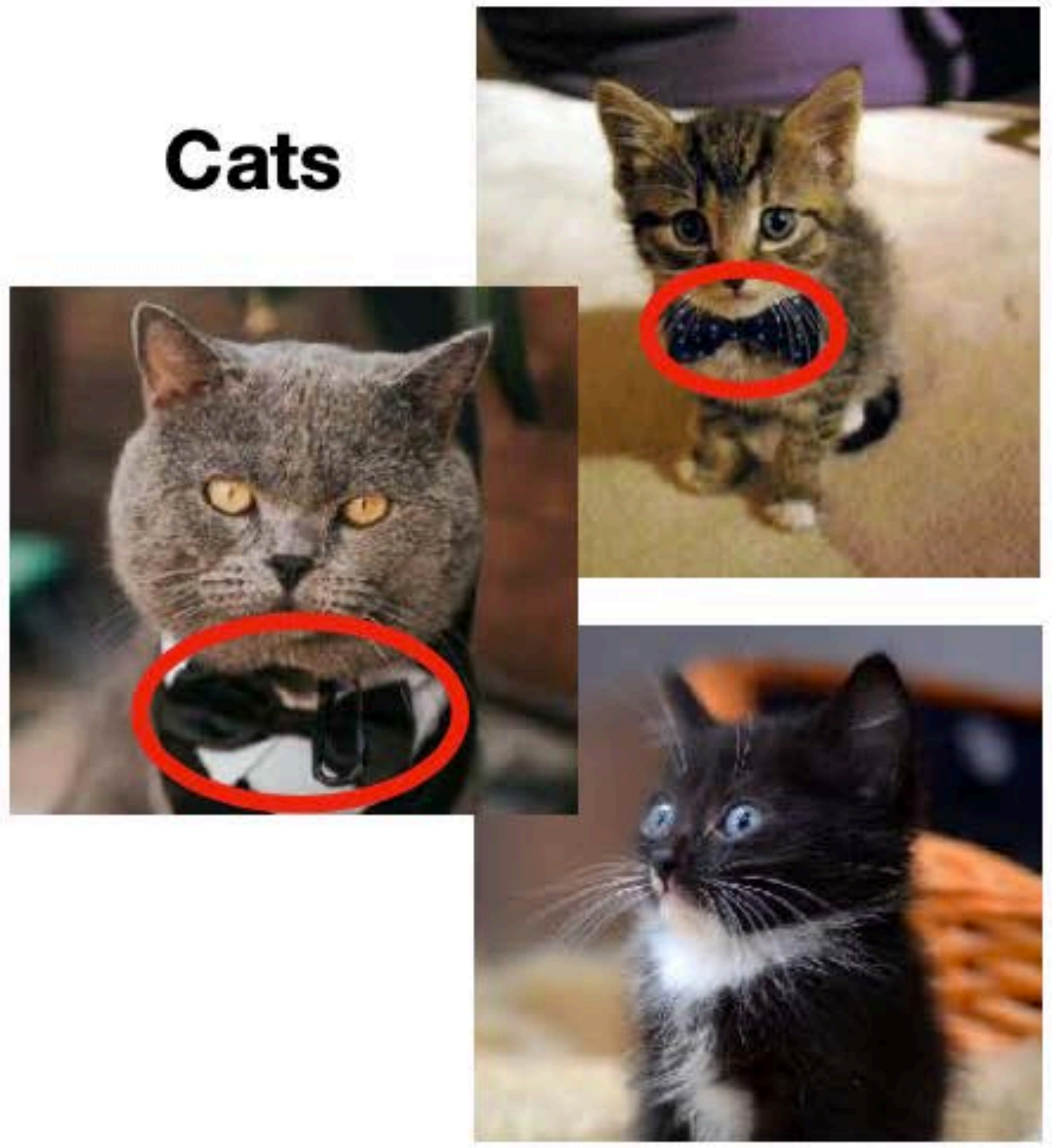


- Many features may be correlated with the label and hence predictive and help with accuracy, *beyond what humans would use.*

illustration: Aleksander Madry

Where do these correlations come from?

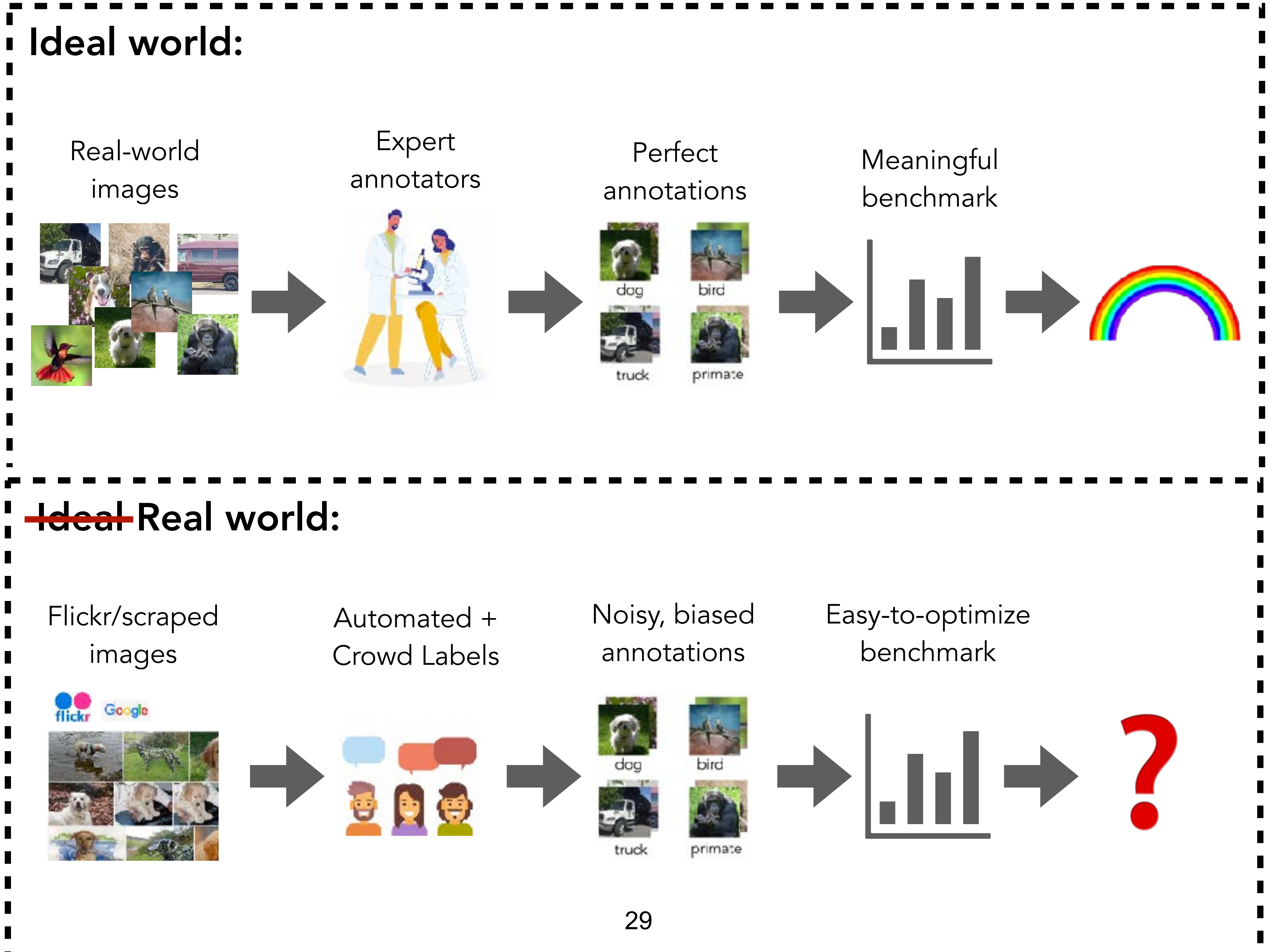
- Data



"Fish" from the ImageNet training set

Where do these correlations come from?

- …and how we create datasets



It's all "shortcuts"

- Shortcuts: features correlated with label in the training data, but not under realistic distribution shifts

- Models will use them and not generalize if features are no longer correlated

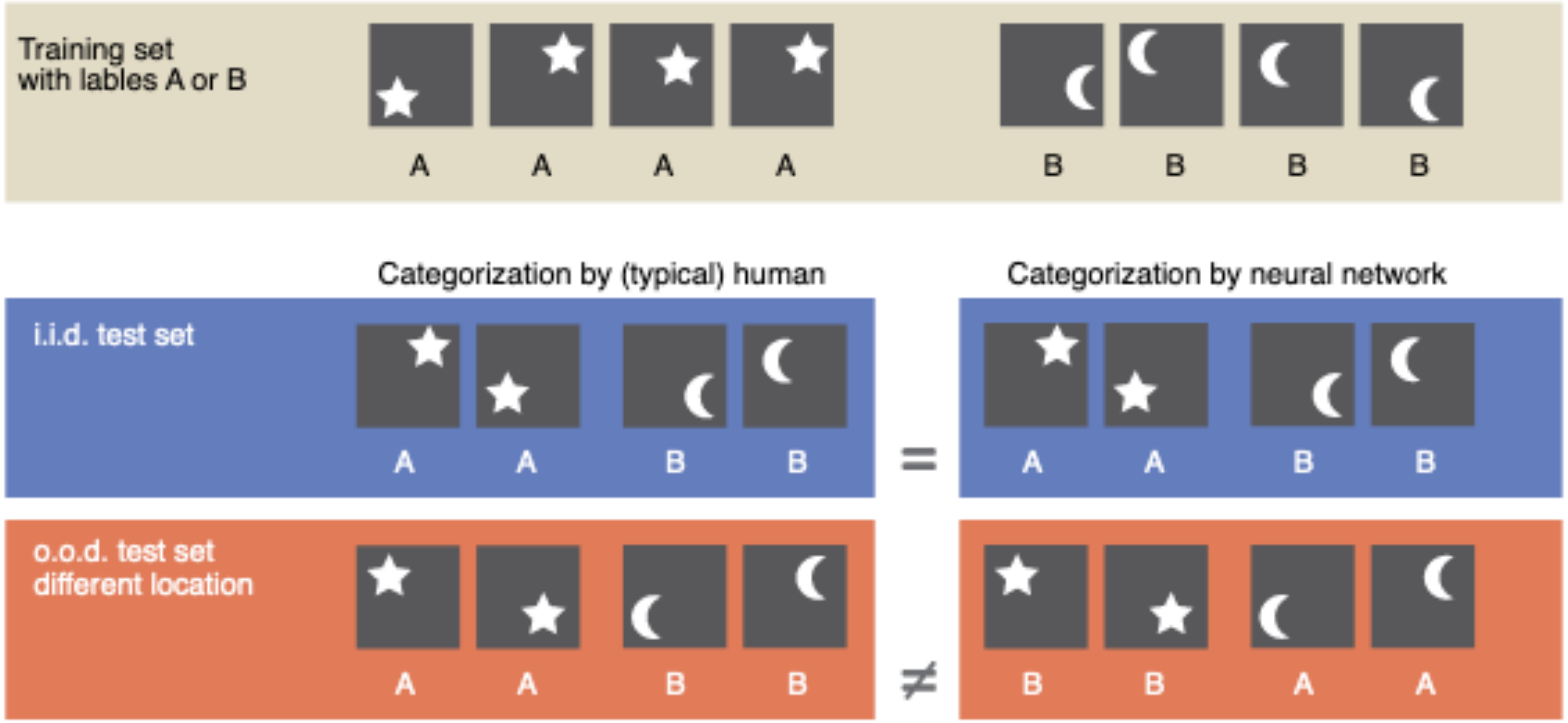


illustration: Geirhos et al 2020

It's all "shortcuts"

- Shortcuts: features correlated with label in the training data, but not under realistic distribution shifts
- Models will use them and not generalize if features are no longer correlated
- This is related to data, not models: ***adversarial examples transfer across models trained on the same dataset***

What can these shortcuts look like?

