

# **Lecture 27:** Ethics in computer vision (part 2)

# Announcements

- Last class :(
- Extra office hours:
  - Today: end of class until 2pm
  - Thurs: 12:00 - 1:00pm.
- PS8 grades out (regrade requests **due Friday**)
- PS9, PS10 due tonight. **No late days allowed.**



# Garbage in, garbage out

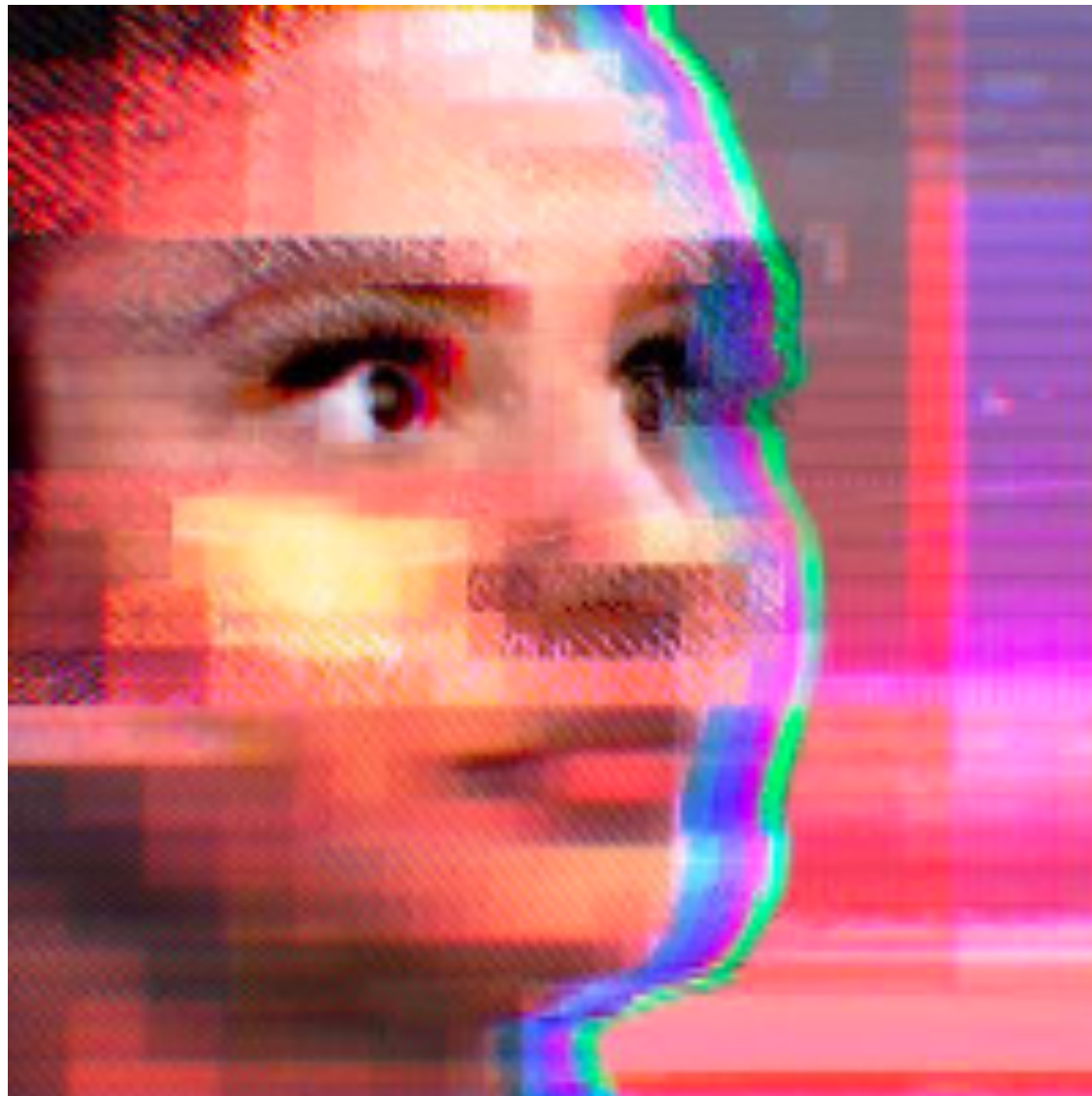
A machine learning algorithm will do whatever the training data tells it to do.

If the data is bad or biased, the learned algorithm will be too.

# Microsoft's Tay chatbot

Chatbot released on twitter.

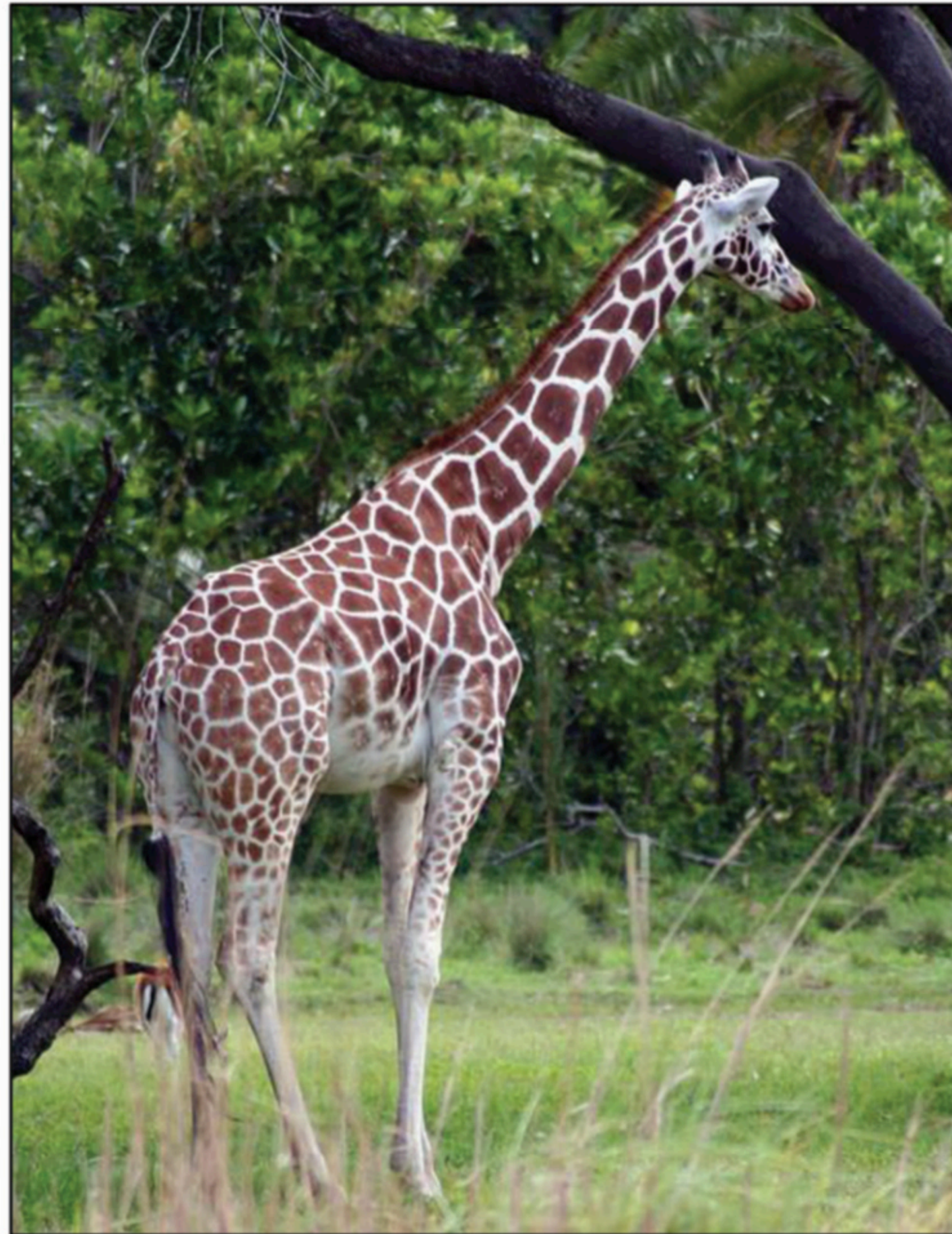
Learned from interactions with users



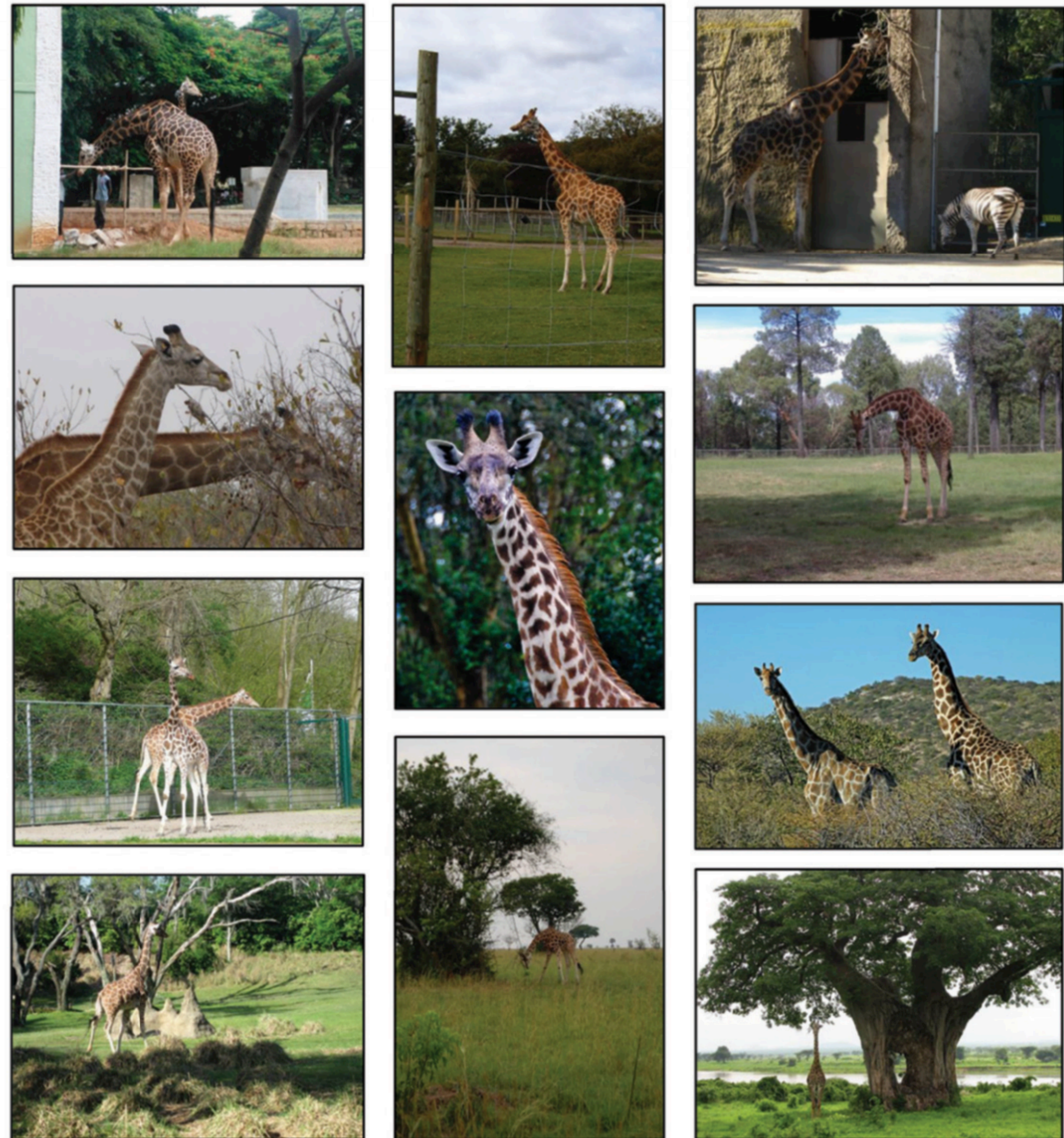
Started mimicking offensive language, was shut down.



# The Giraffe-Tree problem



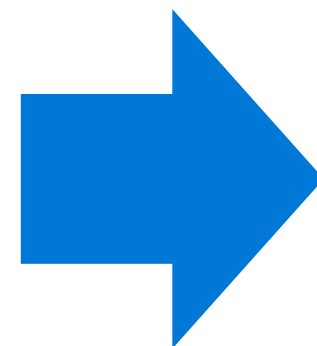
A giraffe standing in the grass next to a tree.



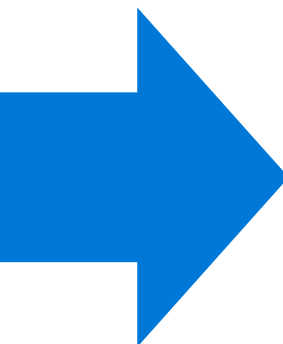


# Nearest neighbor baseline

Test



Train





# Nearest Neighbor



A black and white cat sitting in a bathroom sink.



Two zebras and a giraffe in a field.



# Image captioning



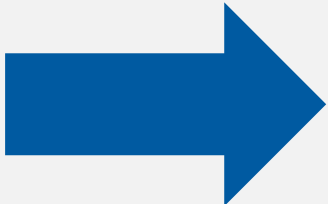
A man riding a motorcycle on a beach.

An airplane is parked on the tarmac at an airport.



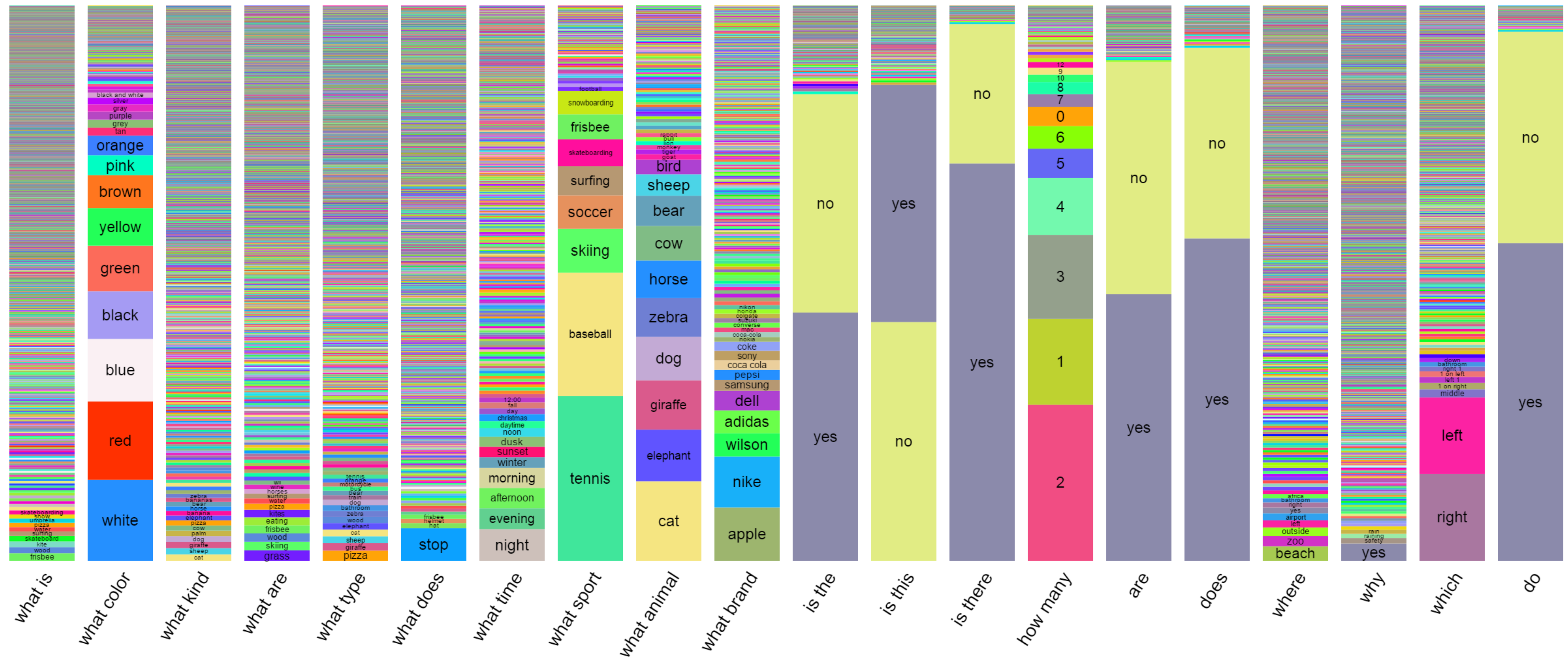
# Results

## COCO Caption Challenge



	CIDEr-D	Meteor	ROUGE-L	BLEU-4
Google <sup>[4]</sup>	0.943	0.254	0.53	0.309
MSR Captivator <sup>[9]</sup>	0.931	0.248	0.526	0.308
m-RNN <sup>[15]</sup>	0.917	0.242	0.521	0.299
MSR <sup>[8]</sup>	0.912	0.247	0.519	0.291
Nearest Neighbor <sup>[11]</sup>	0.886	0.237	0.507	0.280
m-RNN (Baidu/ UCLA) <sup>[16]</sup>	0.886	0.238	0.524	0.302
Berkeley LRCN <sup>[2]</sup>	0.869	0.242	0.517	0.277
Human <sup>[5]</sup>	0.854	0.252	0.484	0.217
Montreal/Toronto <sup>[10]</sup>	0.85	0.243	0.513	0.268
PicSOM <sup>[13]</sup>	0.833	0.231	0.505	0.281
MLBL <sup>[7]</sup>	0.74	0.219	0.499	0.26
ACVT <sup>[1]</sup>	0.709	0.213	0.483	0.246
NeuralTalk <sup>[12]</sup>	0.674	0.21	0.475	0.224
Tsinghua Bigeye <sup>[14]</sup>	0.673	0.207	0.49	0.241
MIL <sup>[6]</sup>	0.666	0.214	0.468	0.216
Brno University <sup>[3]</sup>	0.517	0.195	0.403 <sub>9</sub>	0.134