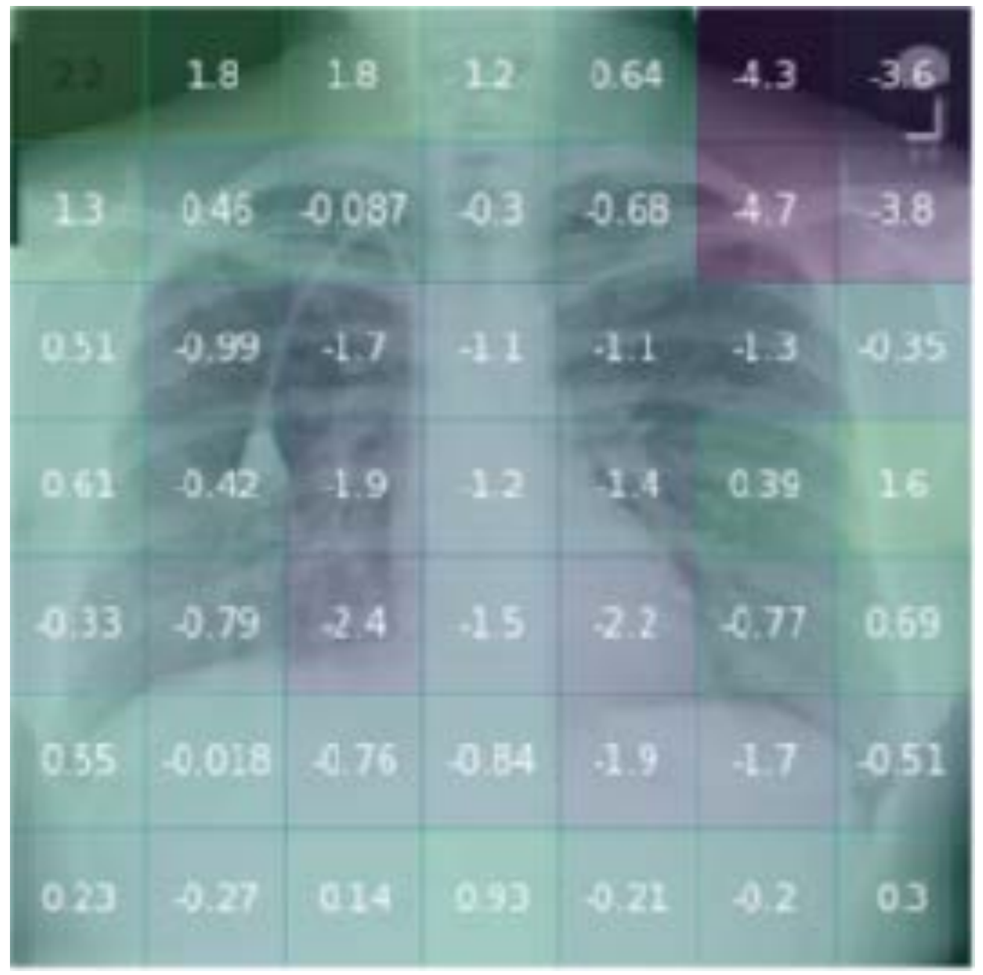
A herd of sheep grazing on a l
Tags: grazing, sheep, mountai

Left: A man is holding a dog in his hand
Right: A woman is holding a dog in her hand
Image: @SouperSarah

flock of birds flying in the air
up of giraffe standing next to a tree
/www.flickr.com/photos/gratapictures - CC-BY-NC

images: https://www.aiweirdness.com/do-neural-nets-dream-of-electric-18-03-02/

What can these shortcuts look like?



"CNNs were able to detect where an x-ray was acquired [...] and calibrate predictions accordingly."

[Zech et al. 2018]

"...if an image had a ruler in it, the algorithm was more likely to call a tumor malignant..."

[Esteva et al. 2017]

not all predictive patterns are desirable

Many more…

illustration: Geirhos et al 2020, Shortcut learning in deep neural networks

Transformers Learn Shortcuts to Automata

Bingbin Liu[1*] Jordan T. Ash[2] Surbhi Goel[2,3] Akshay Krishnamurthy[2] Cyril Zhang[2]

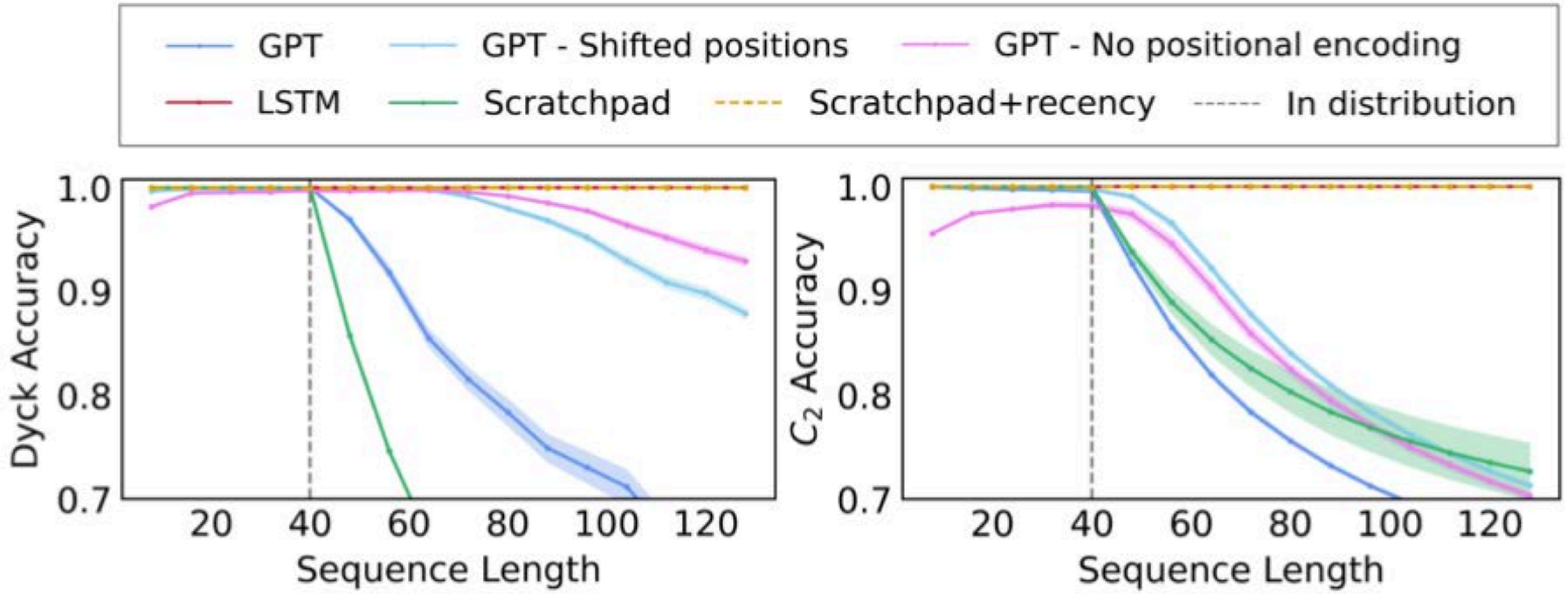
[1]Carnegie Mellon University [2]Microsoft Research NYC [3]University of Pennsylvania

bingbinl@cs.cmu.edu, {ash.jordan, goel.surbhi, akshaykr, cyrilzhang}@microsoft.com

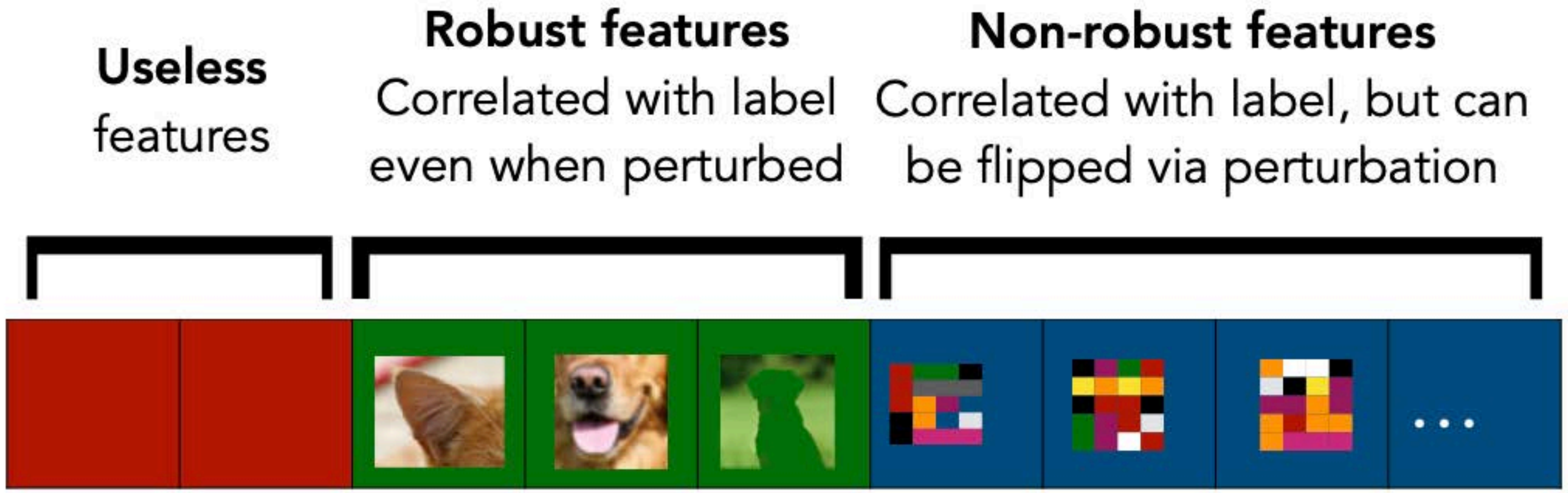
Abstract

Algorithmic reasoning requires capabilities which are most naturally understood through recurrent models of computation, like the Turing machine. However, Transformer models, while lacking recurrence, are able to perform such reasoning using far fewer layers than the number of reasoning steps. This raises the question: *what solutions are these shallow and non-recurrent models finding?* We investigate this question in the setting of learning automata, discrete dynamical systems naturally suited to recurrent modeling and expressing algorithmic tasks. Our theoretical results completely characterize *shortcut solutions*, whereby a shallow Transformer with only $o(T)$ layers can exactly replicate the computation of an automaton on an input sequence of length T. By representing automata using the algebraic structure of their underlying transformation semigroups, we obtain $O(\log T)$-depth simulators for all automata and $O(1)$-depth simulators for all automata whose associated groups are sc
synthetic experiments by training Transformers to simulate a wide vari
shortcut solutions can be learned via standard training. We further in
solutions and propose potential mitigations.