# Visual Question Answering Dataset



Source: L. Zitnick





Source: Isola, Torralba, Freeman

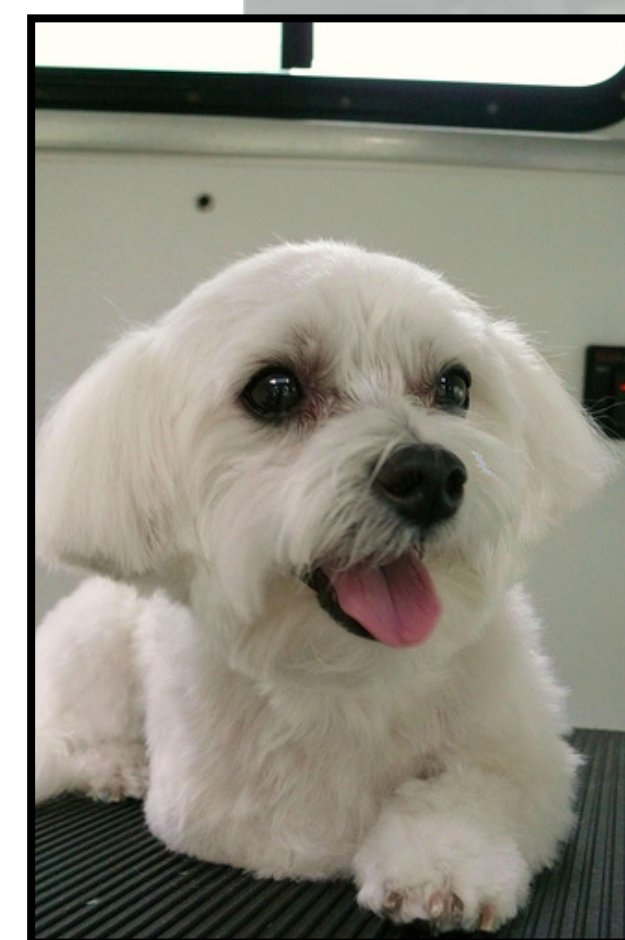
[“Colorful image colorization”, Zhang et al., ECCV 2016]





[“Colorful image colorization”, Zhang et al., ECCV 2016]





13

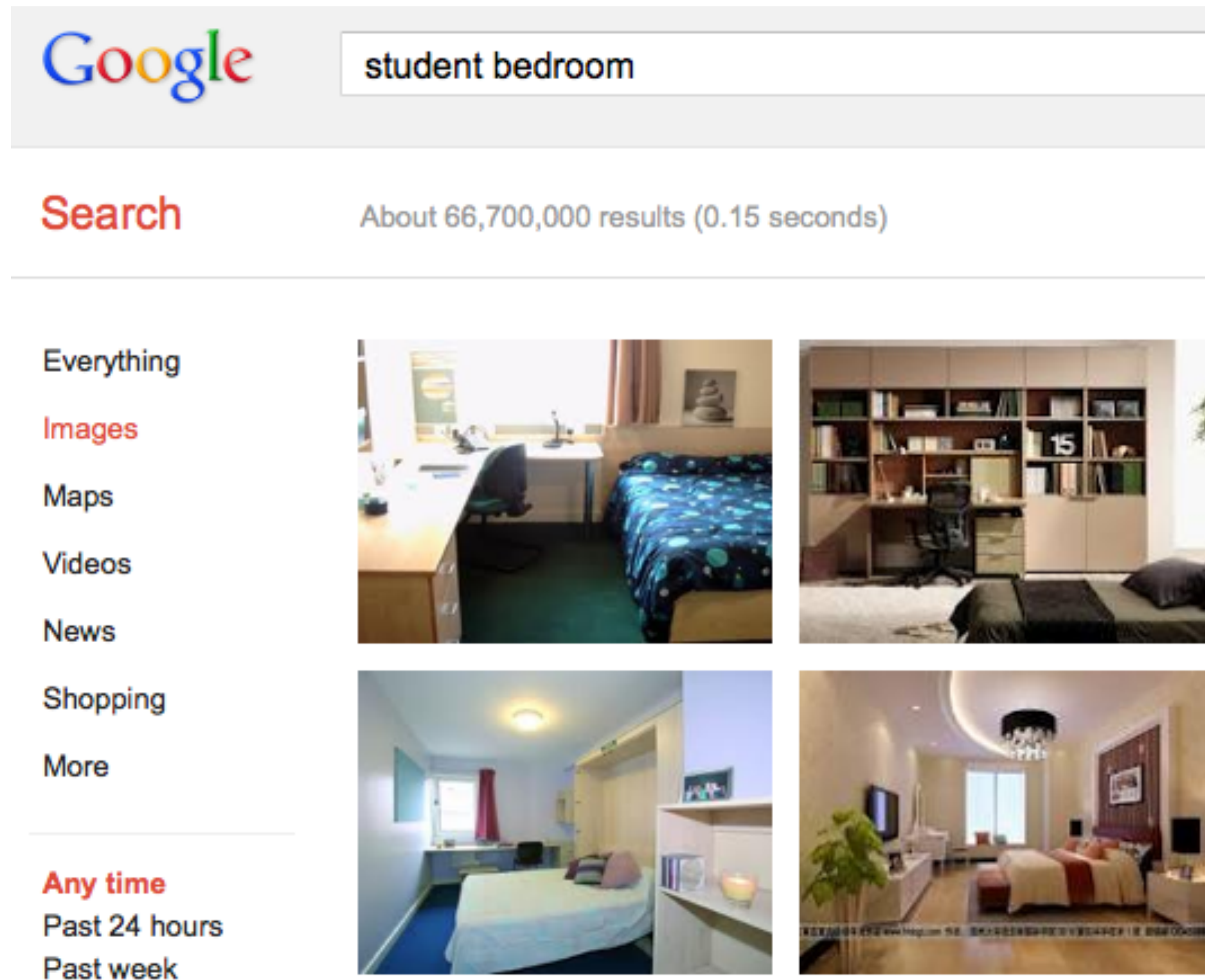


# Generalization



# Training data

What Google thinks are  
student bedrooms



# Test data





# Training data

Driving simulator (GTA)



# Test data

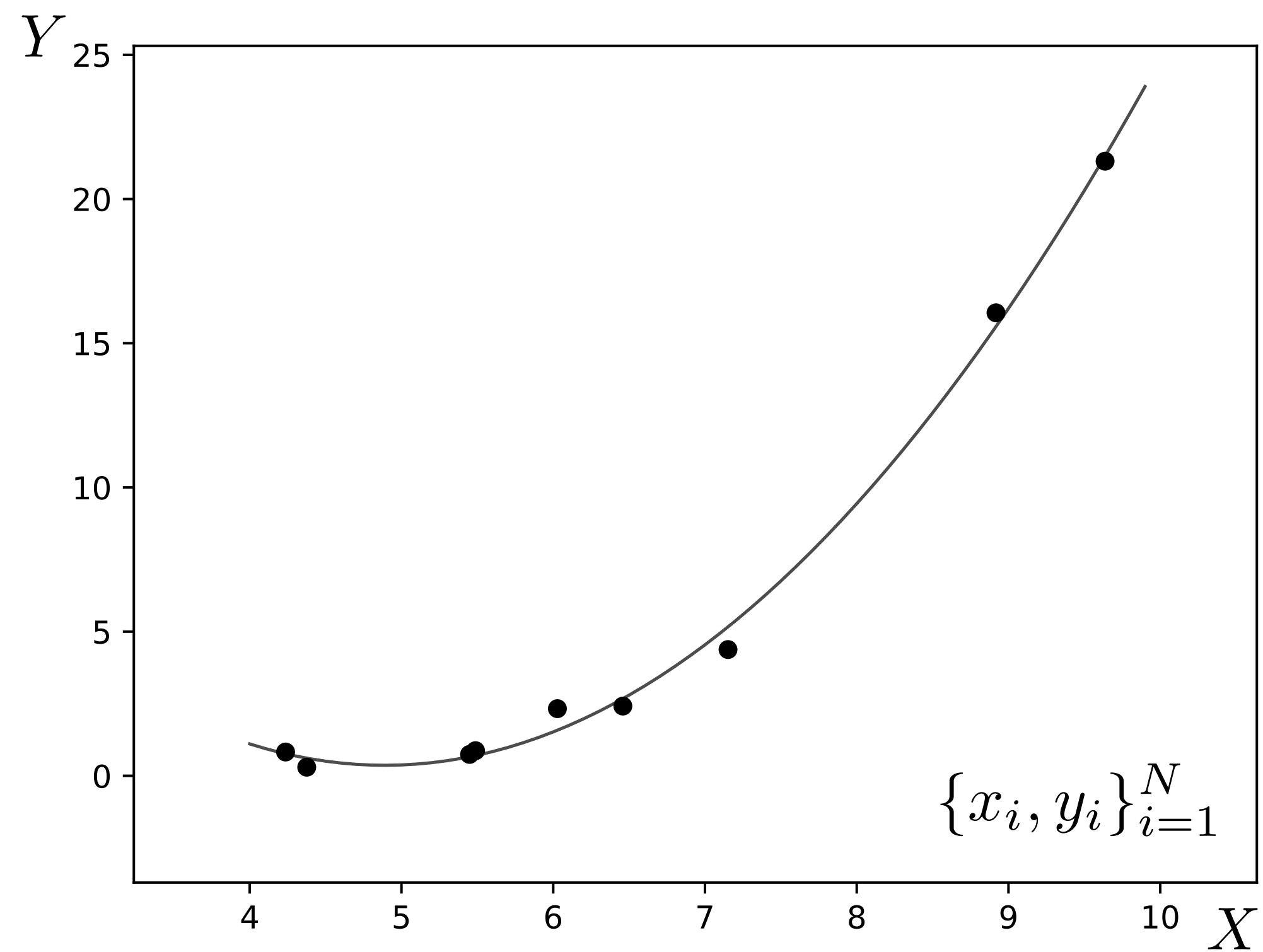
Driving in the real world



Need learning methods that can bridge this domain gap!

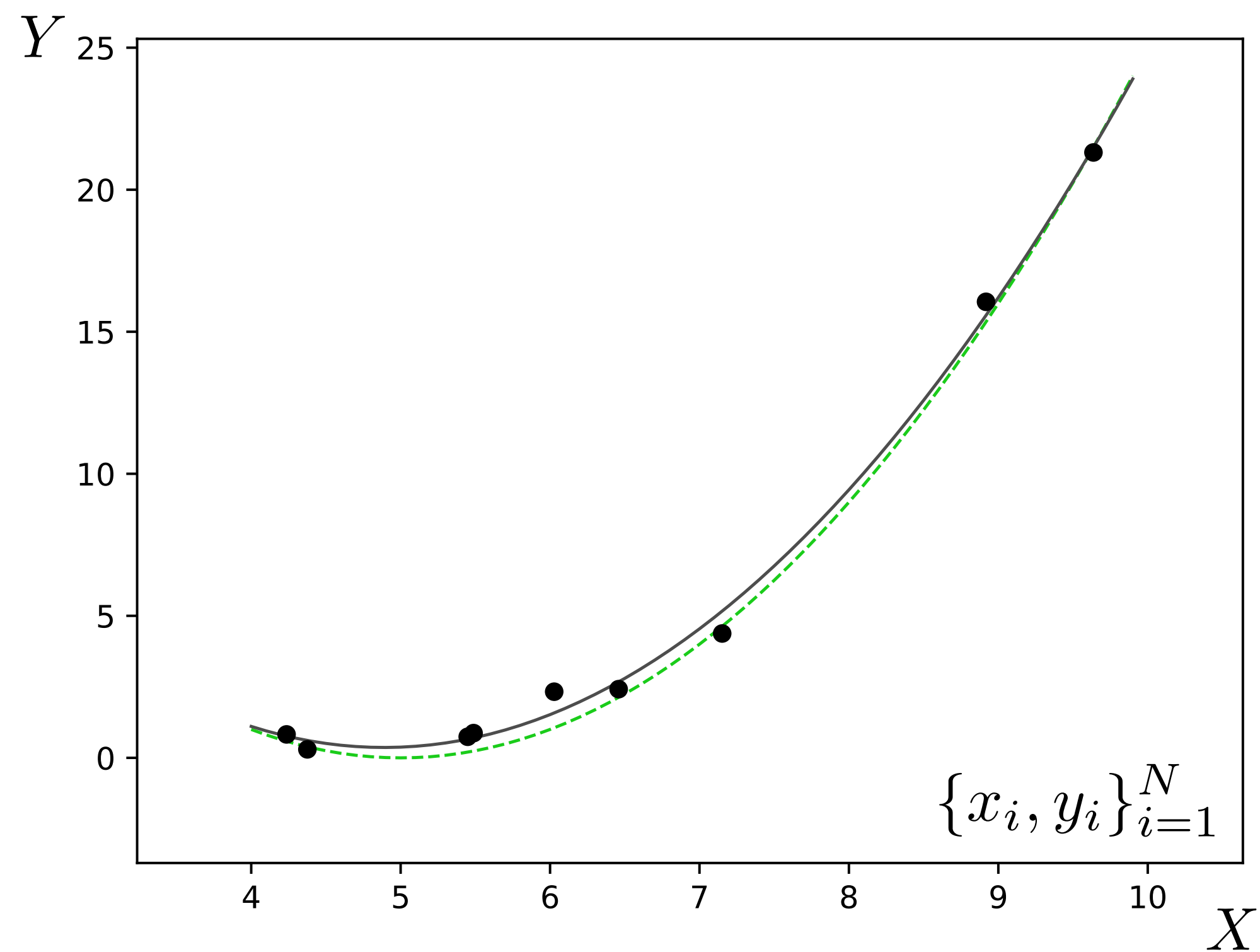
# Revisiting the problem of generalization

# Training data





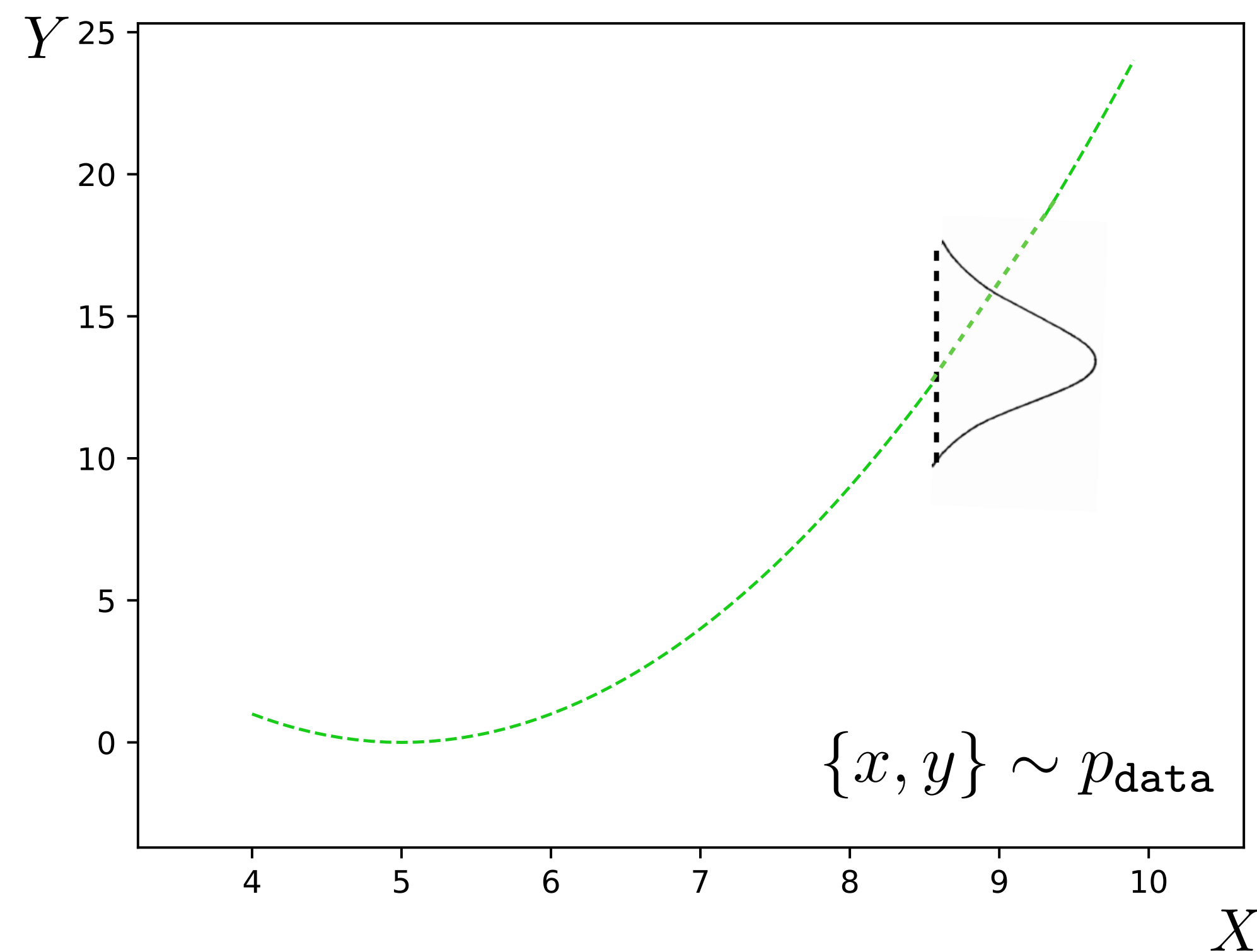
# Training data



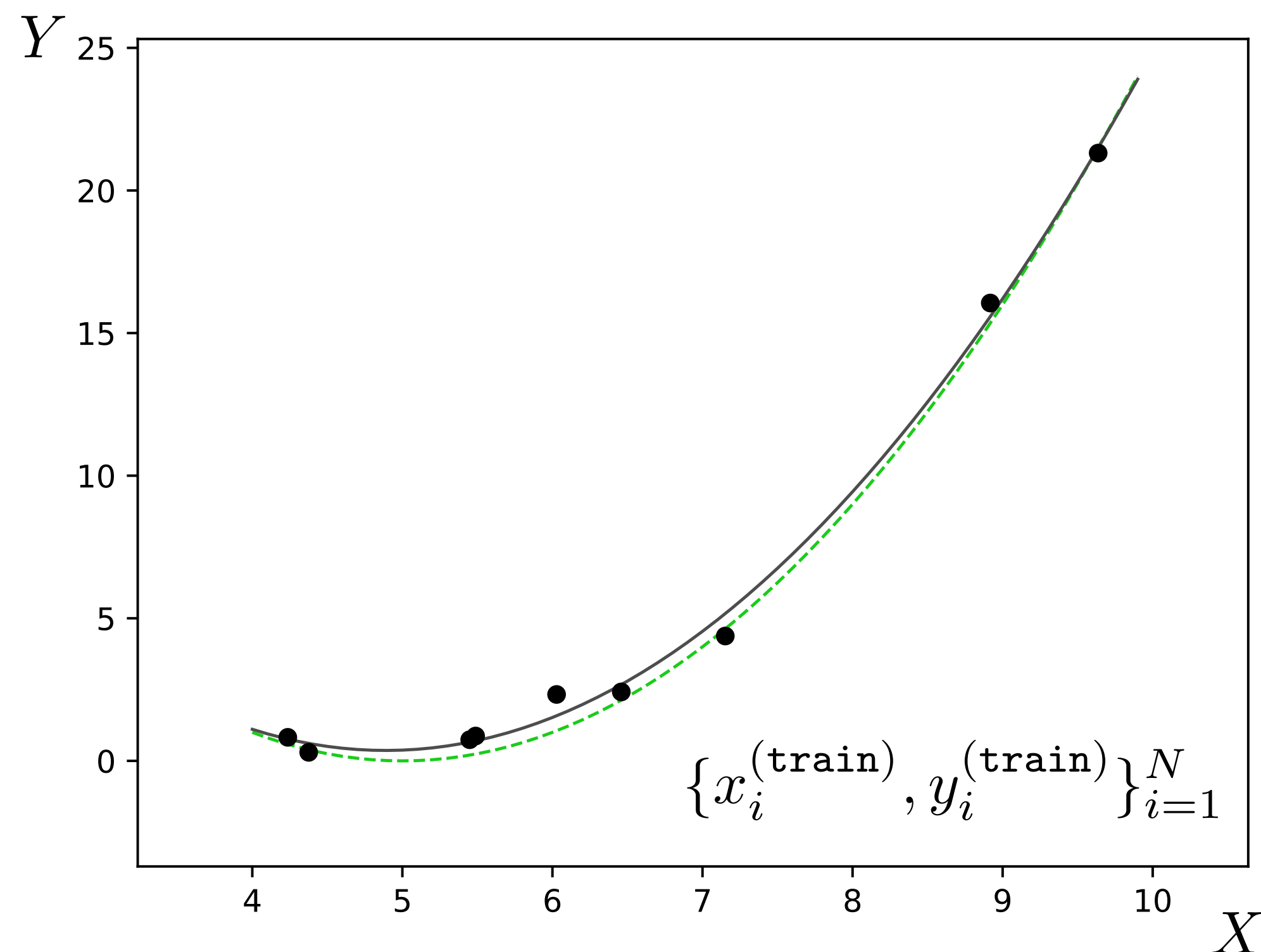
True data-generating process

$p_{\text{data}}$

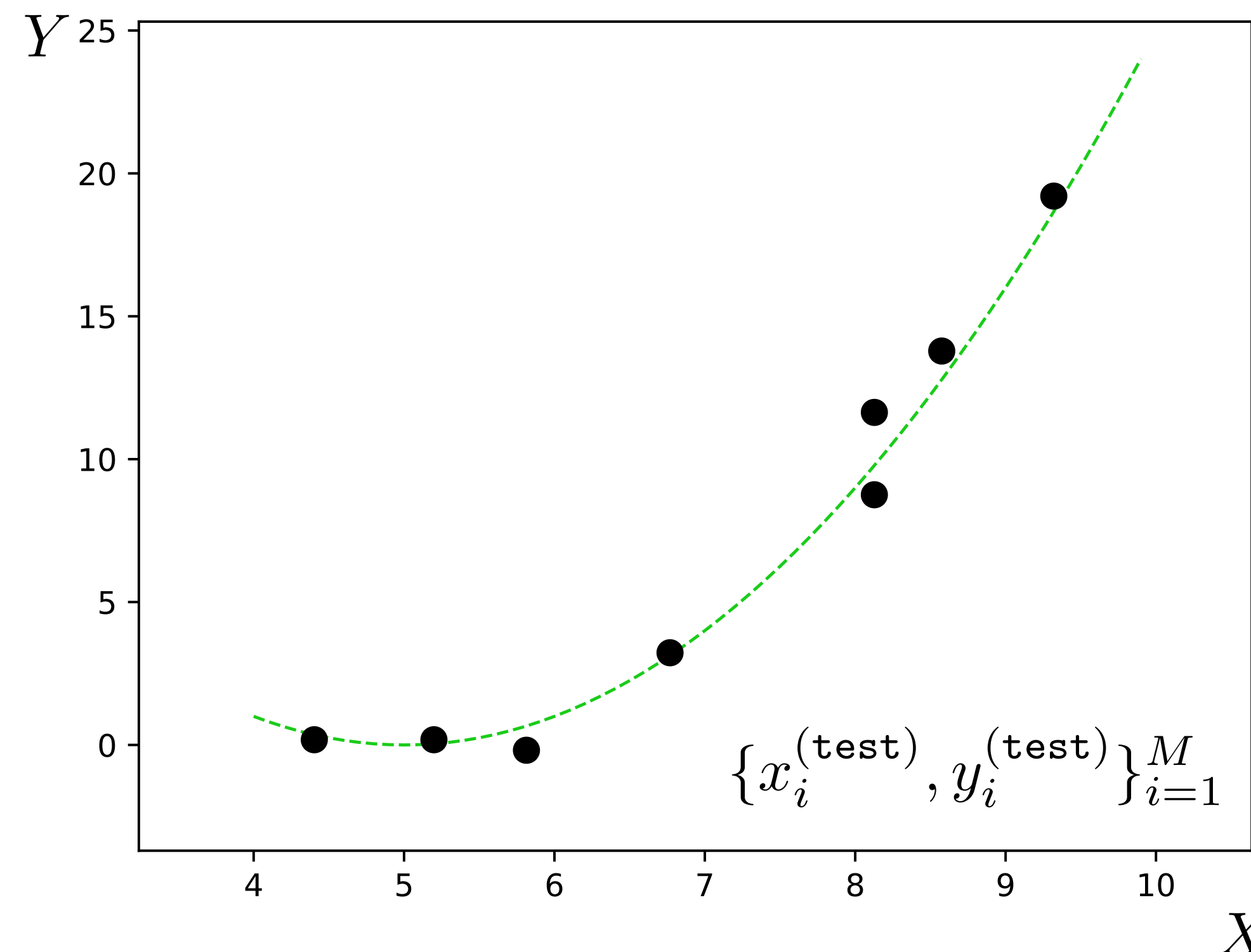
# Test data



# Training data



# Test data



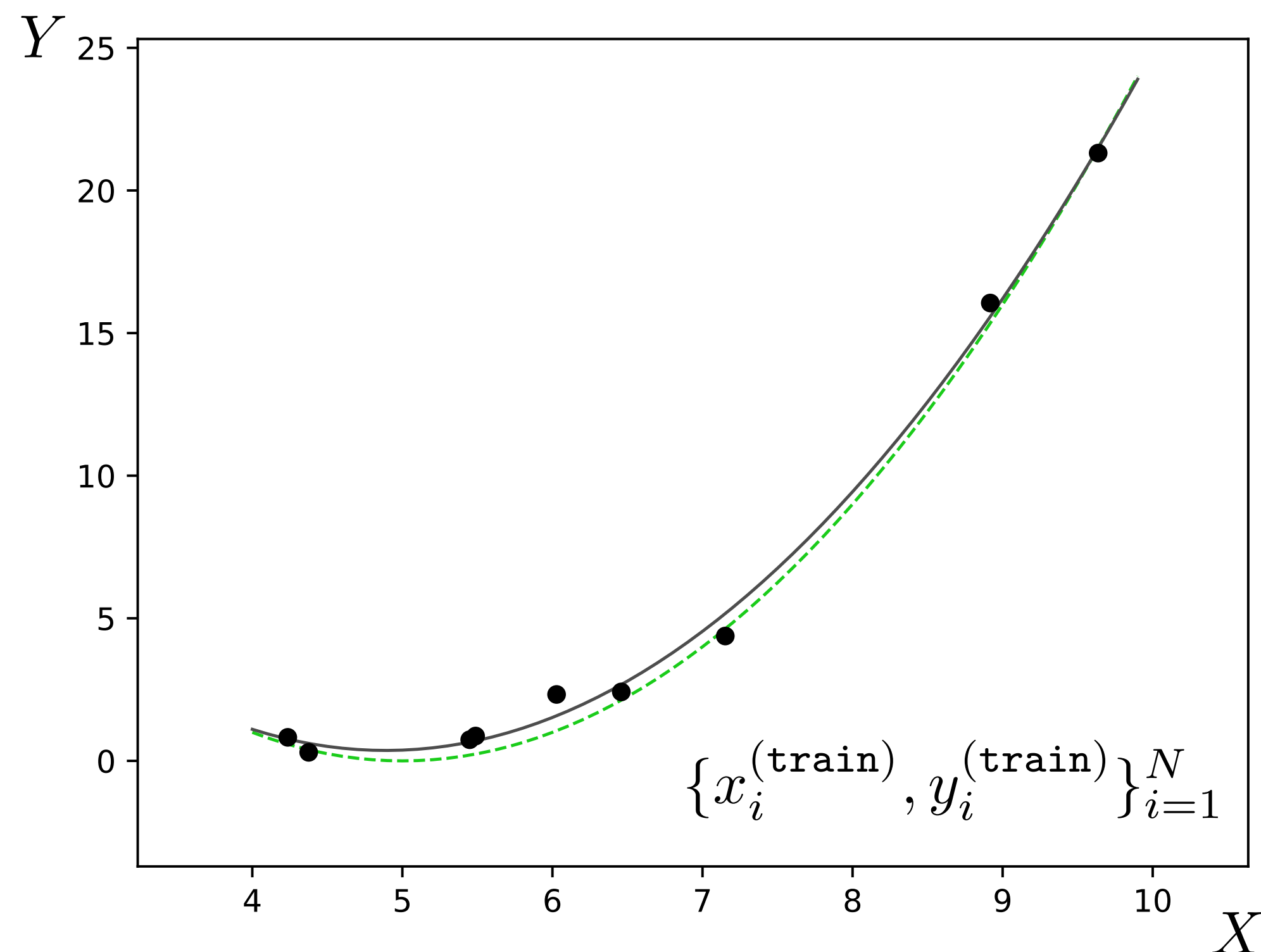
True data-generating process

$p_{\text{data}}$

$$\{x_i^{(\text{train})}, y_i^{(\text{train})}\} \stackrel{\text{iid}}{\sim} p_{\text{data}}$$

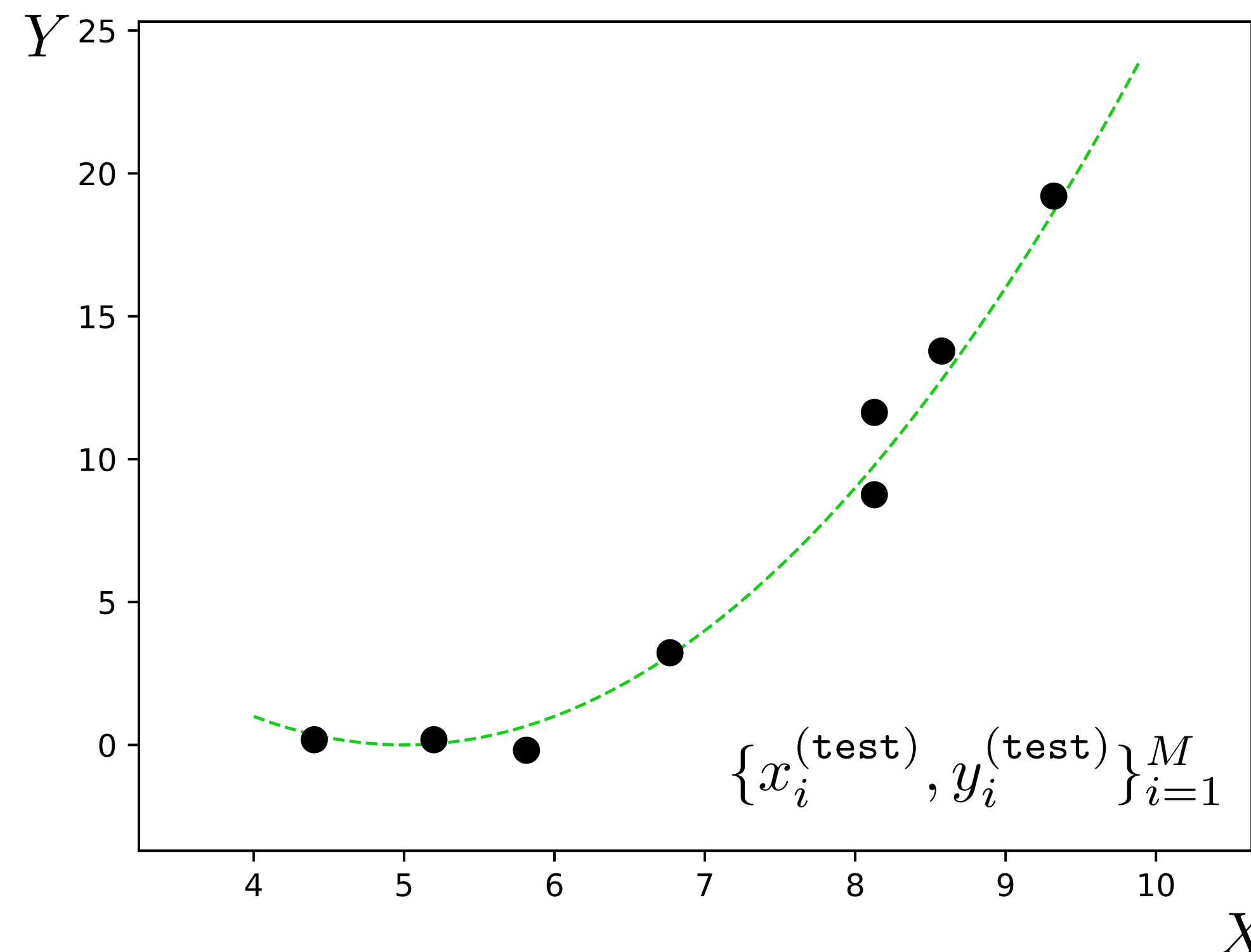
$$\{x_i^{(\text{test})}, y_i^{(\text{test})}\} \stackrel{\text{iid}}{\sim} p_{\text{data}}$$

# Training data



This is a huge assumption!  
Almost never true in practice!