parallel solutions generalize within-distribution,
but not out-of-distribution



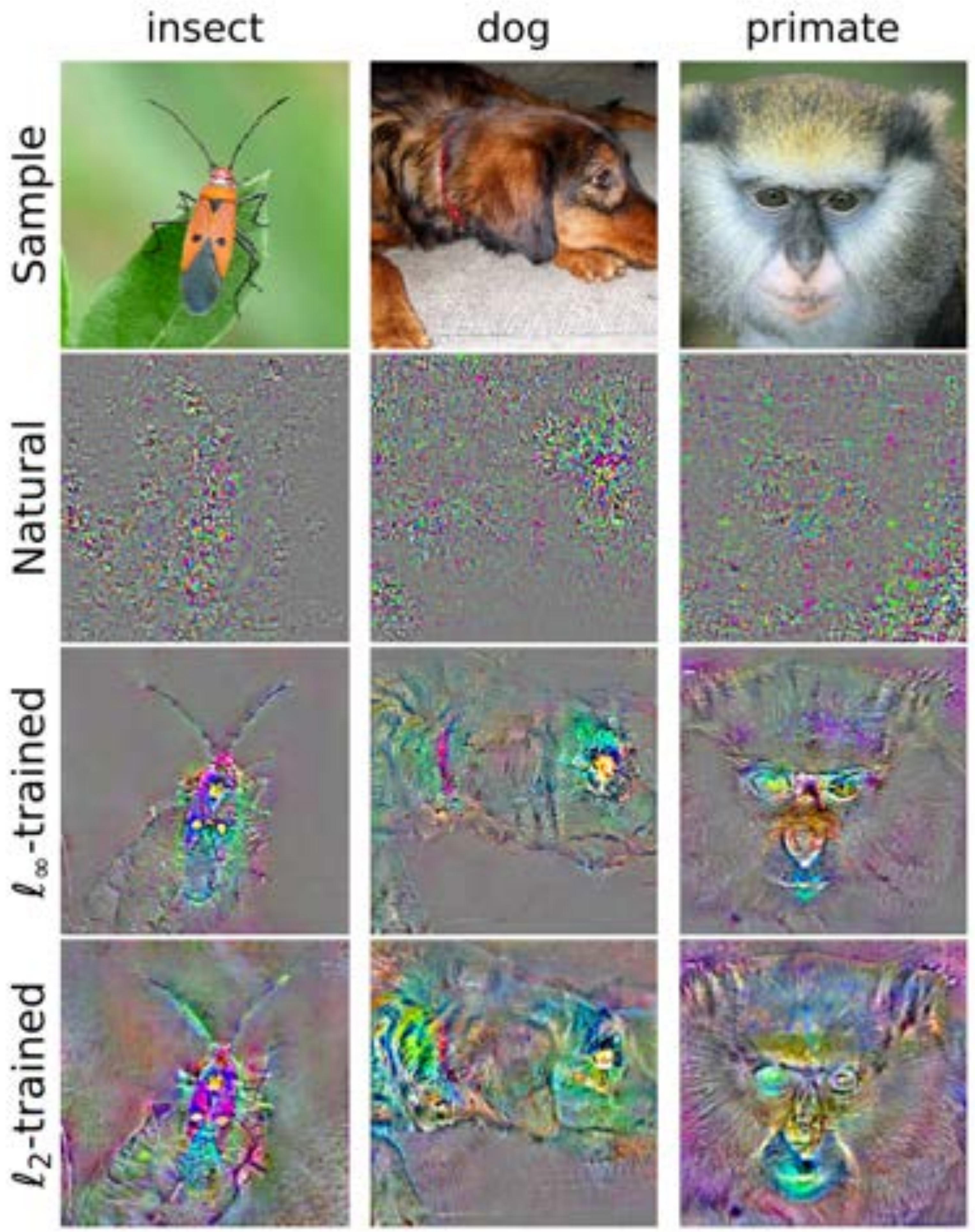
Effect of adversarial training



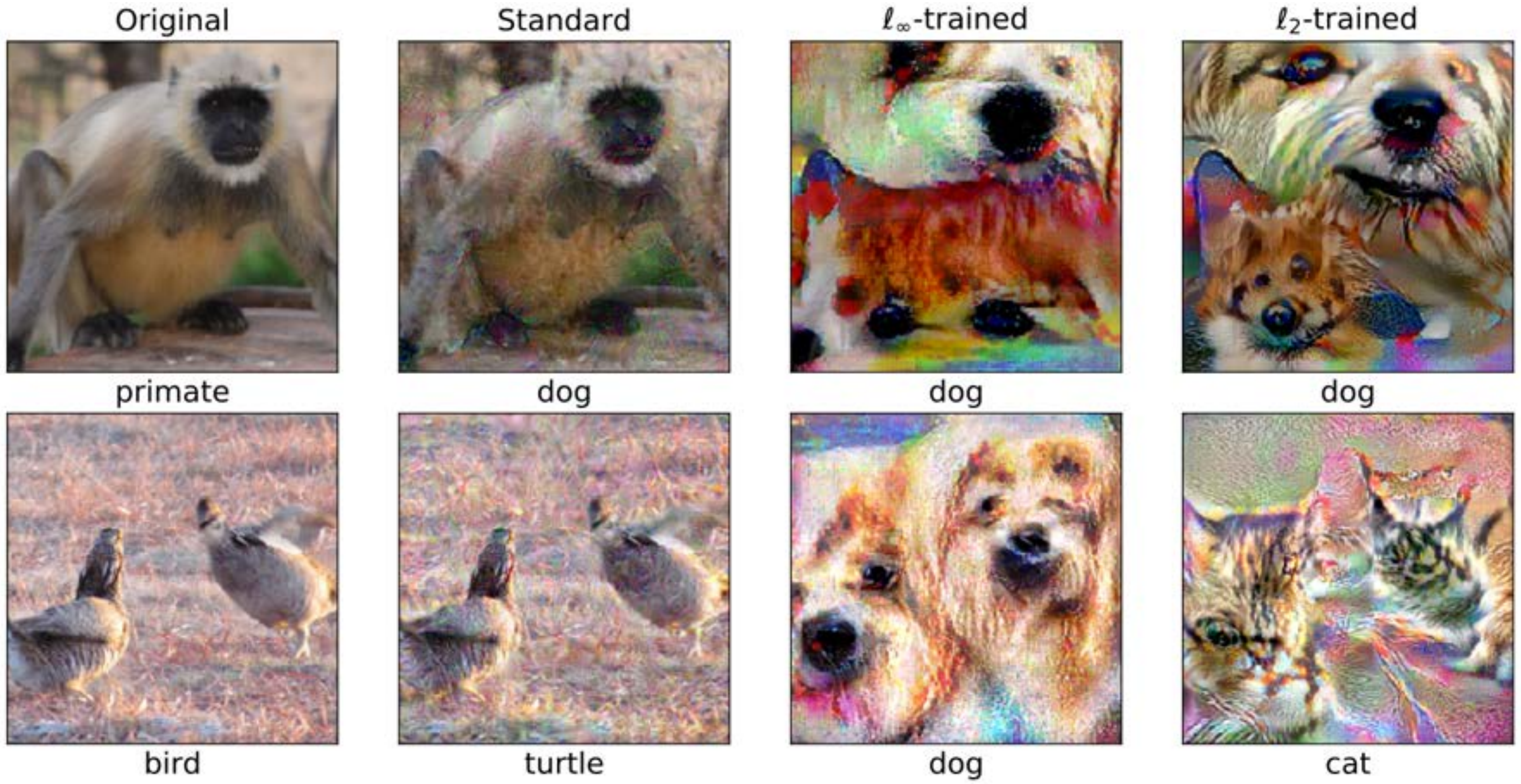
- model output should be stable under adversarial perturbations
  => teaches invariance to non-robust features

Effect of adversarial training

Loss gradients with respect to input pixels (most important features) show: robust model relies less on “non-robust” features, and more on human-intuitive features



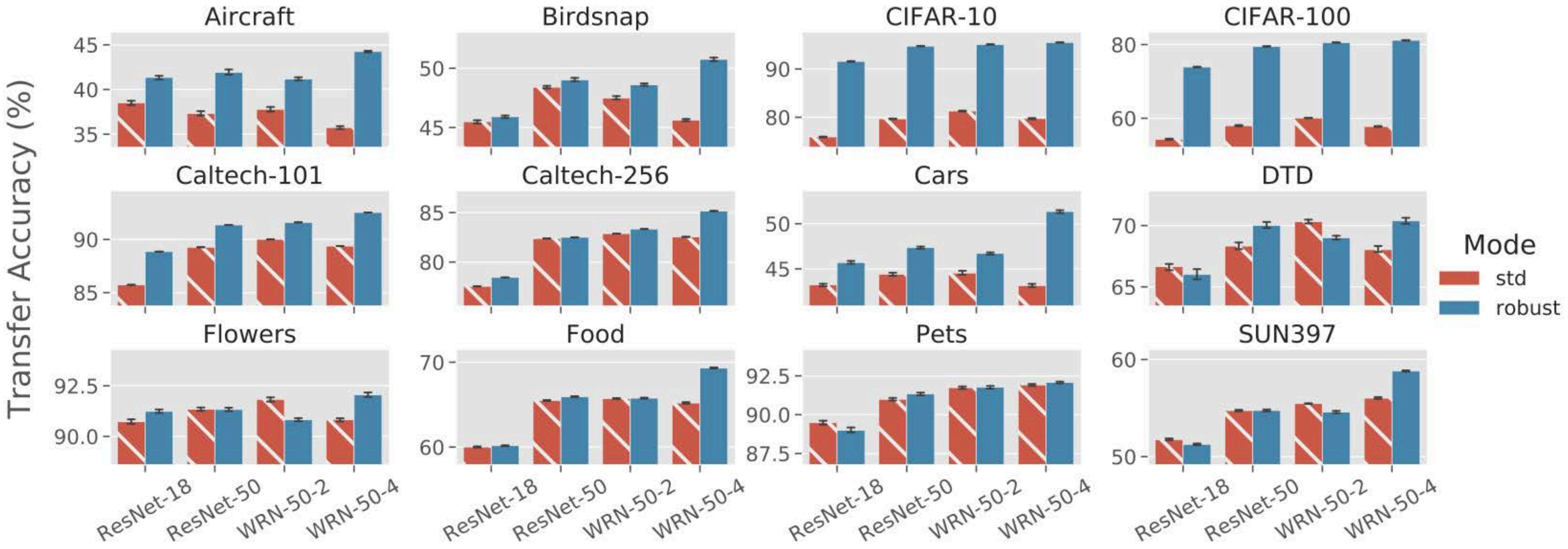
Adversarial examples for standard and robust models



(Tsipras et al. 2019, Robustness may be at odds with accuracy.)

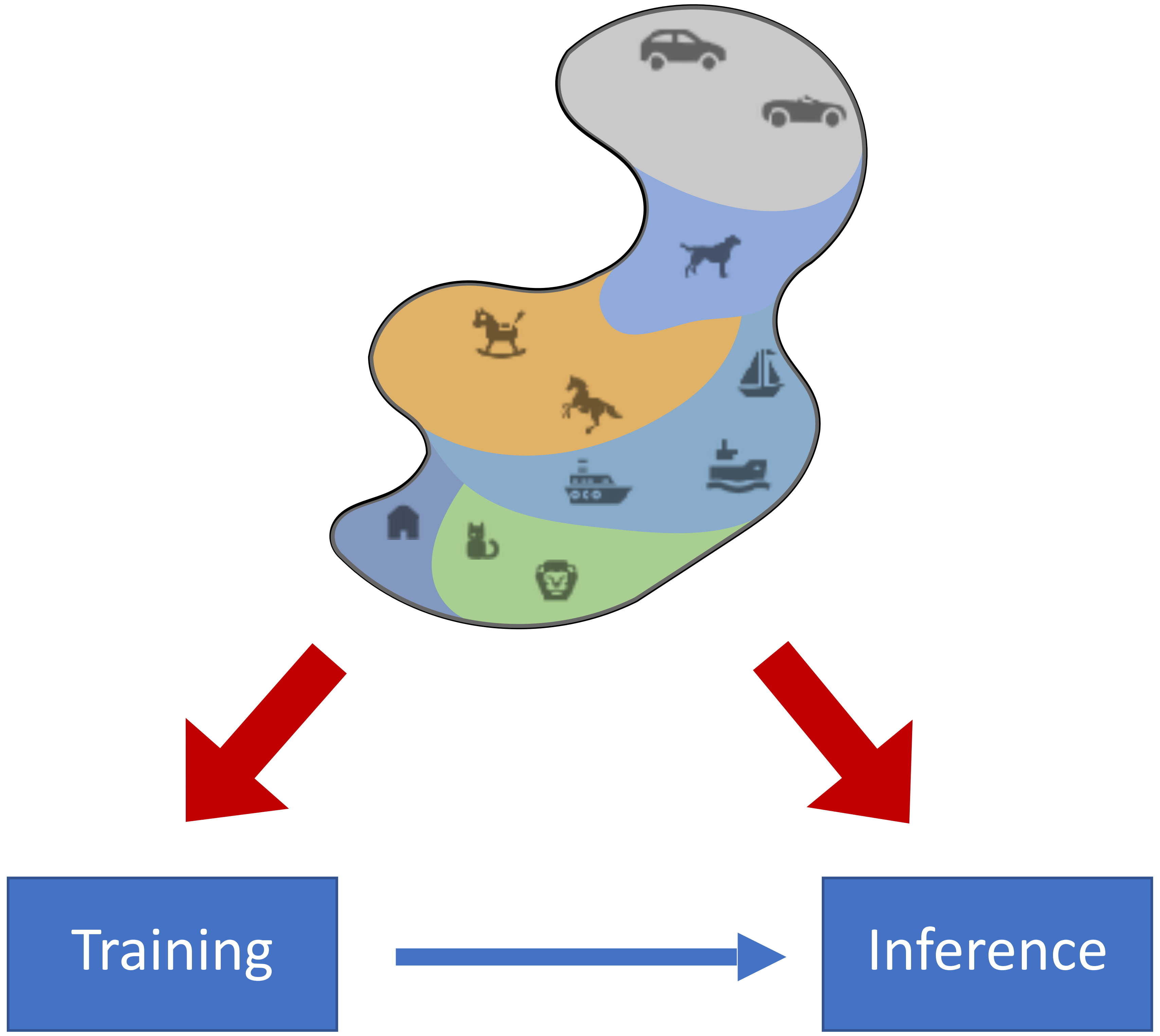
Effect of adversarial training: transfer learning

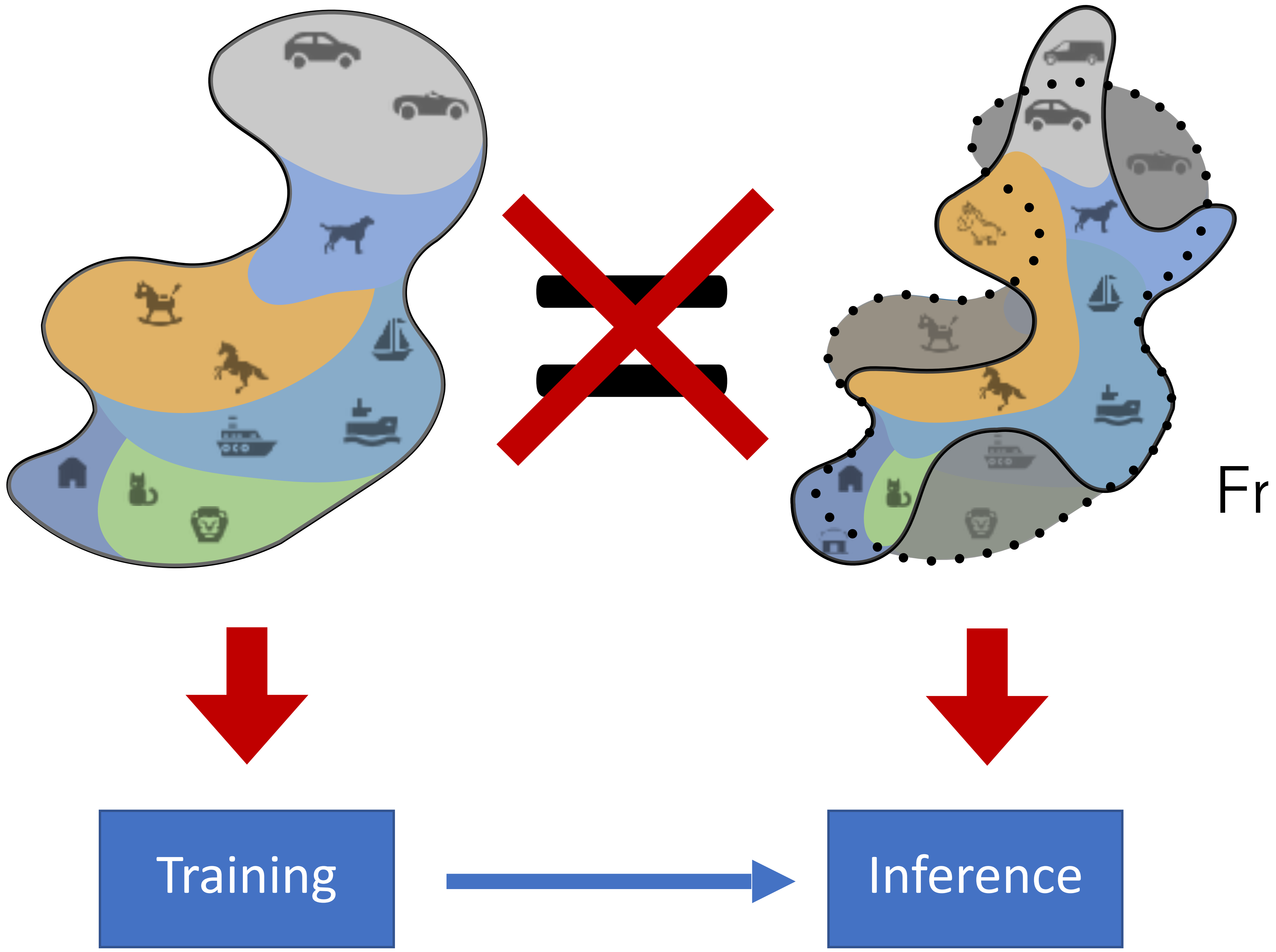
- adversarially trained models transfer better to other datasets

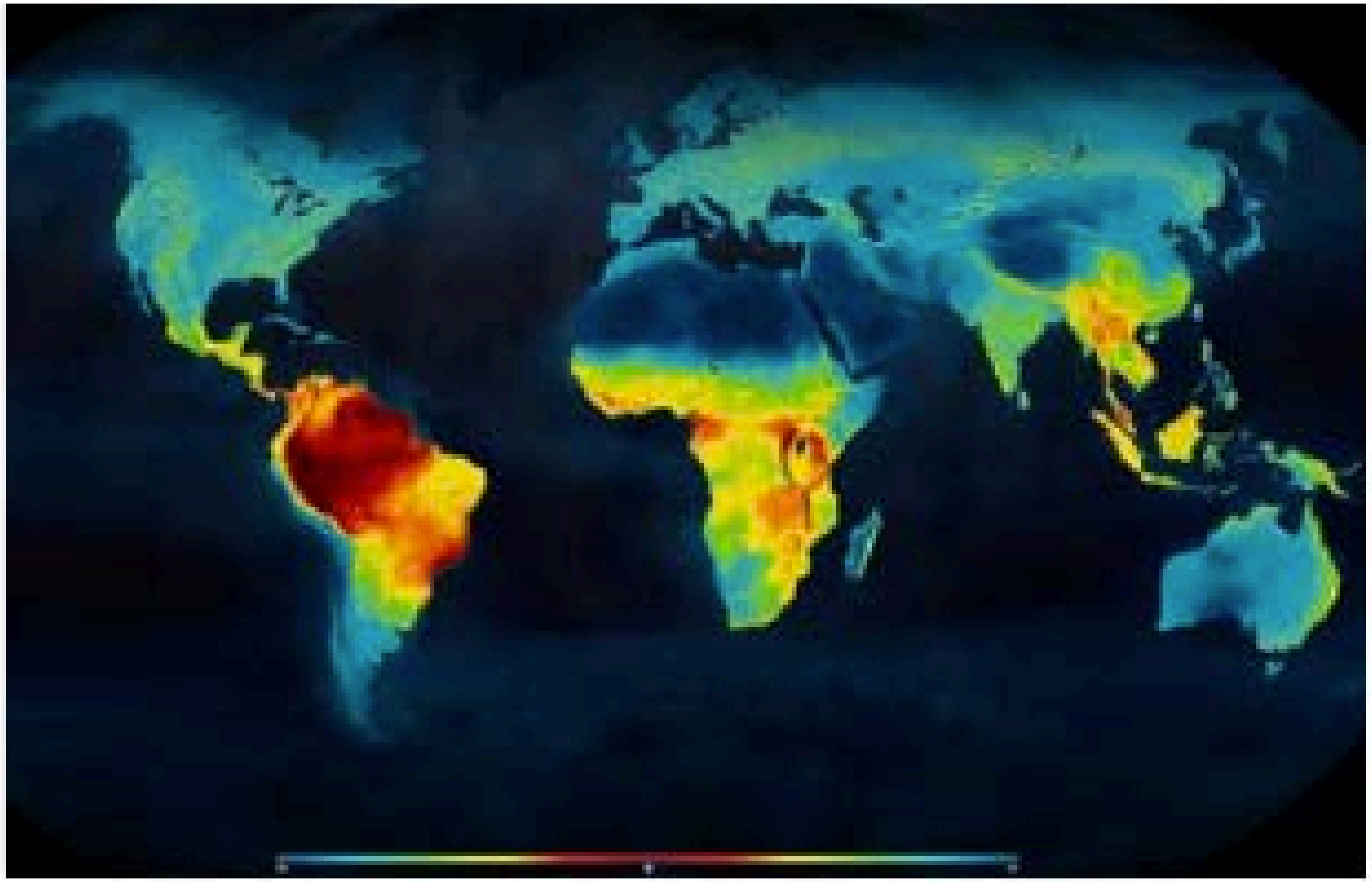


(Salman et al. 2020, Do adversarially robust ImageNet models transfer better?)

Distribution shifts


Training
Inference


Fr
Training
Inference

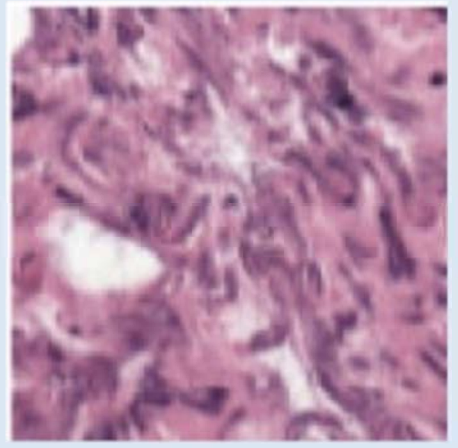
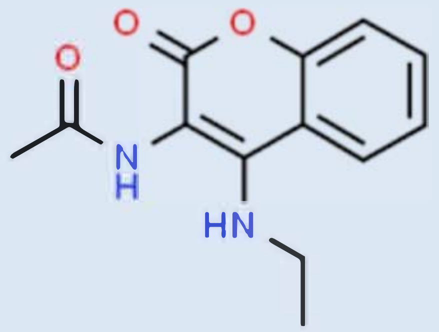
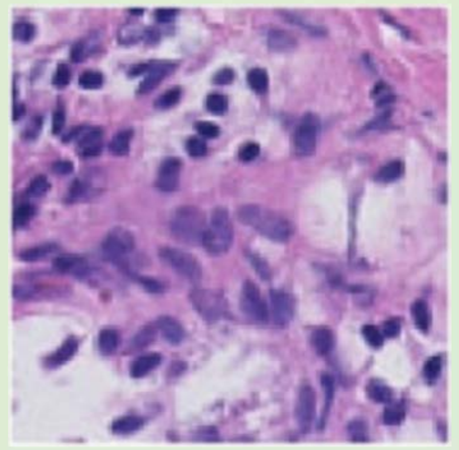
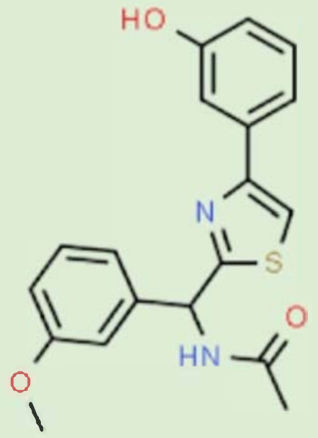


Map of global biodiversity

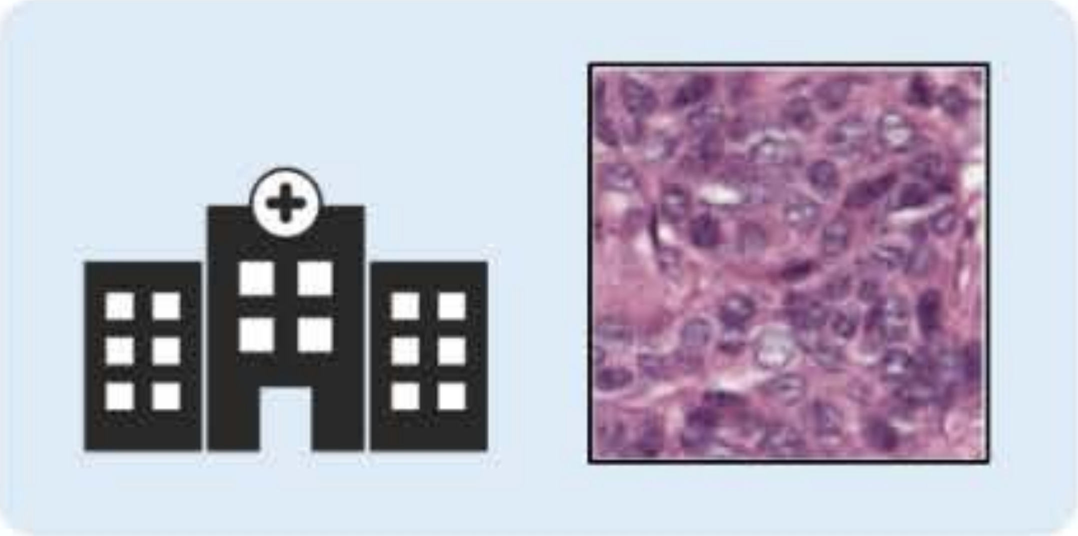
Species occurrence data in GBIF

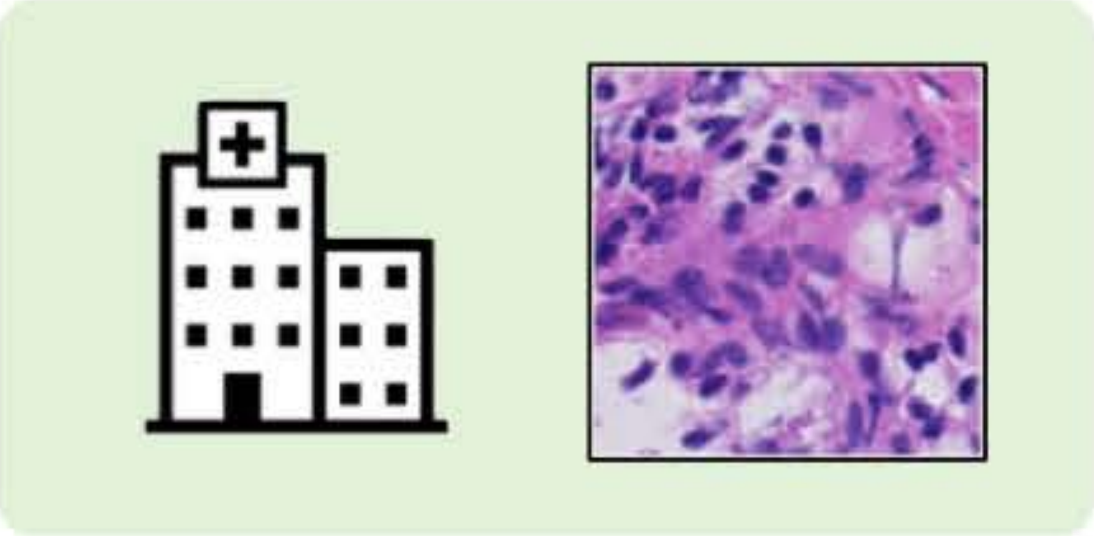
WILDS

Pang Wei Koh*, Shiori Sagawa*, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang