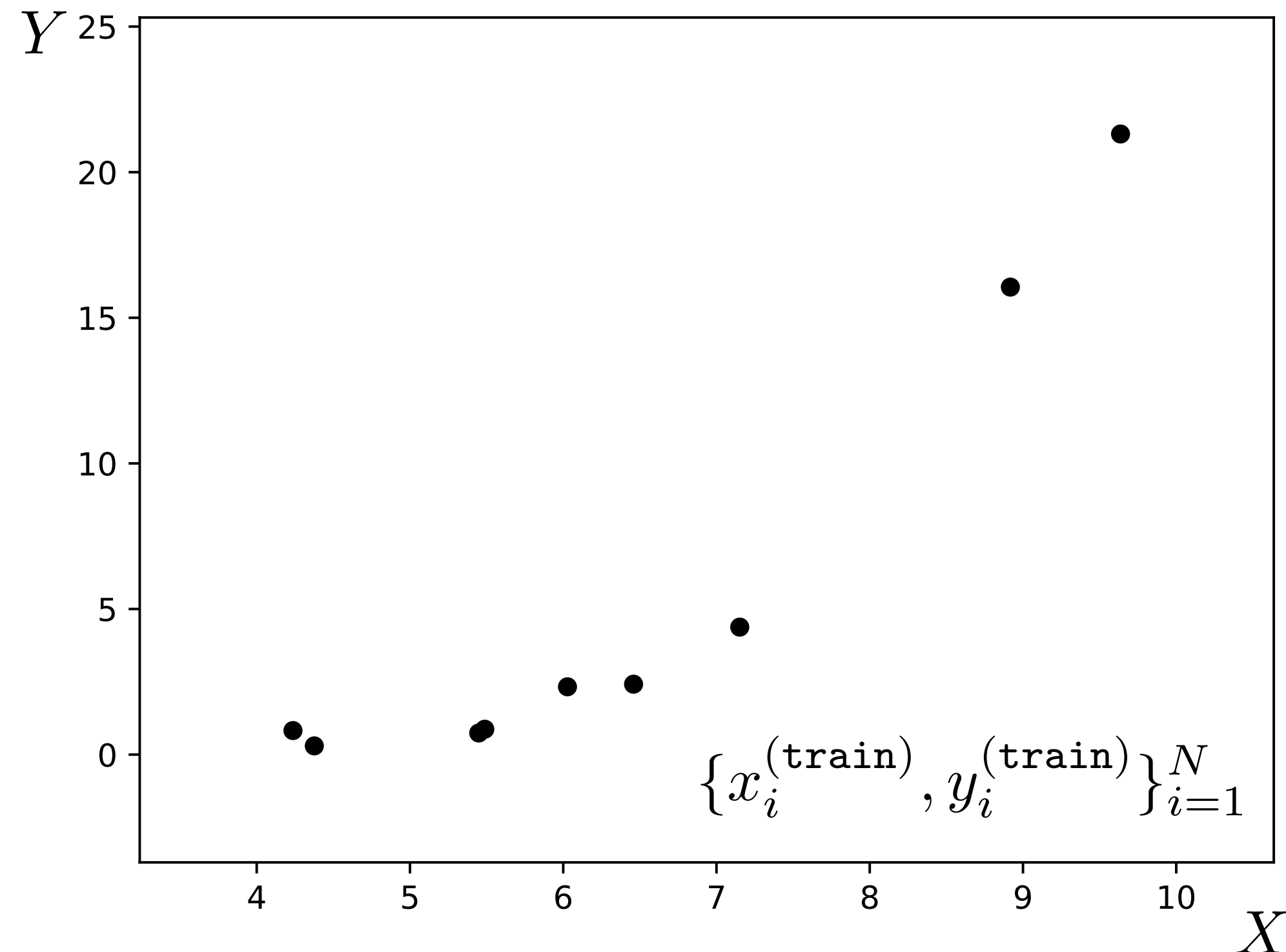
# Test data



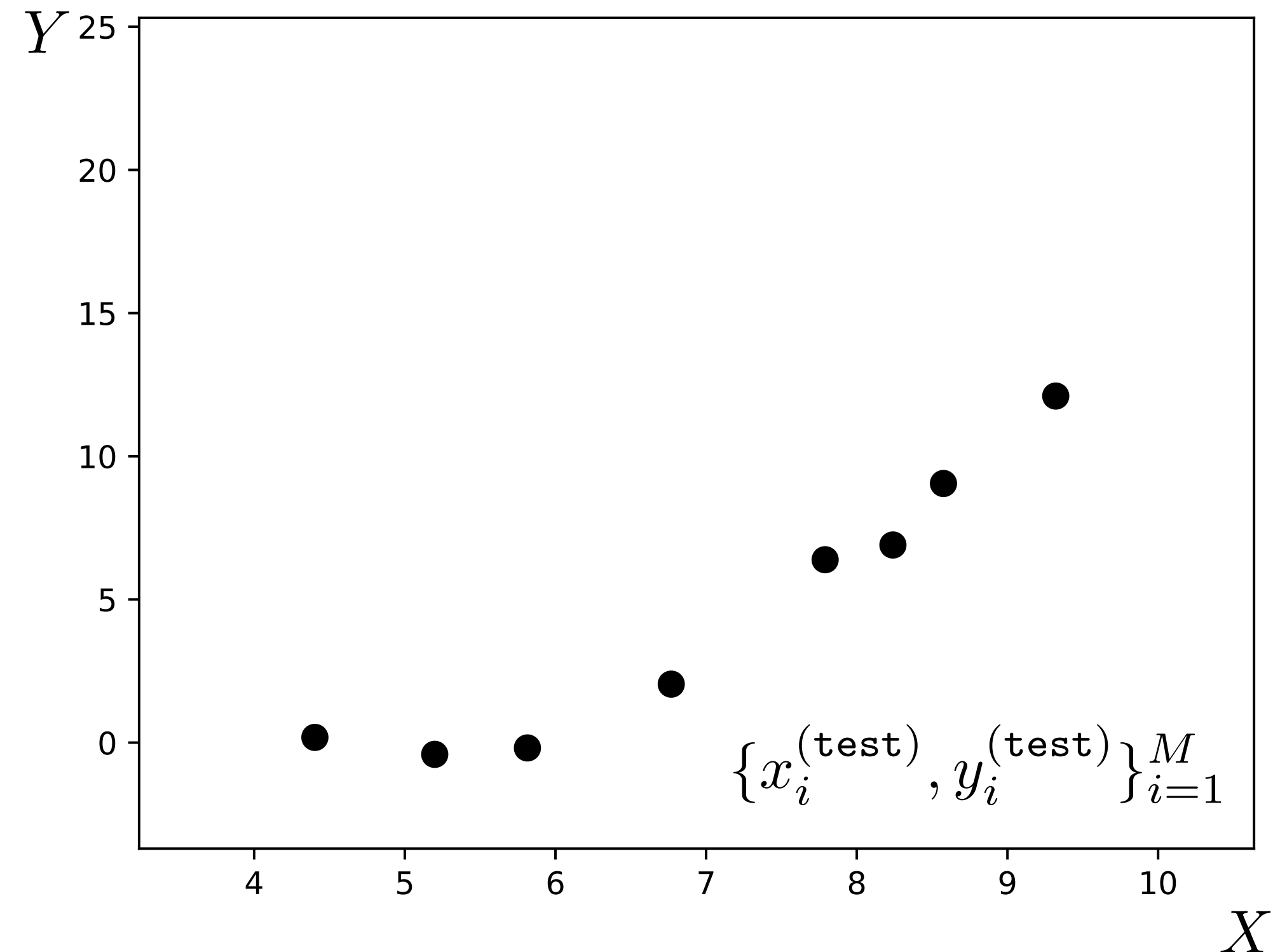
$$\{x_i^{(\text{train})}, y_i^{(\text{train})}\} \stackrel{\text{iid}}{\sim} p_{\text{data}}$$

$$\{x_i^{(\text{test})}, y_i^{(\text{test})}\} \stackrel{\text{iid}}{\sim} p_{\text{data}}$$

# Training data



# Test data



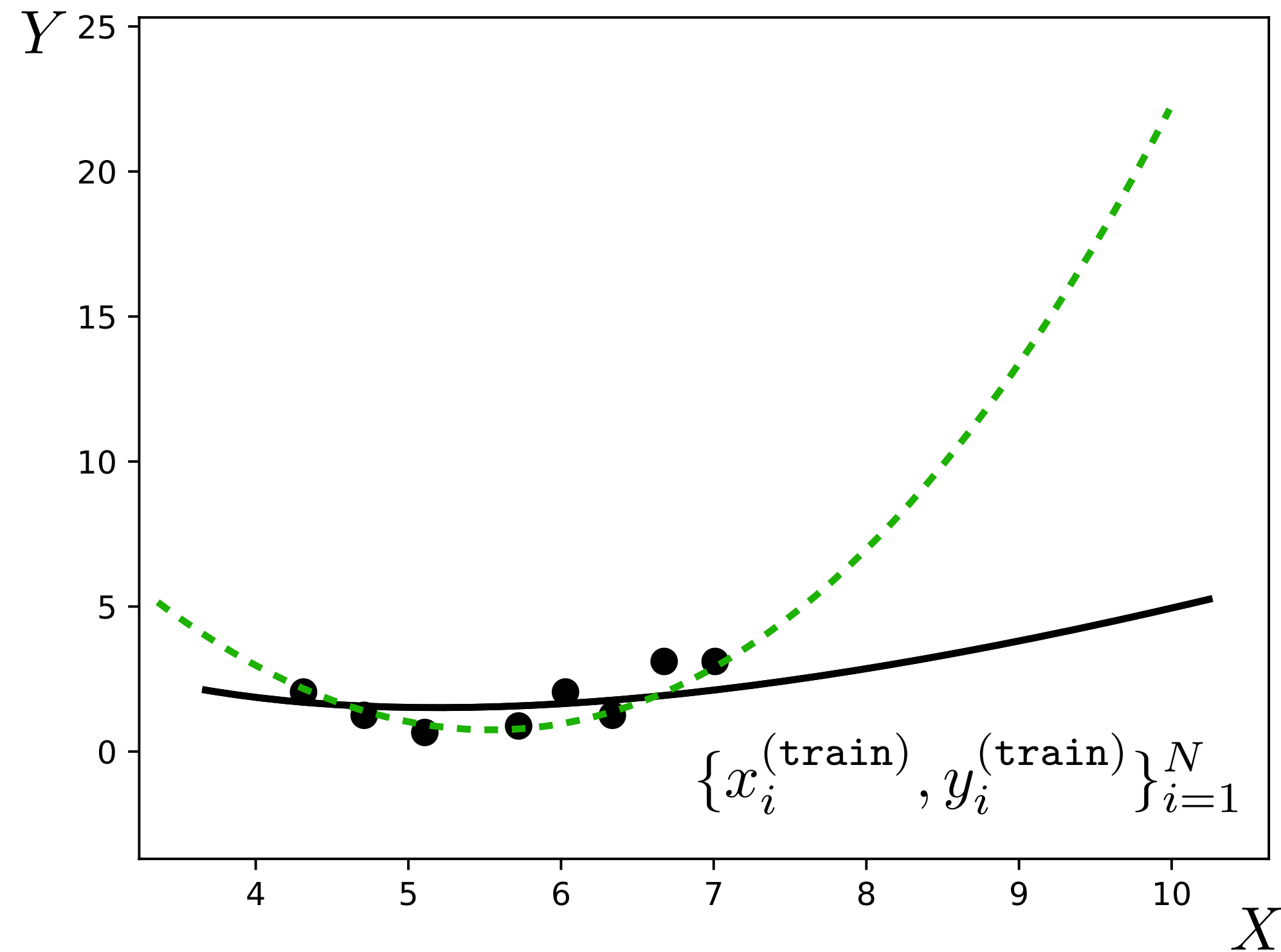
Much more commonly, we have

$$p_{\text{train}} \neq p_{\text{test}}$$

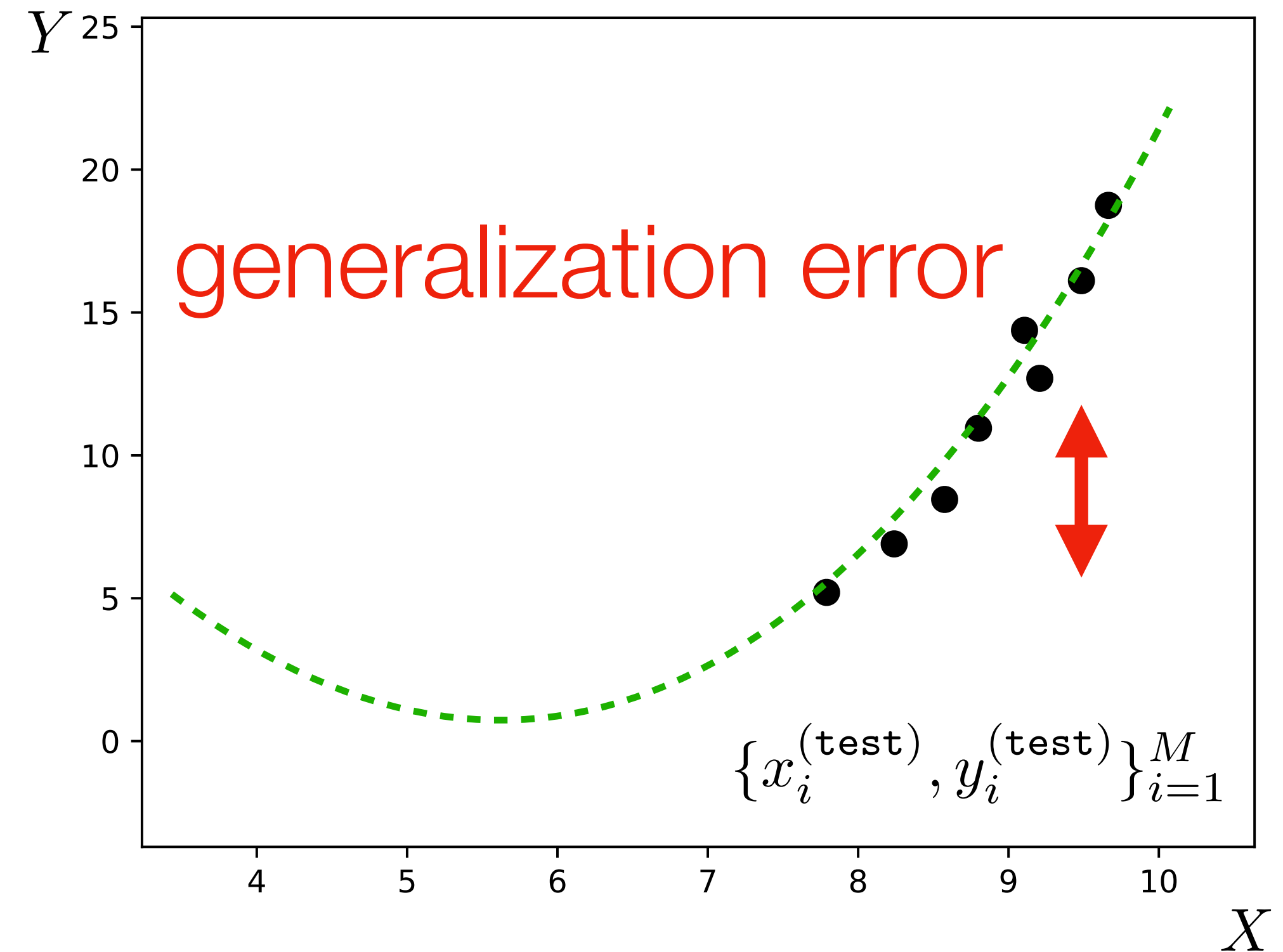
$$\{x_i^{(\text{train})}, y_i^{(\text{train})}\} \stackrel{\text{iid}}{\sim} p_{\text{train}}$$

$$\{x_i^{(\text{test})}, y_i^{(\text{test})}\} \stackrel{\text{iid}}{\sim} p_{\text{test}}$$

Training data



Test data



Our training data did cover the part of the distribution that was tested  
**(biased data)**

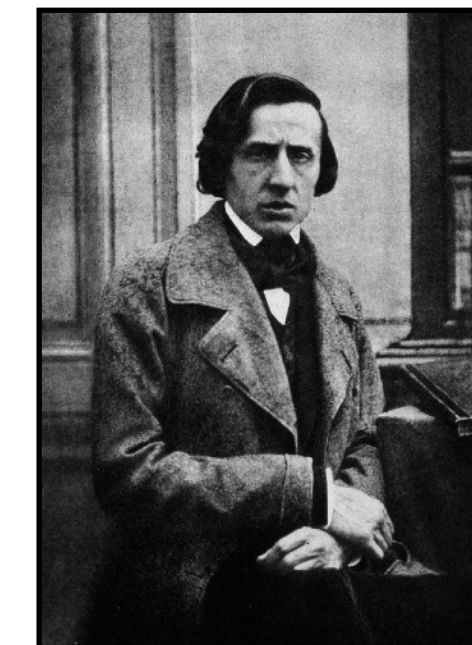
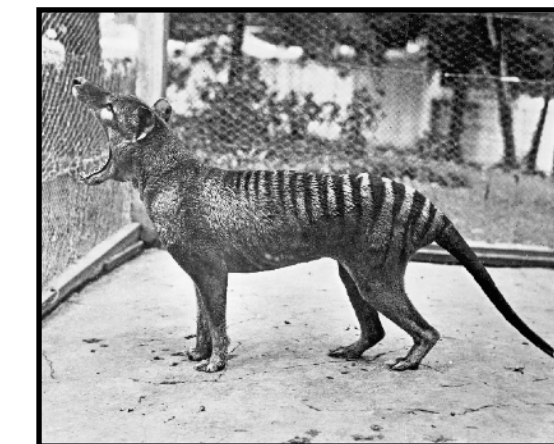
training domain

testing domain  
(where we actual use our model)

**Domain gap** between  $p_{\text{train}}$  and  $p_{\text{test}}$  will cause us to fail to generalize.

Space of natural images

Training data



Test data

24



# Social consequences

## Color Matters in Computer Vision

Facial recognition algorithms made by Microsoft, IBM and Face++ were more likely to misidentify the gender of black women than white men.






Gender was misidentified in **up to 1 percent of lighter-skinned males** in a set of 385 photos.



Gender was misidentified in **35 percent of darker-skinned females** in a set of 271 photos.



# Algorithmic Bias

Gender Classifier	Darker Male	Darker Female	Lighter Male	Lighter Female	Largest Gap
 Microsoft	94.0% <div><div></div></div>	79.2% <div><div></div></div>	100% <div><div></div></div>	98.3% <div><div></div></div>	20.8% <div><div></div></div>
 FACE++	99.3% <div><div></div></div>	65.5% <div><div></div></div>	99.2% <div><div></div></div>	94.0% <div><div></div></div>	33.8% <div><div></div></div>
 IBM	88.0% <div><div></div></div>	65.3% <div><div></div></div>	99.7% <div><div></div></div>	92.9% <div><div></div></div>	34.4% <div><div></div></div>



<http://gendershades.org/overview.html>

## Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification\*

Joy Buolamwini

MIT Media Lab 75 Amherst St. Cambridge, MA 02139

JOYAB@MIT.EDU

Timnit Gebru

Microsoft Research 641 Avenue of the Americas, New York, NY 10011

TIMNIT.GEBRU@MICROSOFT.COM

Editors: Sorelle A. Friedler and Christo Wilson

### Abstract

Recent studies demonstrate that machine learning algorithms can discriminate based on classes like race and gender. In this work, we present an approach to evaluate bias present in automated facial analysis algorithms and datasets with respect to phenotypic subgroups. Using the dermatologist approved Fitzpatrick Skin Type classification system, we characterize the gender and skin type distribution of two facial analysis benchmarks, IJB-A and Adience. We find that these datasets are overwhelmingly composed of lighter-skinned subjects (79.6% for IJB-A and 86.2% for Adience) and introduce a new facial analysis dataset which is balanced by gender and skin type. We evaluate 3 commercial gender classification systems using our dataset and show that darker-skinned females are the most misclassified group (with error rates of up to 34.7%). The maximum error rate for lighter-skinned males is 0.8%. The substantial disparities in the accuracy of classifying darker females, lighter females, darker males, and lighter males in gender classification systems require urgent attention if commercial companies are to build genuinely fair, transparent and accountable facial analysis algorithms.

**Keywords:** Computer Vision, Algorithmic Audit, Gender Classification

### 1. Introduction

Artificial Intelligence (AI) is rapidly infiltrating every aspect of society. From helping determine

\* Download our gender and skin type balanced PPB dataset at [gendershades.org](http://gendershades.org)

who is hired, fired, granted a loan, or how long an individual spends in prison, decisions that have traditionally been performed by humans are rapidly made by algorithms (O’Neil, 2017; Citron and Pasquale, 2014). Even AI-based technologies that are not specifically trained to perform high-stakes tasks (such as determining how long someone spends in prison) can be used in a pipeline that performs such tasks. For example, while face recognition software by itself should not be trained to determine the fate of an individual in the criminal justice system, it is very likely that such software is used to identify suspects. Thus, an error in the output of a face recognition algorithm used as input for other tasks can have serious consequences. For example, someone could be wrongfully accused of a crime based on erroneous but confident misidentification of the perpetrator from security video footage analysis.

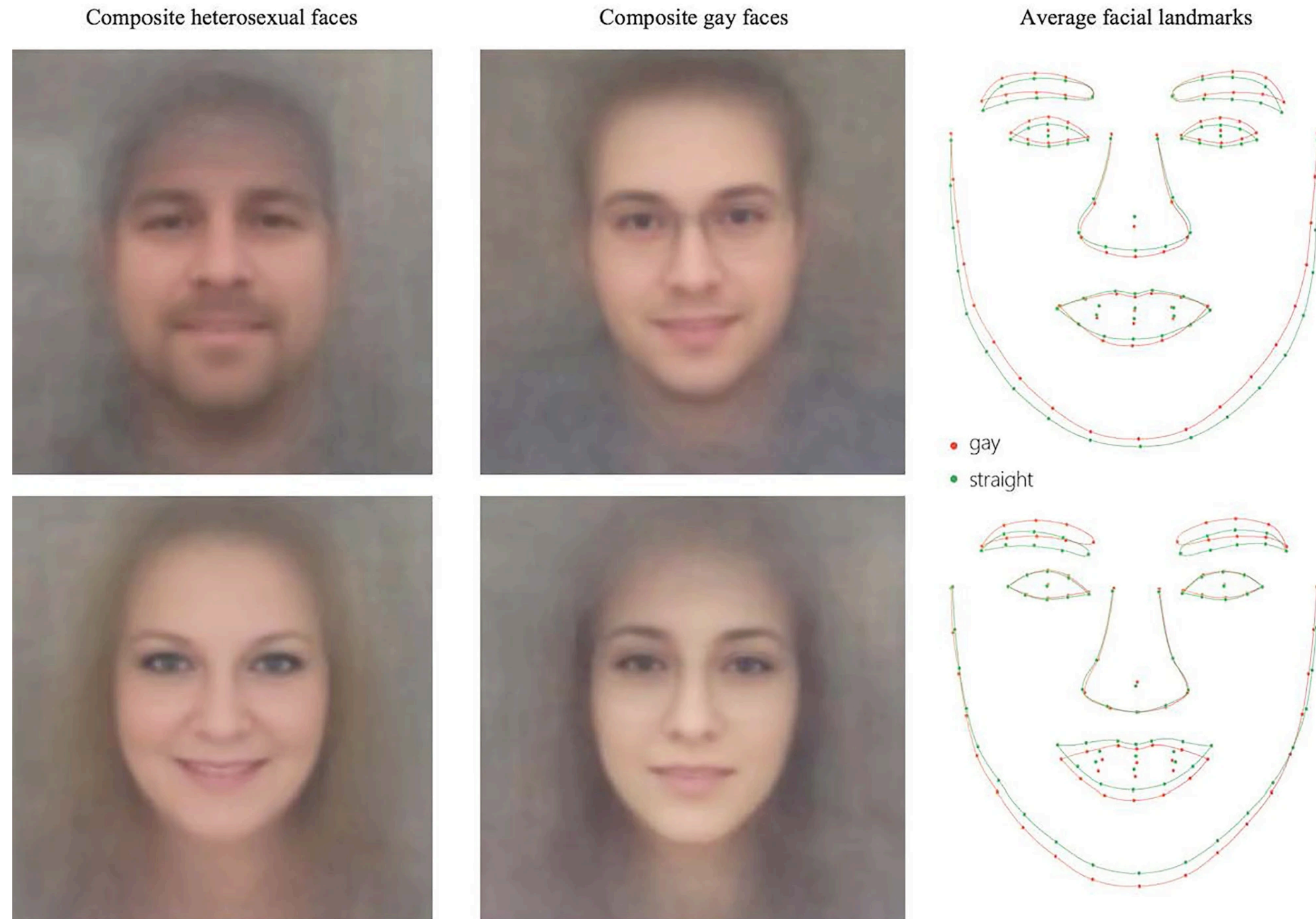
Many AI systems, e.g. face recognition tools, rely on machine learning algorithms that are trained with labeled data. It has recently been shown that algorithms trained with biased data have resulted in algorithmic discrimination (Bolukbasi et al., 2016; Caliskan et al., 2017). Bolukbasi et al. even showed that the popular word embedding space, Word2Vec, encodes societal gender biases. The authors used Word2Vec to train an analogy generator that fills in missing words in analogies. The analogy man is to computer programmer as woman is to “X” was completed with “homemaker”, conforming to the stereotype that programming is associated with men and homemaking with women. The biases in Word2Vec are thus likely to be propagated throughout any system that uses this embedding.

<http://proceedings.mlr.press/v81/buolamwini18a/buolamwini18a.pdf>

Source: Isola, Torralba, Freeman

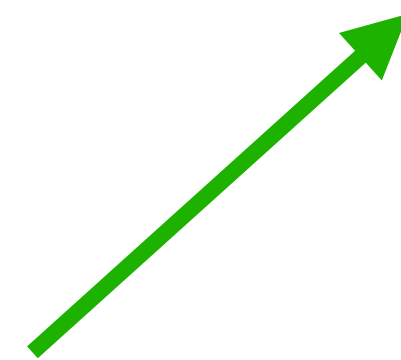


# Questionable data choices



<https://www.nytimes.com/2017/10/09/science/stanford-sexual-orientation-study.html>

# How can we collect good data?



- + Correctly labeled
- + Unbiased (good coverage of all relevant kinds of data)

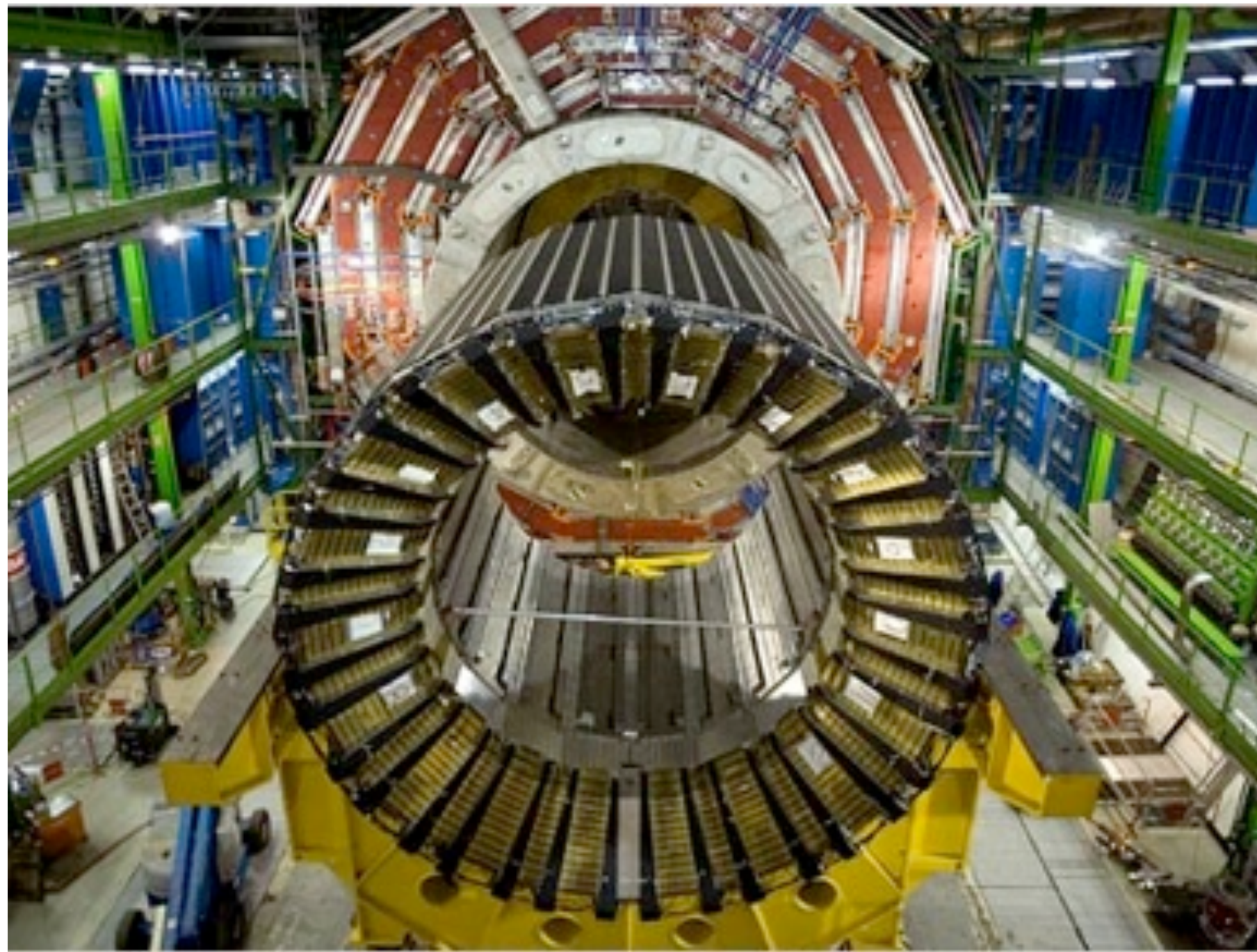


# Crowdsourcing





# The value of data



The Large Hadron Collider

\$  $10^{10}$

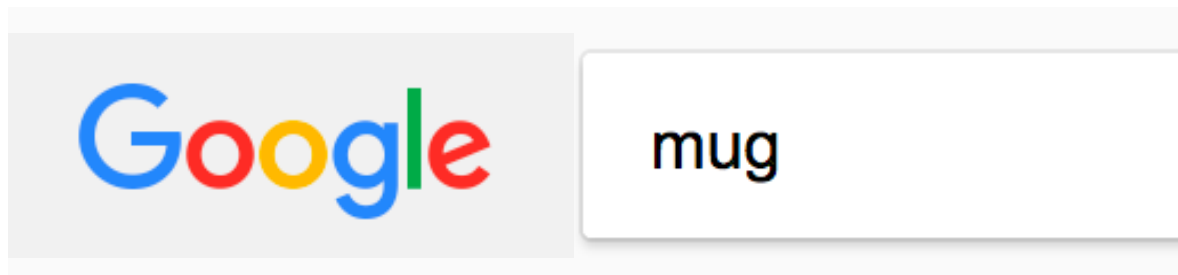


Amazon Mechanical Turk

30 \$  $10^2 - 10^4$

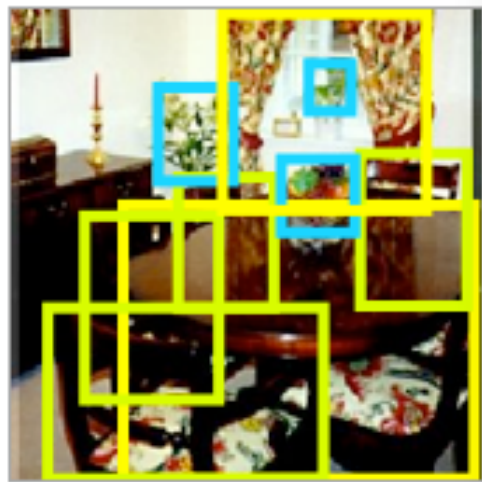


But can humans collect good data?