	Camelyon17	iWildCam	PovertyMap	FMoW	Amazon	CivilComments	OGB-MolPCBA
Shift	Hospitals	Locations	Countries	Time	Users	Demographics	Scaffold
Train					Overall a solid package that has a good quality of construction for the price.	What do Black and LGBT people have to do with bicycle licensing?	
Test					I *loved* my French press, it's so perfect and came with all this fun stuff!	As a Christian, I will not be patronizing any of those businesses.	
Adapted from	Bandi et al. 2018	Beery et al. 2020	Yeh et al. 2020	Christie et al. 2018	Ni et al. 2019	Borkan et al. 2019	Hu et al. 2020

shifts across hospitals in histopathology





ID accuracy 93.2% → **-22.9%** → **OOD accuracy** 70.3%

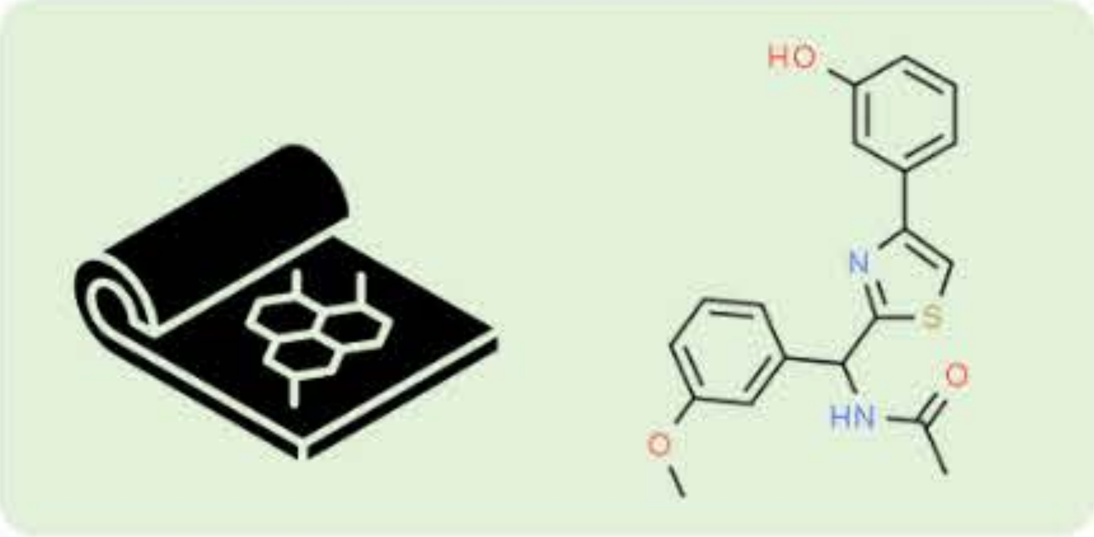
shifts across time in satellite imagery

ID accuracy 48.6% → **-16.3%** → **OOD accuracy** 32.3%

shifts across regions in wheat head detection

ID accuracy 63.3% → **-13.7%** → **OOD accuracy** 49.6%

shifts across scaffold in bioassay prediction



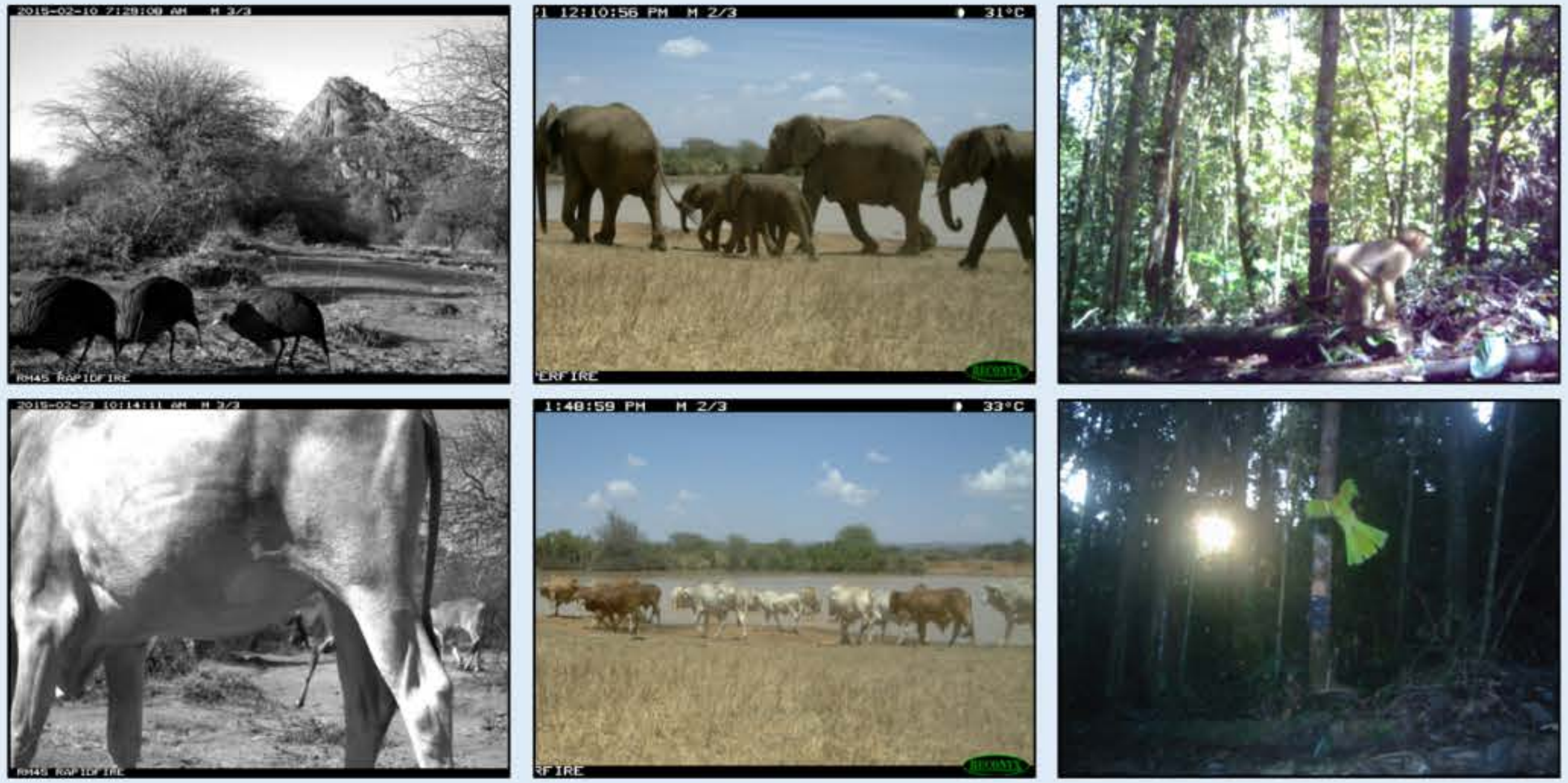
ID AP 34.4% → **-7.2%** → **OOD AP** 27.2%

[Koh et al., 2021]

Slide from Shiori Sagawa

Training data

Camera 1 Camera 2 ... Camera 245



Out-of-distribution (OOD) test data

Camera 246

...