# Getting more humans in the annotation loop

Labeling to get a Ph.D.



Labeling for money  
(Sorokin, Forsyth, 2008)

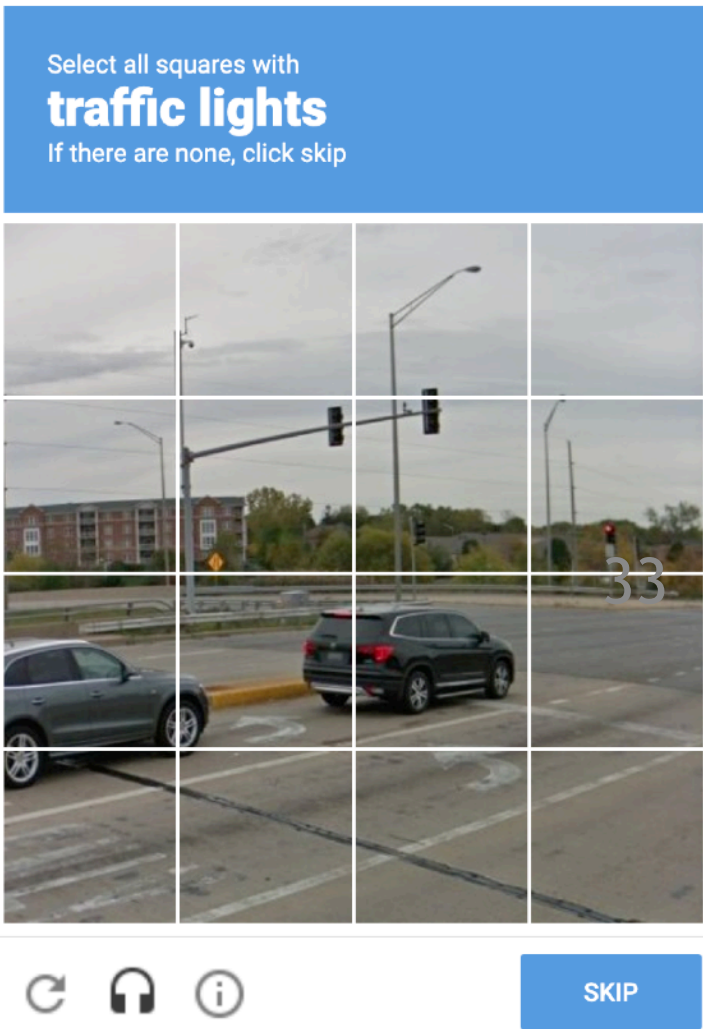


Labeling for fun

Luis Von Ahn and Laura Dabbish 2004



Labeling to prove  
you're human



Source: Isola, Torralba, Freeman

Labeling because it  
gives you added value



Visipedia  
(Belongie, Perona, et al)



# Beware of the human in your loop

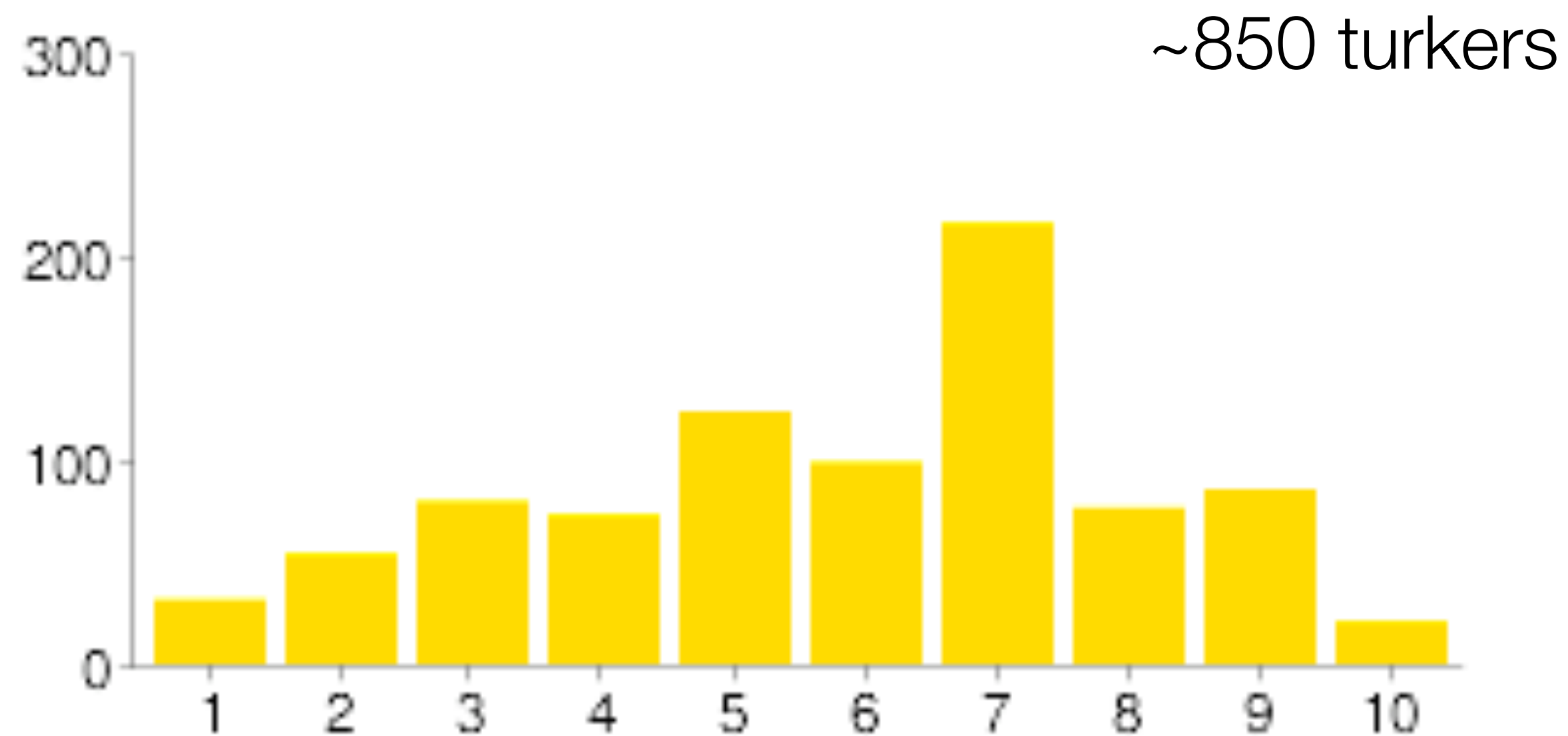
- What do you know about them?
- Will they do the work you pay for?

Let's check a few simple experiments



# People have biases...

Turkers were offered 1 cent to pick a number from 1 to 10.



35

Experiment by Greg Little

From <http://groups.csail.mit.edu/uid/deneme/>

# Do humans have consistent biases?

Choose Item

Requester:

 SimpleSphere

Reward:

 \$0.01 per HIT

HITs Available:

 1

Duration:

 60 minutes

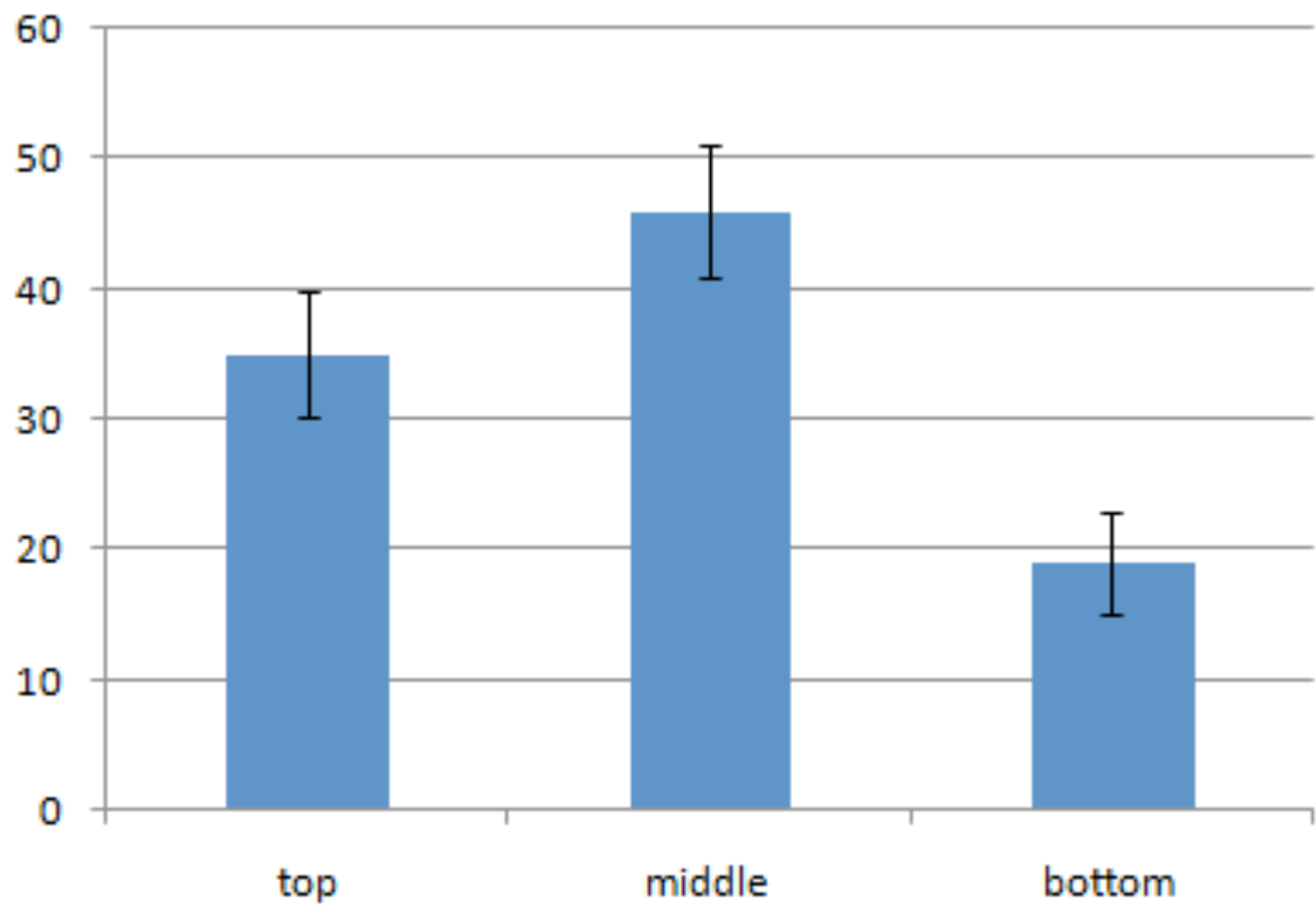
Qualifications Required:

 None

Please choose one of the following:



Results form 100 HITS:



Experiment by Greg Little  
From <http://groups.csail.mit.edu/uid/deneme/>



# Are humans reliable even in simple tasks?

Choose the given item.

**Requester:** SimpleSphere

**Reward:** \$0.01 per HIT

**HITs Available:** 1

**Duration:** 60 minutes

**Qualifications Required:** None

Please click button B:

B

C

A

Results of 100 HITS:

A: 2

B: 96

C: 2

37

Experiment by Greg Little

From <http://groups.csail.mit.edu/uid/deneme/>

# Do humans do what you ask for?

Flip a coin

Requester: ROBERT C MILLER

Reward: \$0.01 per HIT

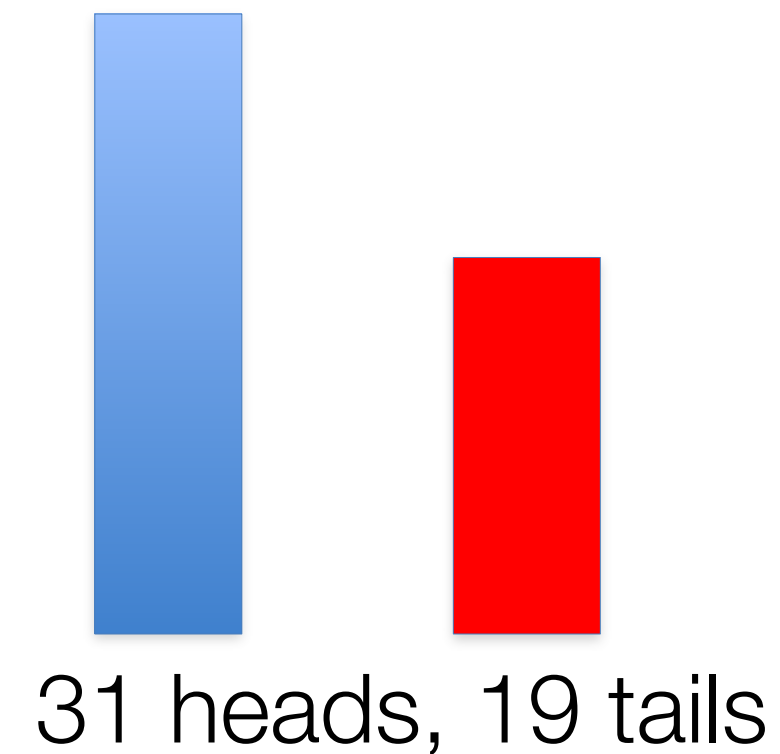
HITs Available: 3

Duration: 5 minutes

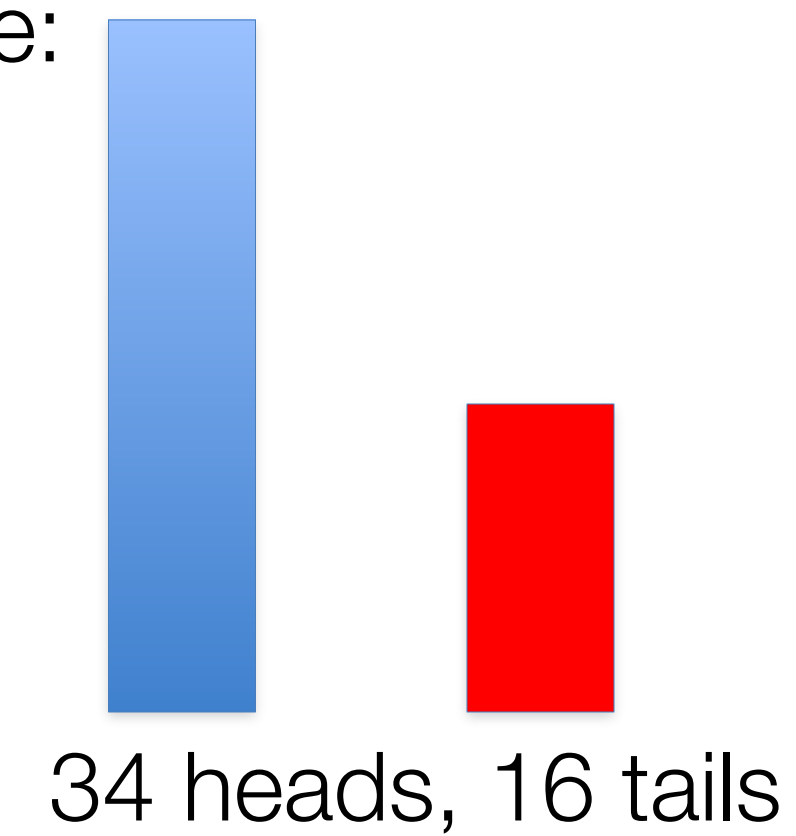
Qualifications Required: None

Please flip an actual coin and type either H or T below.

After 50 HITS:



And 50 more:



38

Experiment by Rob Miller

From <http://groups.csail.mit.edu/uid/deneme/>



So we can sometimes collect good training data.

But suppose we messed up. Our test setting doesn't look like the training data!

How can we bridge the domain gap?



# Finding more representative images

## Places365 Kitchen



[Fouhey et al., "From Lifestyle Vlogs to Everyday Actions", 2017]



# Finding more representative images

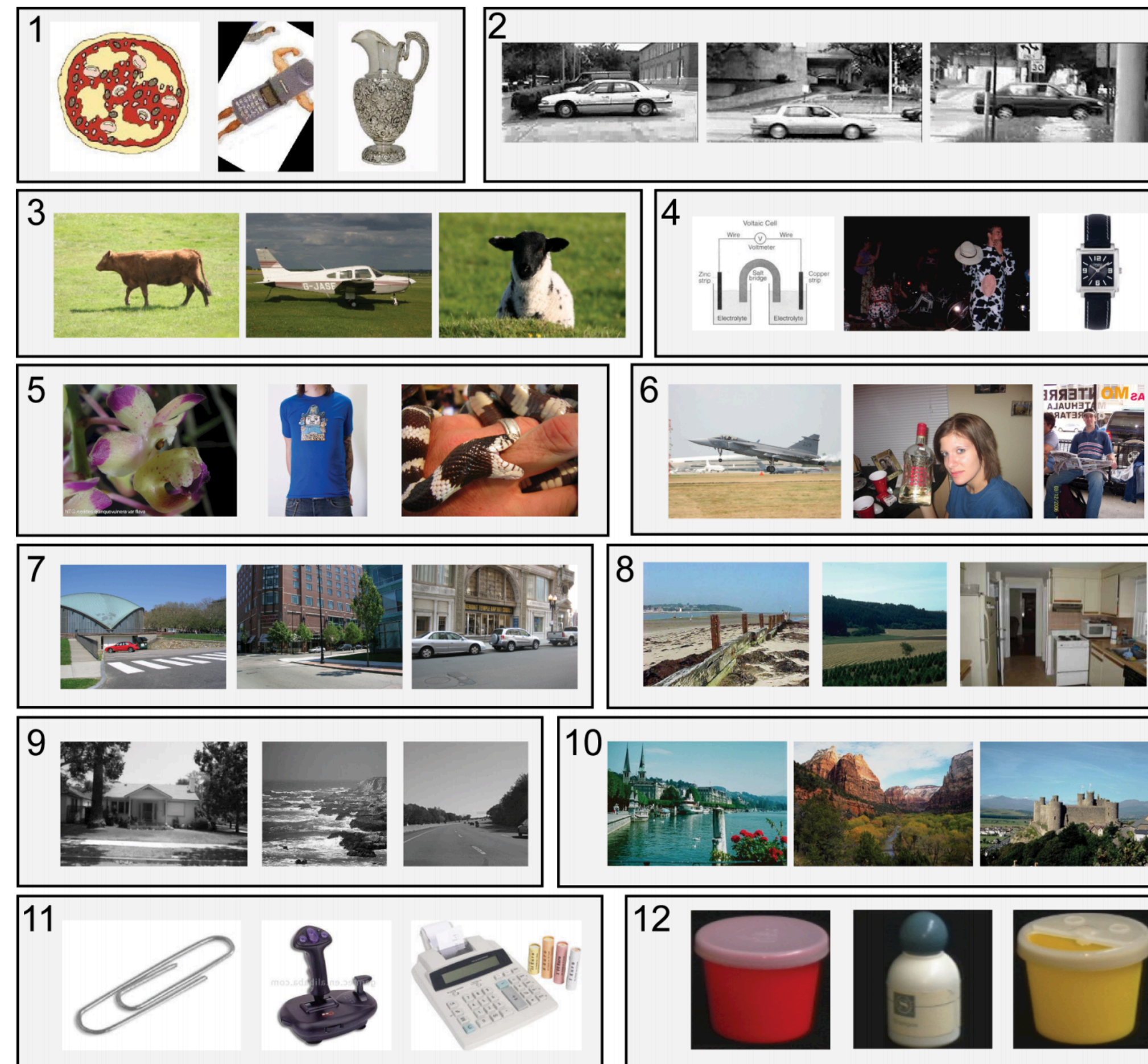
## VLOG Kitchen



[Fouhey et al., "From Lifestyle Vlogs to Everyday Actions", 2017]



# Name that dataset game



Caltech101 <input type="checkbox"/>	Tiny <input type="checkbox"/>	LabelMe <input type="checkbox"/>	15 Scenes <input type="checkbox"/>
MSRC <input type="checkbox"/>	Corel <input type="checkbox"/>	COIL-100 <input type="checkbox"/>	Caltech256 <input type="checkbox"/>
UIUC <input type="checkbox"/>	PASCAL 07 <input type="checkbox"/>	ImageNet <input type="checkbox"/>	SUN09 <input type="checkbox"/>

[Torralba and Efros, "An unbiased look at dataset bias," 2011]



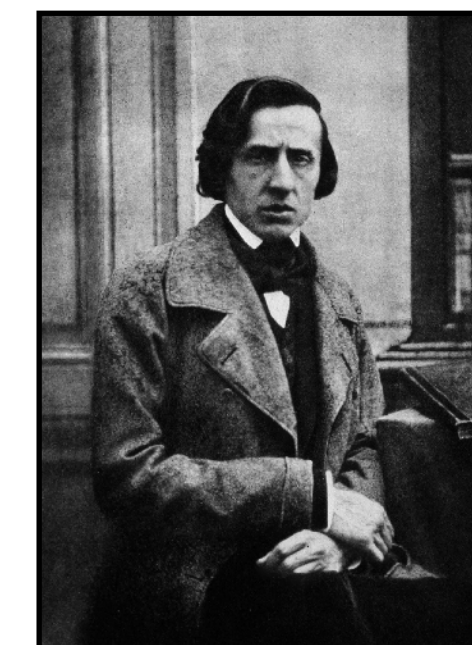
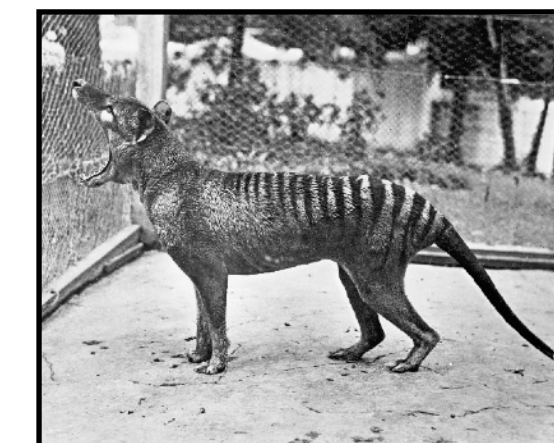
training domain

testing domain  
(where we actual use our model)

**Domain gap** between  $p_{\text{train}}$  and  $p_{\text{test}}$  will cause us to fail to generalize.

Space of natural images

Training data



Test data

43



*source domain*

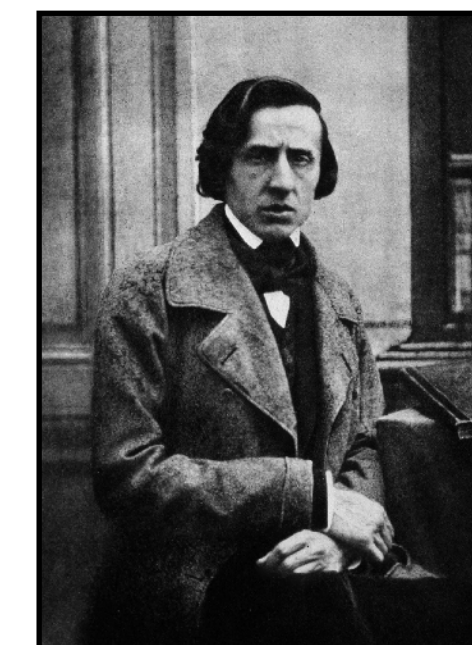
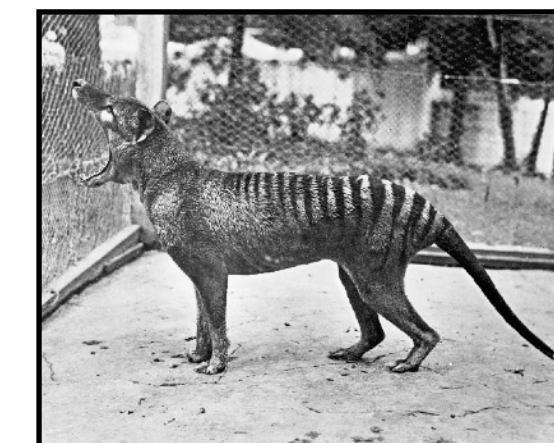
*target domain*

(where we actual use our model)

**Domain gap** between  $p_{\text{source}}$  and  $p_{\text{target}}$  will cause us to fail to generalize.

Space of natural images

Source data

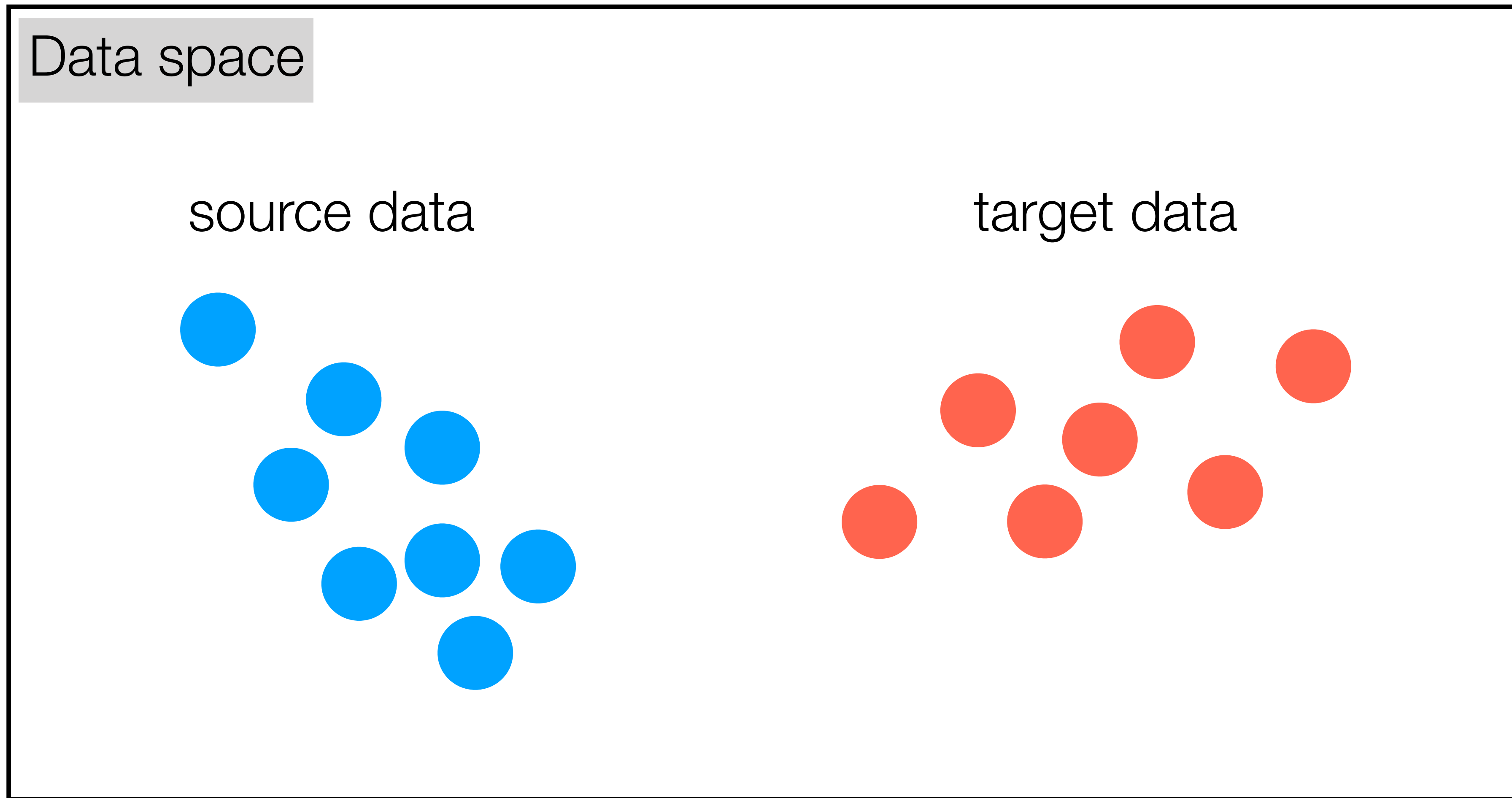


Target data

44



Idea #1: transform the target domain to look like the source domain



45

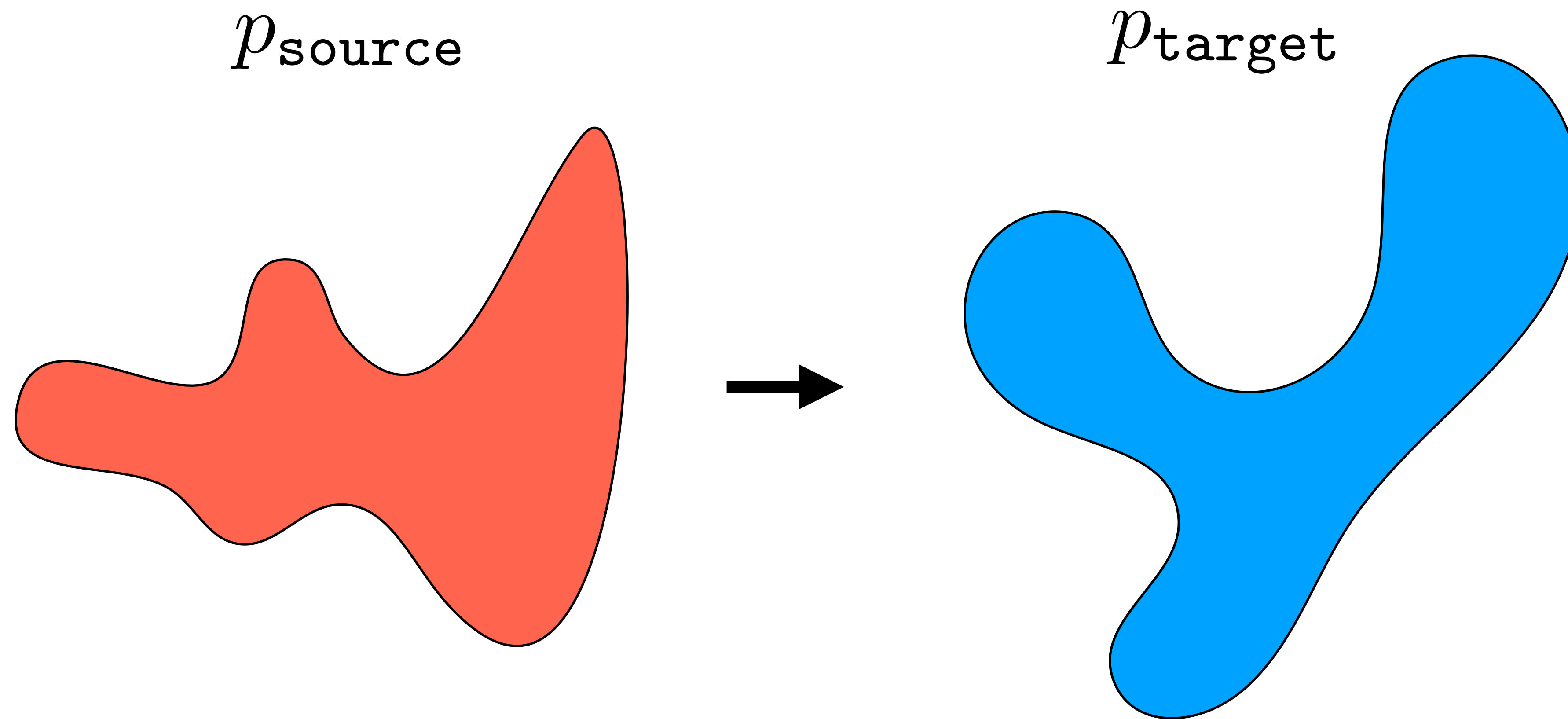
(Or vice versa)

This is called **domain adaptation**

# Domain adaptation

- We have source domain pairs  $\{\mathbf{x}^{\text{source}}, \mathbf{y}^{\text{source}}\}$
- Learn a mapping  $F: \mathbf{x}^{\text{source}} \rightarrow \mathbf{y}^{\text{source}}$
- We want to apply  $F$  to target domain data  $\mathbf{x}^{\text{target}}$
- Find transformation  $T: \mathbf{x}^{\text{target}} \rightarrow \mathbf{x}^{\text{source}}$
- Now apply  $F(T(\mathbf{x}^{\text{target}}))$  to predict  $\mathbf{y}^{\text{target}}$