Control: In-distribution (ID) test data

Camera 1 Camera 2 ... Camera 245

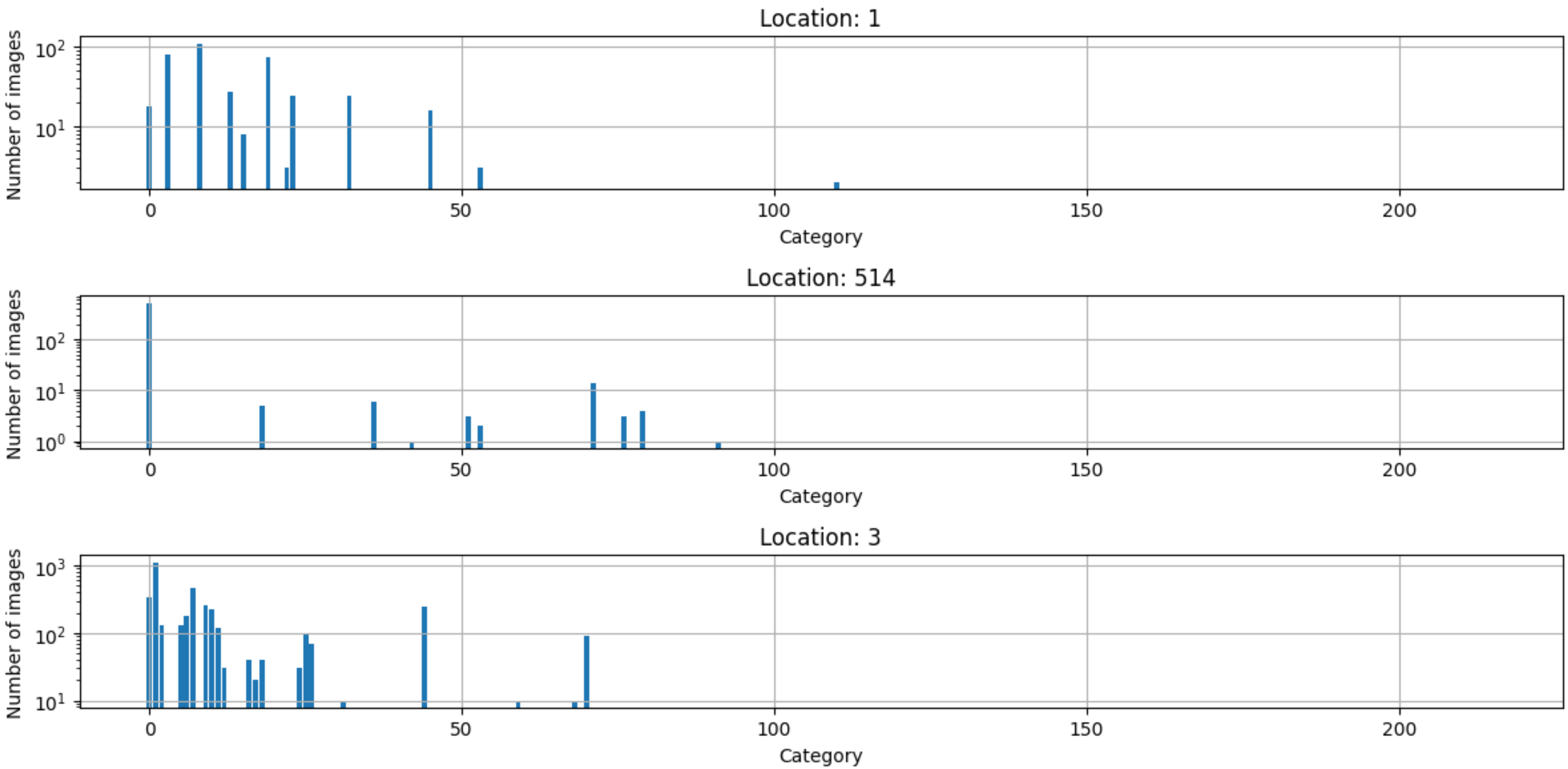
Macro F1

ID 47.0% → -16.0% → OOD 31.0%

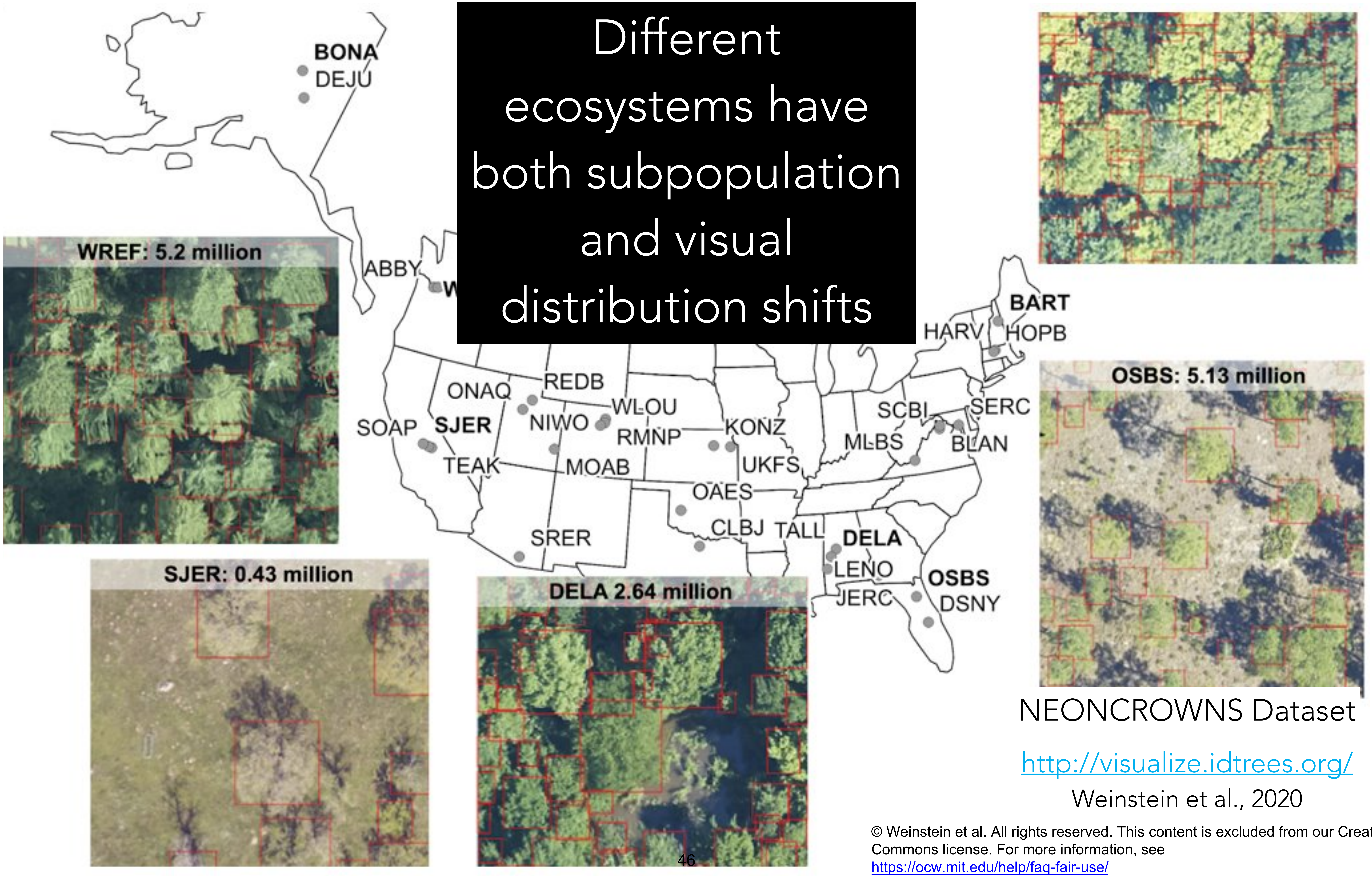
[Beery et al., 2020; Koh et al., 2021]

Slide from Shiori Sagawa

Class distribution is different for each static sensor location



Different ecosystems have both subpopulation and visual distribution shifts

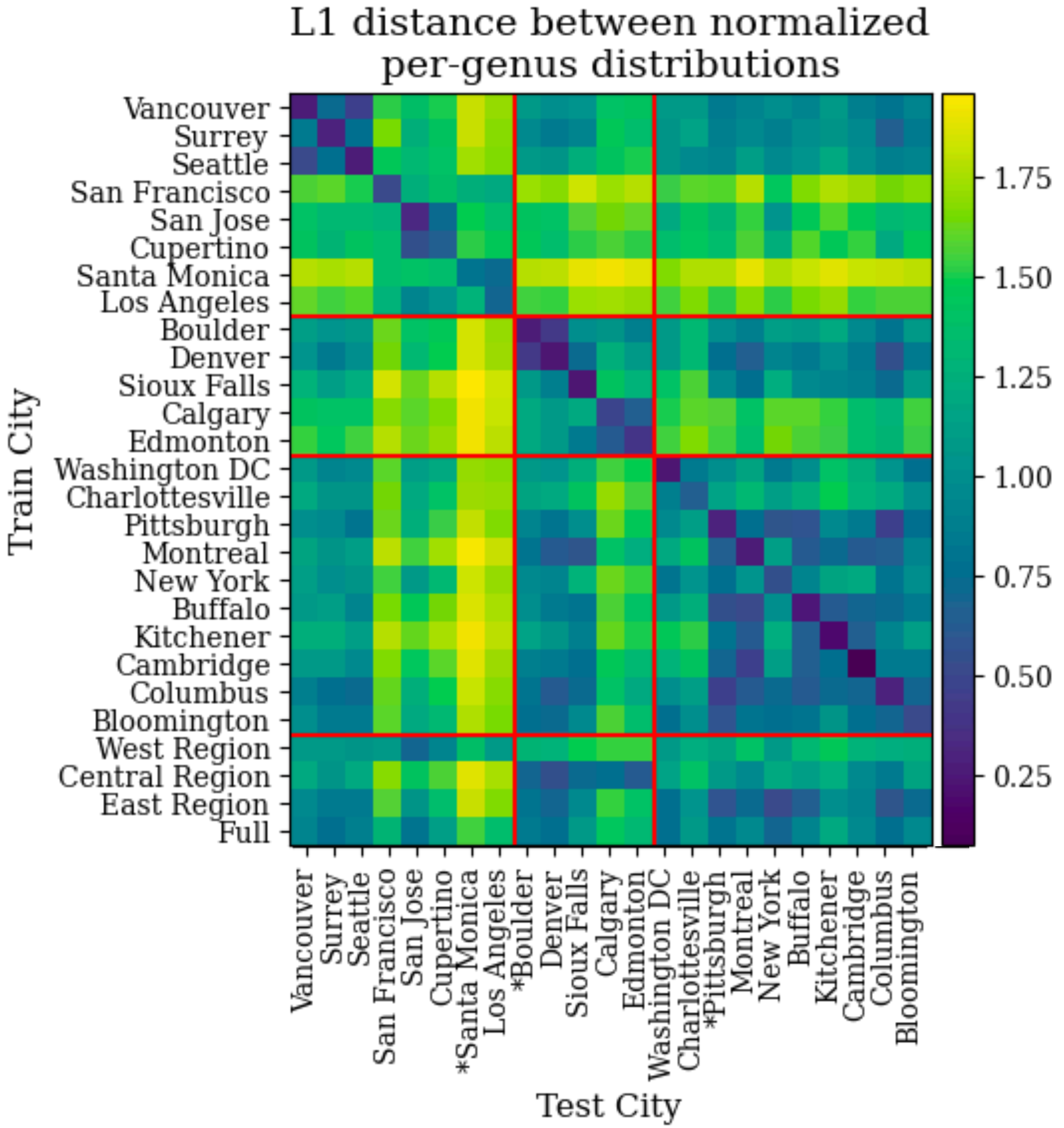


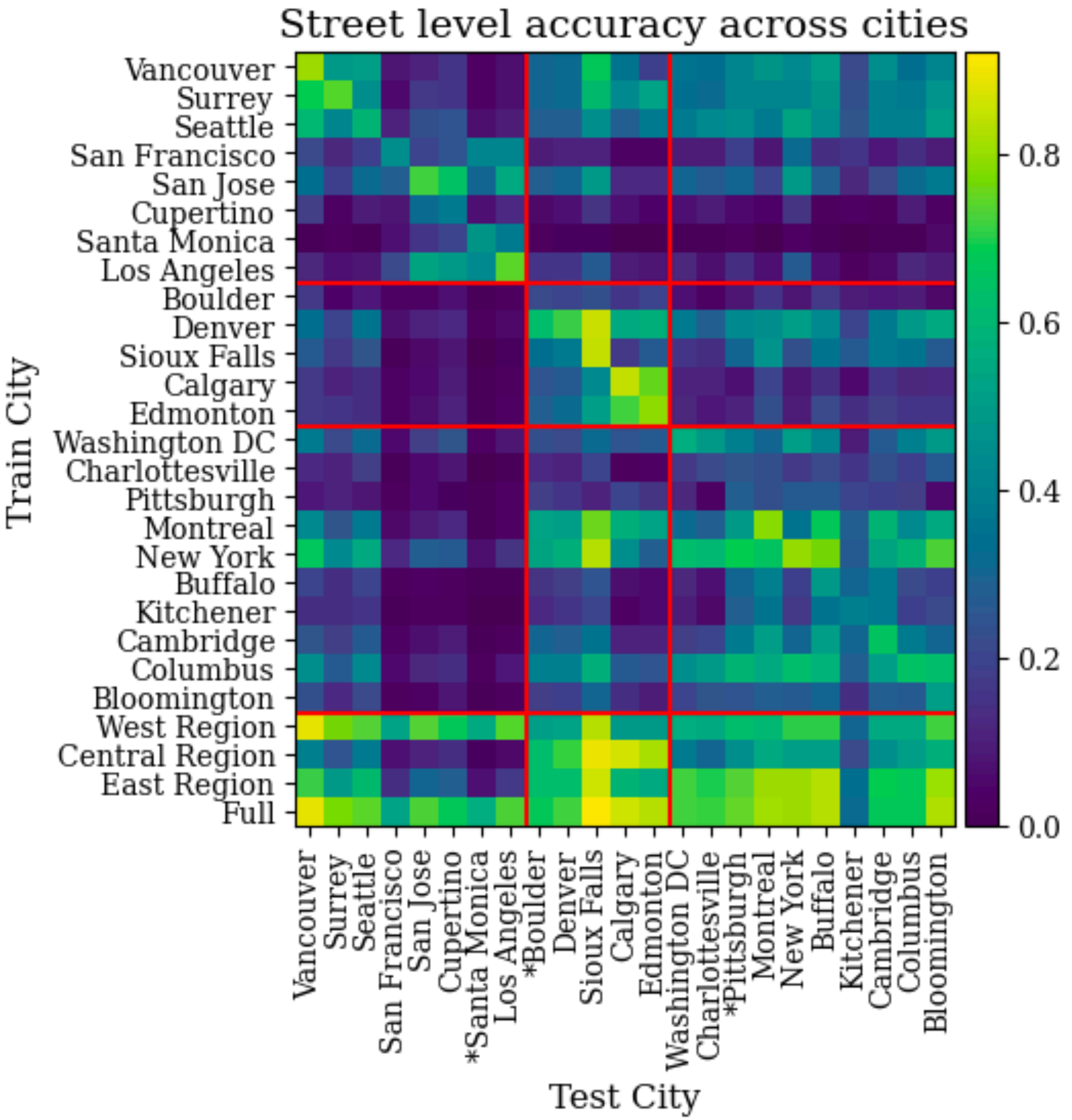
NEONCROWNS Dataset

http://visualize.idtrees.org/

Weinstein et al., 2020

Performance has strong correlation with subpop. distribution similarity

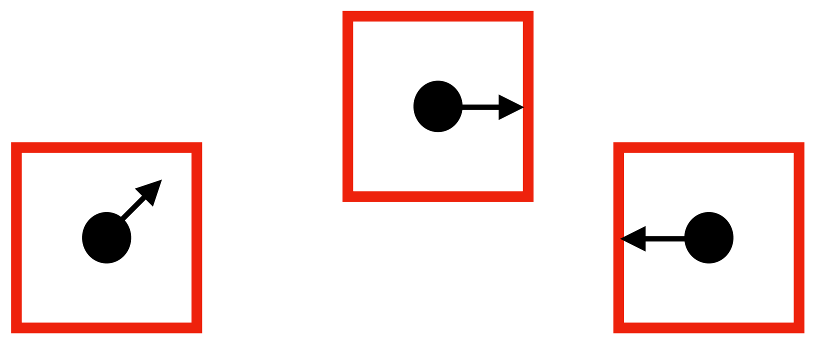




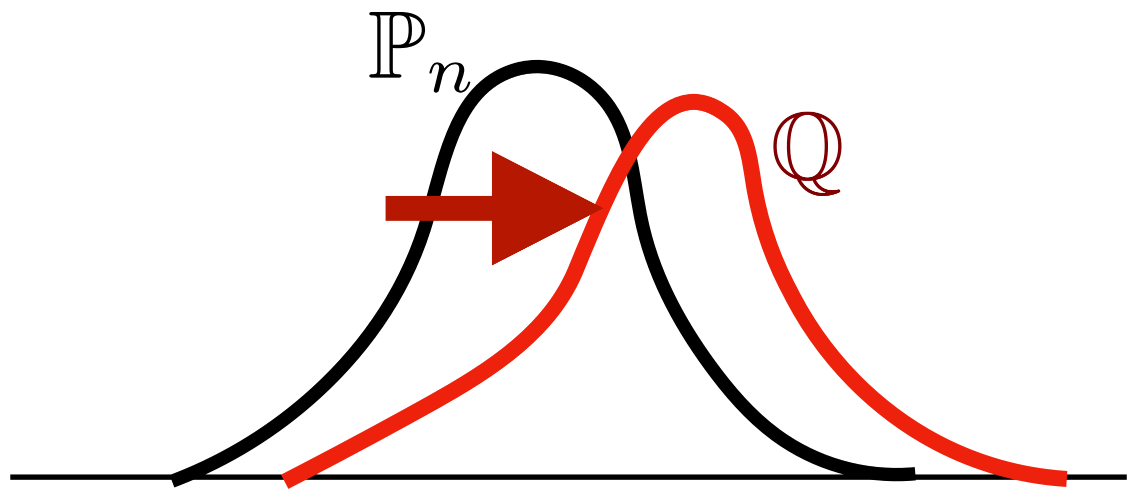
What to do about distribution shift?

One path: distributionally robust optimization

- So far: allowed to perturb each datapoint by a limited amount



- Alternative: we can perturb the entire training distribution (sample) by a certain amount, together



Distributionally robust optimization

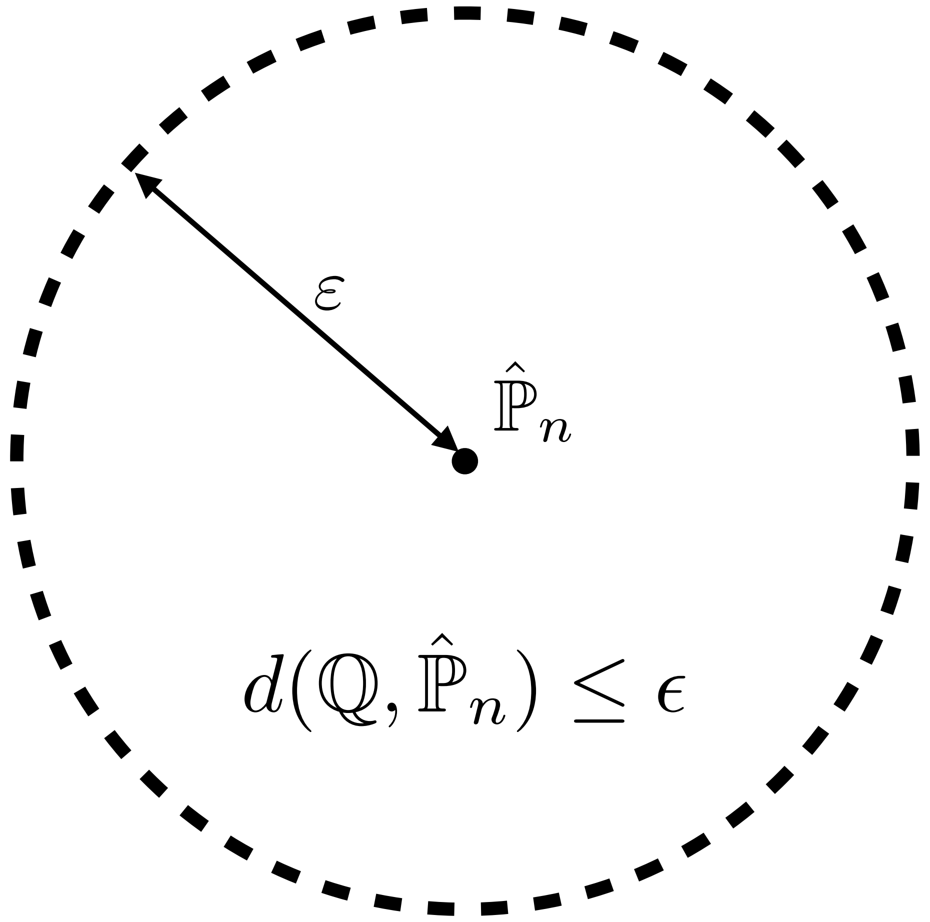
- Standard training: $$\frac{1}{n}\sum_{i=1}^{n} \text{Loss}(f_\theta(\mathbf{x}^{(i)}), y^{(i)}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathbb{P}_n}[\text{Loss}(f_\theta(\mathbf{x}), y)]$$

allow a small perturbation of training sample (discrete distribution)

- Distributionally robust optimization (DRO):

$$\min_{\theta} \max_{\mathbb{Q},\, D(\mathbb{Q},\mathbb{P}_n)<\epsilon} \mathbb{E}_{(\mathbf{x},y)\sim\mathbb{Q}}[\text{Loss}(f_\theta(\mathbf{x}), y)]$$

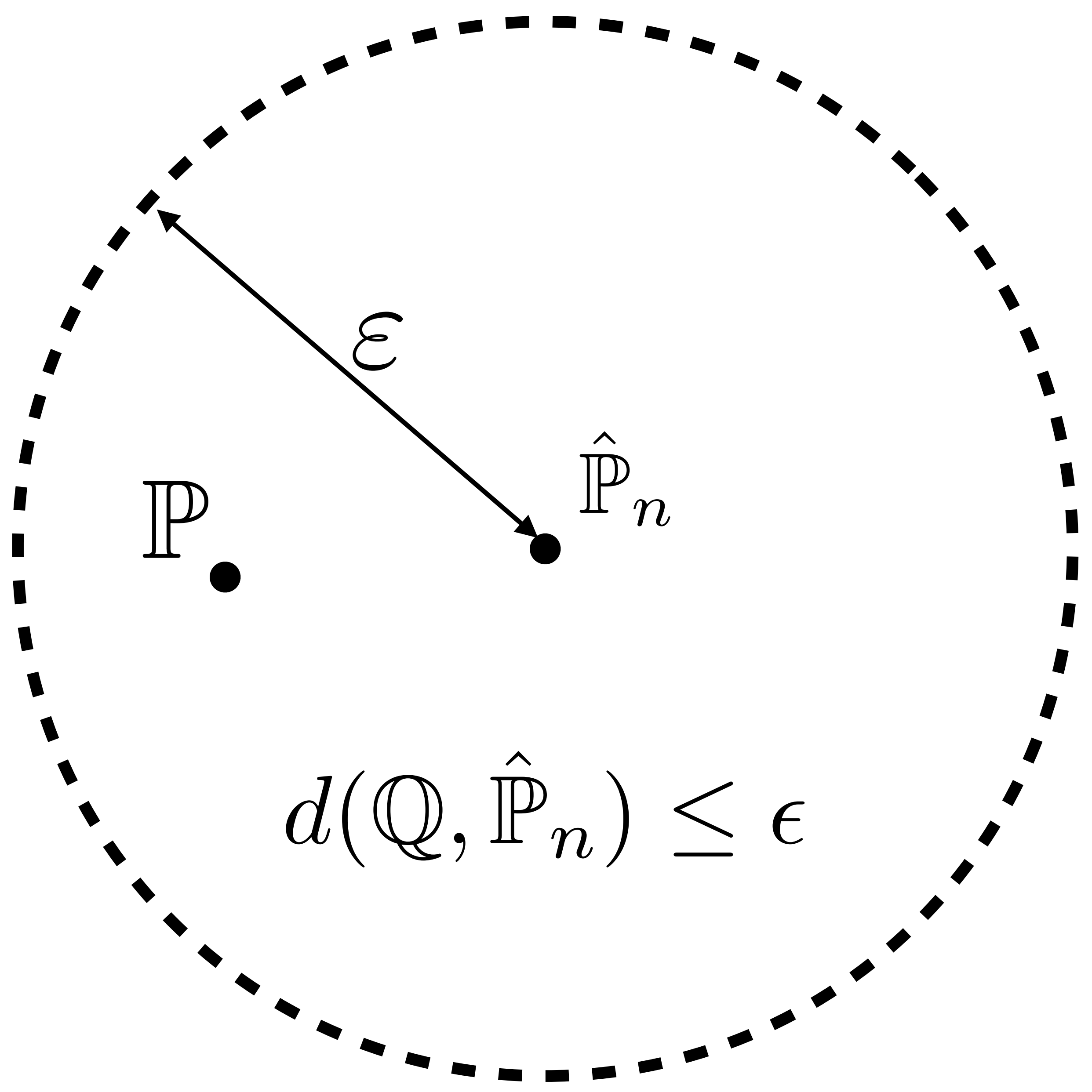
e.g. re-weight or perturb training data points



- Various choices of measuring "distance" between probability distributions: χ^2-distance, Wasserstein distance, maximum mean discrepancy (MMD)…

DRO and generalization

$$\min_{\theta} \max_{\mathbb{Q},\, D(\mathbb{Q},\mathbb{P}_n)<\epsilon} \mathbb{E}_{(\mathbf{x},y)\sim\mathbb{Q}}[\mathrm{Loss}(f_\theta(\mathbf{x}), y)]$$



- DRO optimizes for a set of training data sets/ distributions
- Say underlying data distribution is $\mathbb{P}$
- Empirical training data is $\hat{\mathbb{P}}_n$
- If $D(\mathbb{P}, \hat{\mathbb{P}}_n) < \epsilon$, then we are guaranteed to perform well on $\mathbb{P}$ too, i.e., generalize!

Application: DRO and class imbalance

- Assume population has K sub-groups (example: K=2).
- Usually: minimize "Empirical Risk" (average error)

$$\min_{\theta} \frac{1}{n}\Big(\underbrace{\sum_{i \text{ in group } 1} \text{Loss}(x_i;\theta)}_{80\%} + \underbrace{\sum_{j \text{ in group } 2} \text{Loss}(x_j;\theta)}_{20\%} \Big)$$

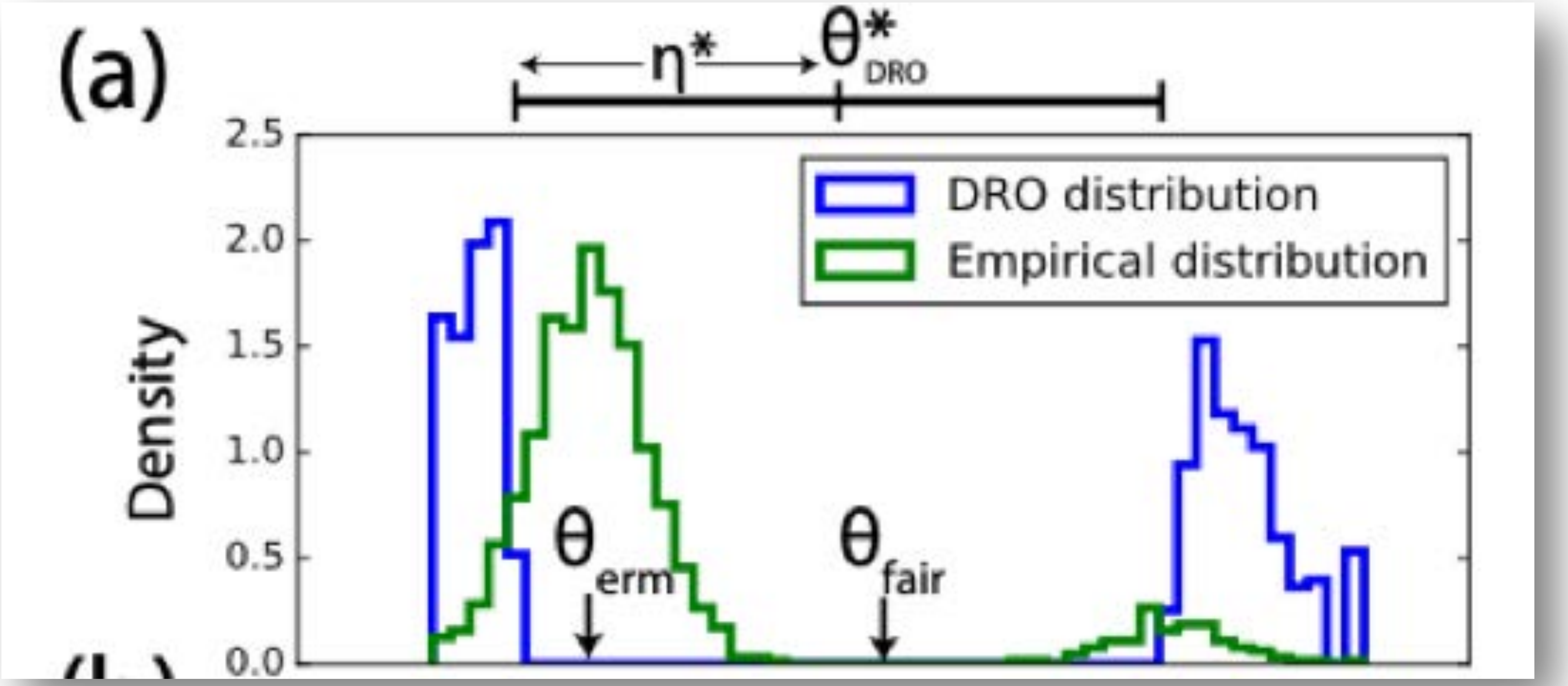
- Here, 50% error on minority group makes only 10% average error.
  (+ statistical patterns for minority may be different)

- We can "ignore" minority group and still get decent loss!

DRO and class imbalance

- Idea: automatically re-weight data via DRO
  => pay more attention to minority class

$$\min_{\theta} \max_{\mathbb{Q},\, D(\mathbb{Q},\mathbb{P}_n)<\epsilon} \mathbb{E}_{(\mathbf{x},y)\sim\mathbb{Q}}[\text{Loss}(f_\theta(\mathbf{x}), y)]$$



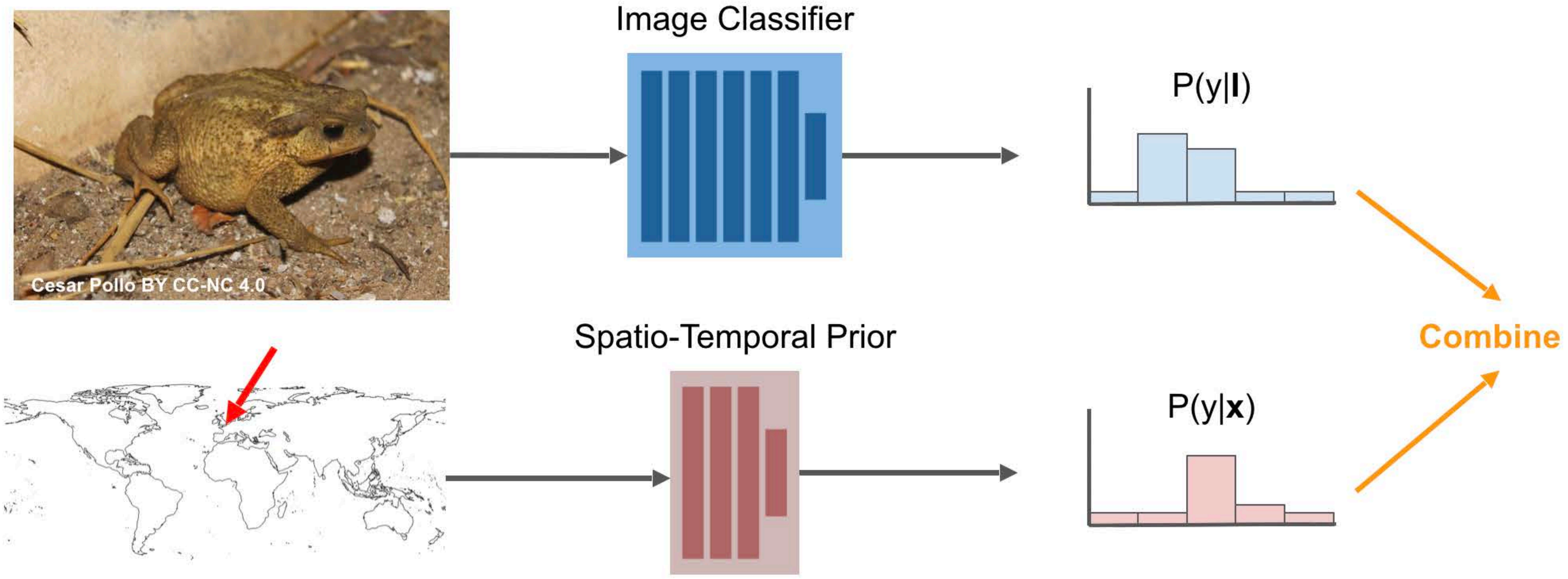
(Hashimoto et al. 2018, *Fairness without demographics in repeated loss minimization*)

What to do about distribution shift?

- Distributionally robust optimization

Learn a spatiotemporal prior

$$P(y|I, \mathbf{x}) \propto P(y|I)P(y|\mathbf{x})$$

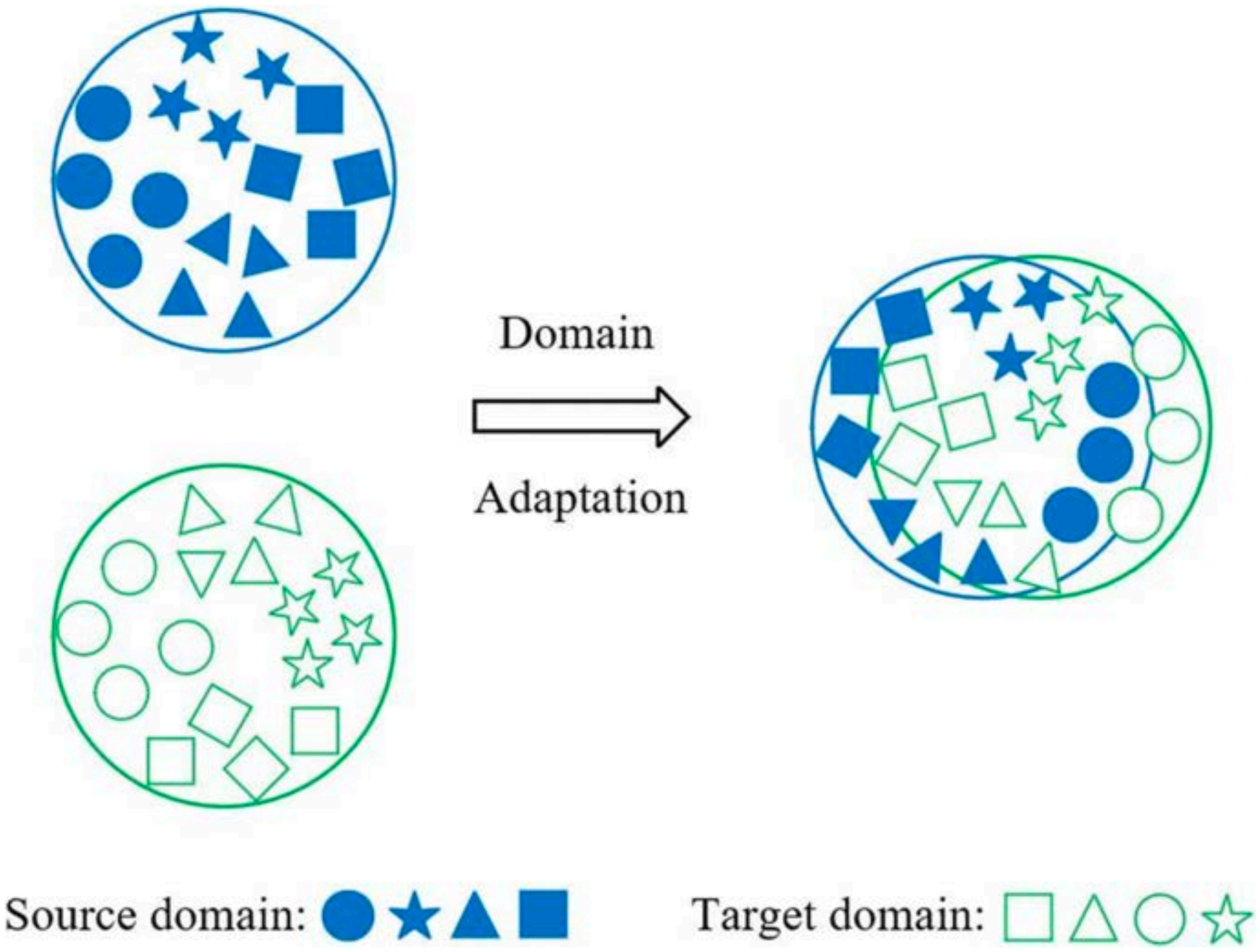


$\mathbf{x}$ = (longitude, latitude, day)

Presence-Only Geographical Priors for Fine-Grained Image Classification, Mac Aodha et al., 2019

What to do about distribution shift?

- Distributionally robust optimization
- Learn (or use) a prior for subpopulation shift



https://link.springer.com/article/10.1007/s10489-022-03709-8

What to do about distribution shift?

- Distributionally robust optimization
- Learn (or use) a prior for subpopulation shift
- Domain adaptation (next lecture!)

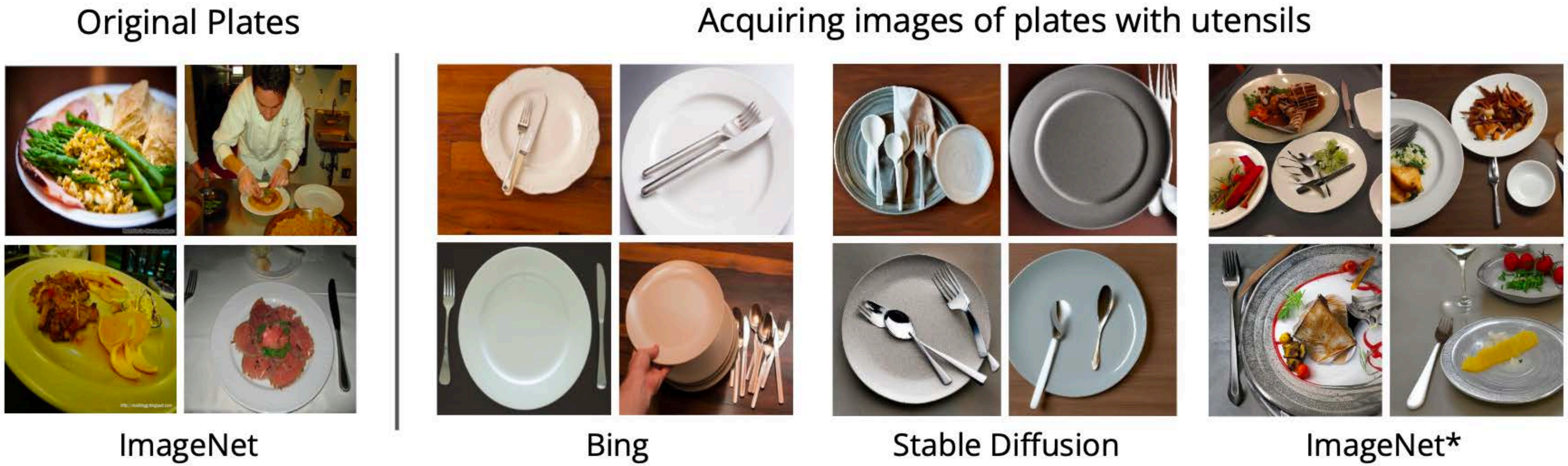


Figure 8: Real images of plates, with and without food and either on a table or in the grass. Below each image is the predicted class by an ImageNet-trained ResNet50.

https://arxiv.org/abs/2302.07865

What to do about distribution shift?

- Distributionally robust optimization
- Learn (or use) a prior for subpopulation shift
- Domain adaptation (next lecture!)
- Diagnose failures

What to do about distribution shift?

- Distributionally robust optimization
- Learn (or use) a prior for subpopulation shift
- Domain adaptation (next lecture!)
- Diagnose failures
- Get training data that is representative of your test domain (works better than any algorithm)

Summary

- Out-of-distribution generalization: big challenge, but helps understand what NNs learn.
    - Adversarial examples and training
    - Distribution shifts