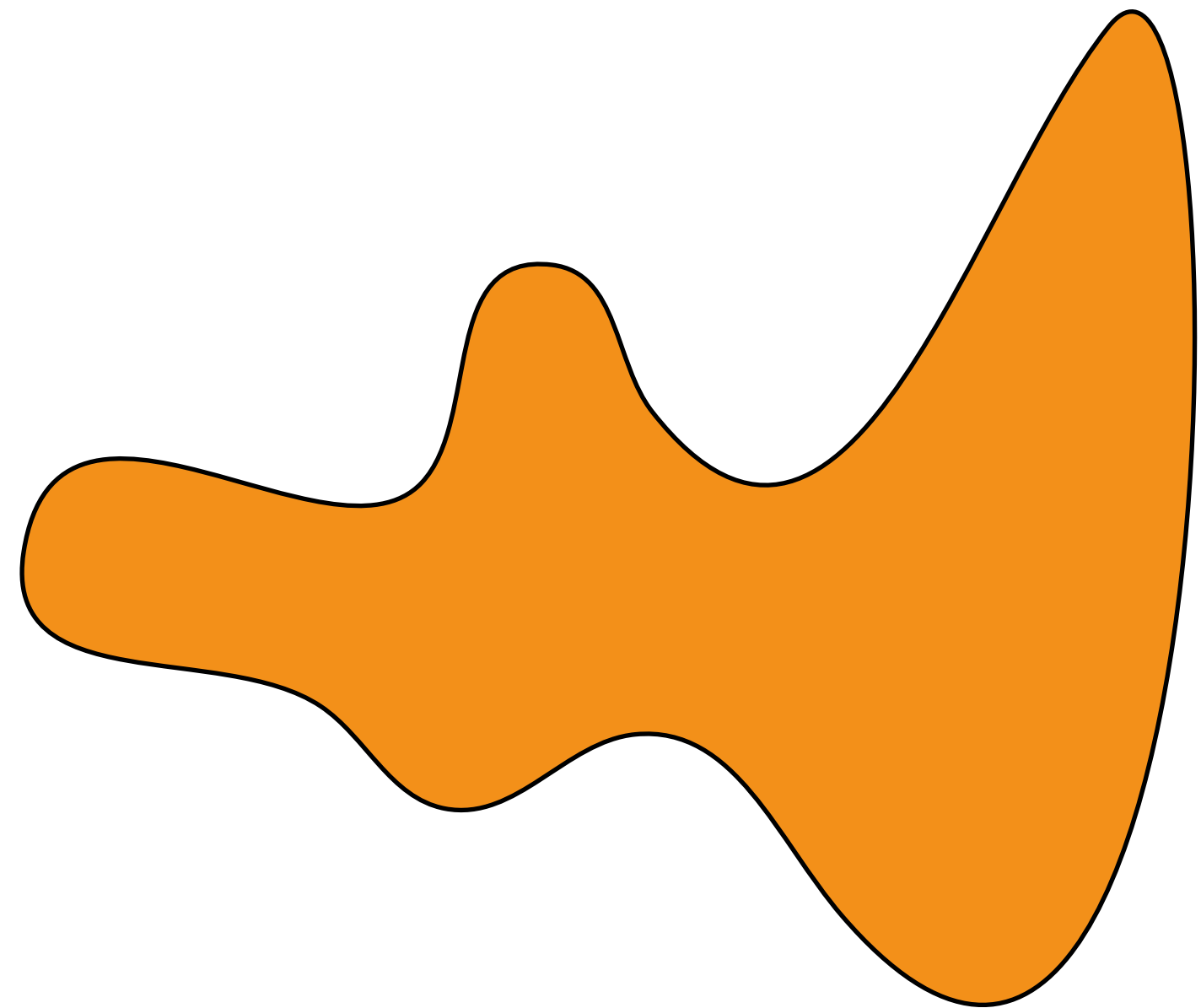


It's a just another distribution mapping problem!

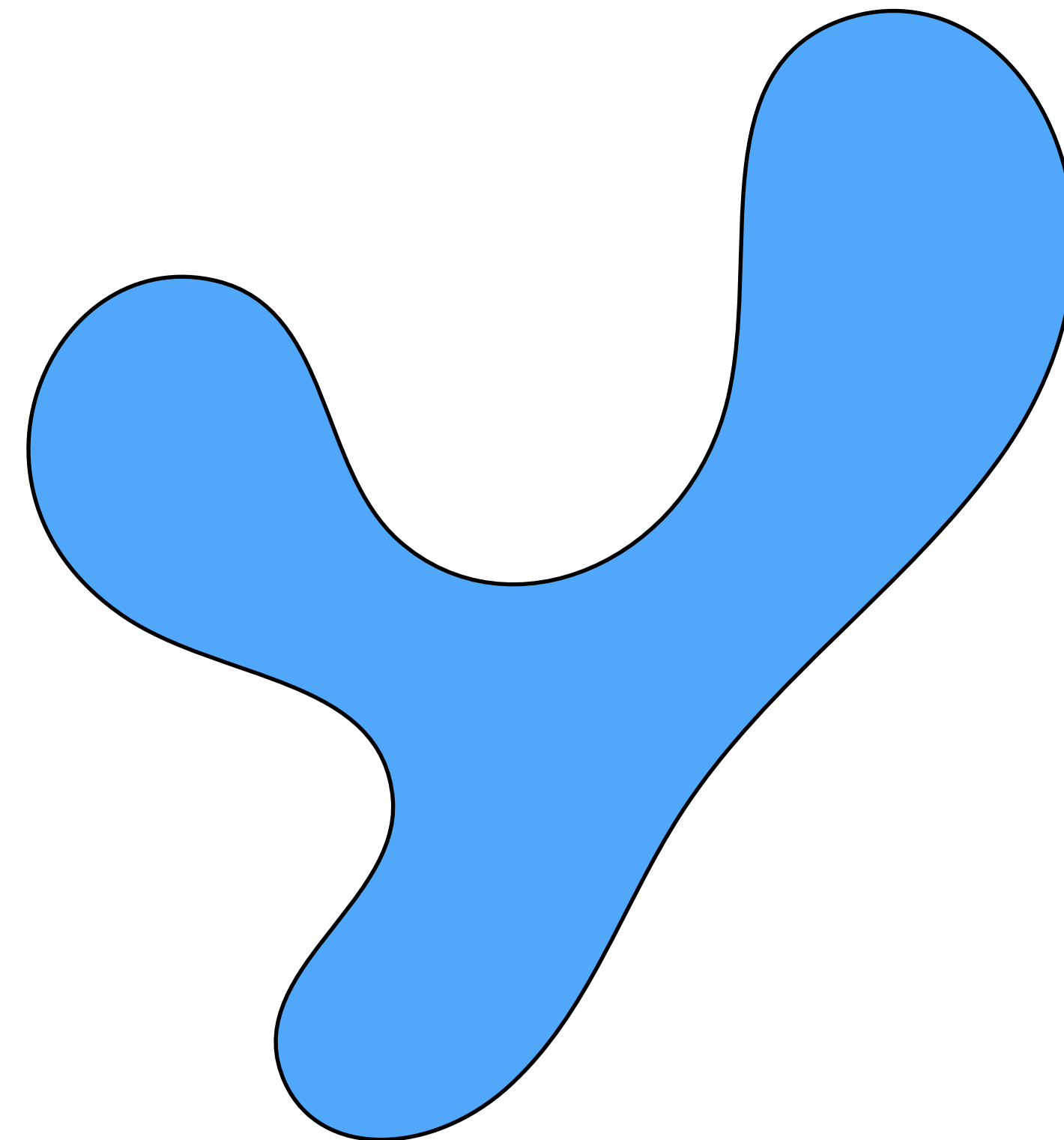
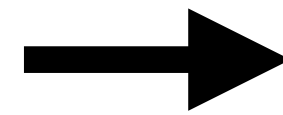
# CycleGAN

Horses

Zebras



$X$

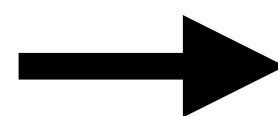
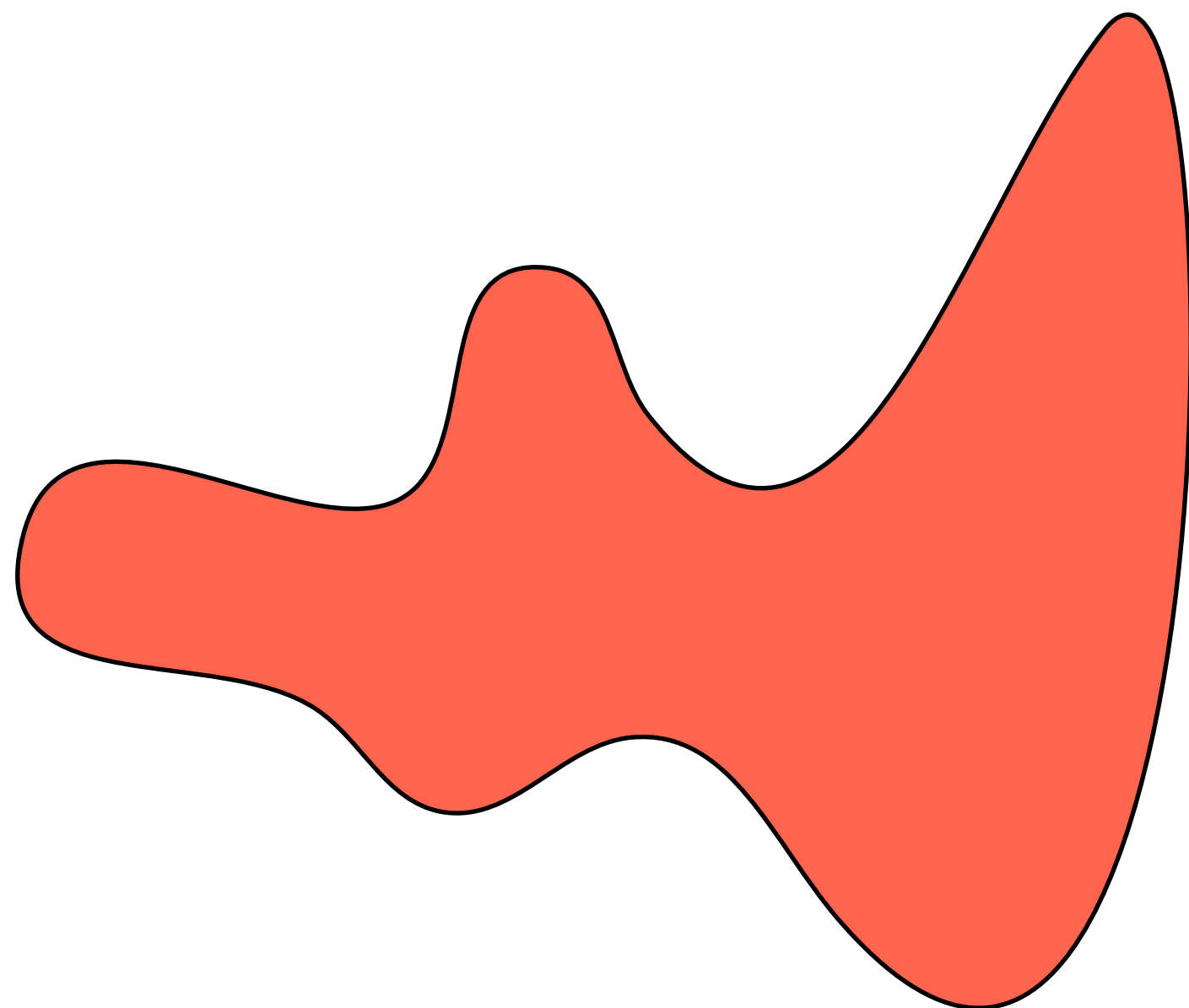


$Y$

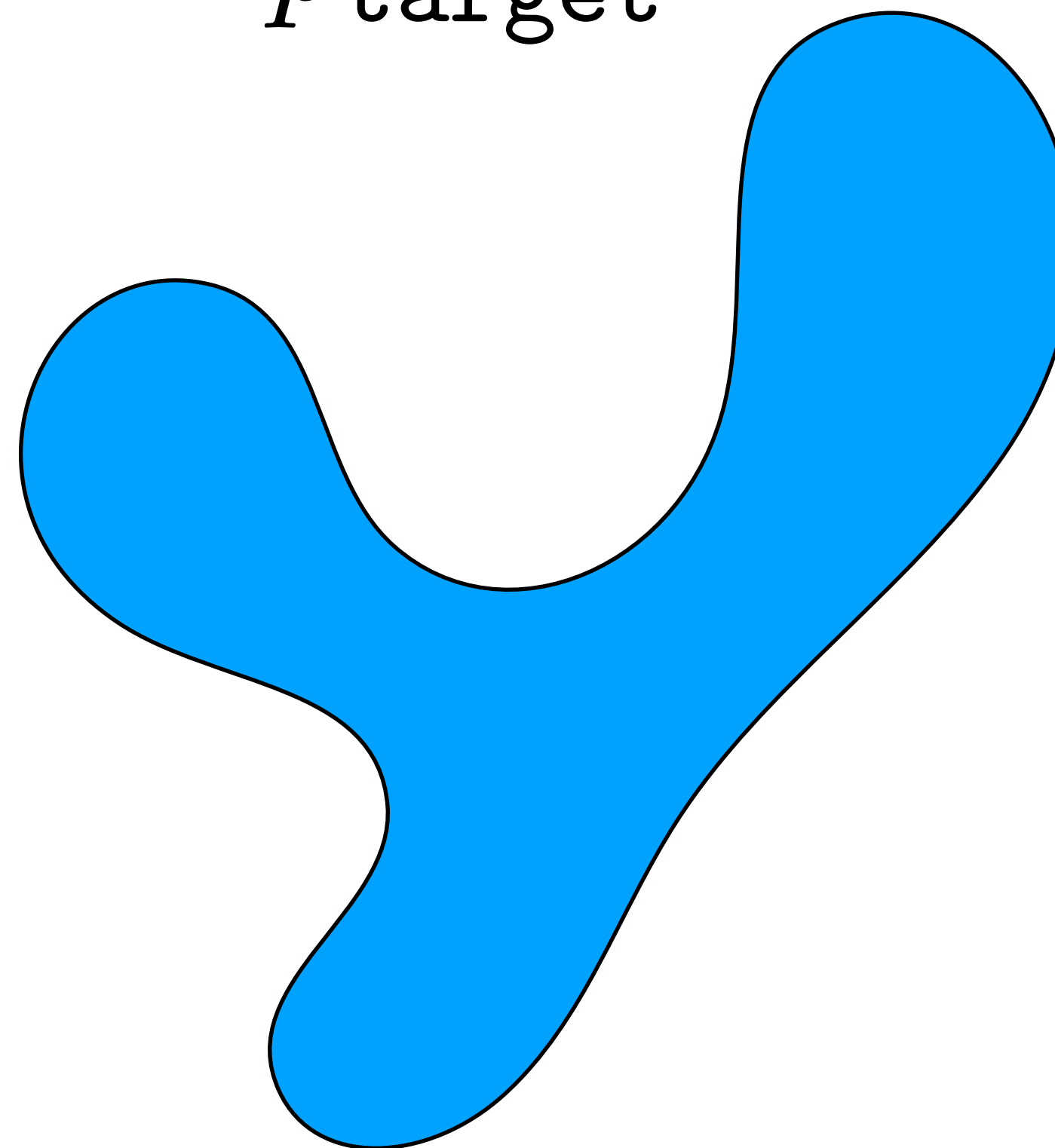


# Domain adaptation

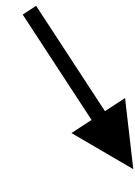
$p_{\text{source}}$



$p_{\text{target}}$



*source domain*



*target domain*

(where we actual use our model)



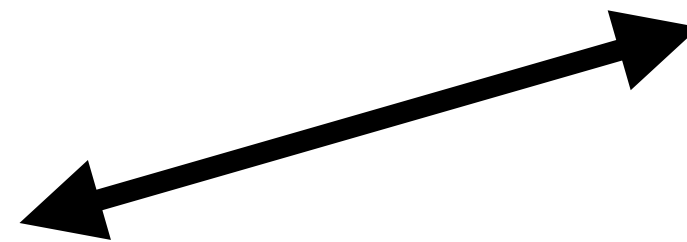
**Domain gap** between  $p_{\text{source}}$  and  $p_{\text{target}}$  will cause us to fail to generalize.

Space of images

Source data



Target data

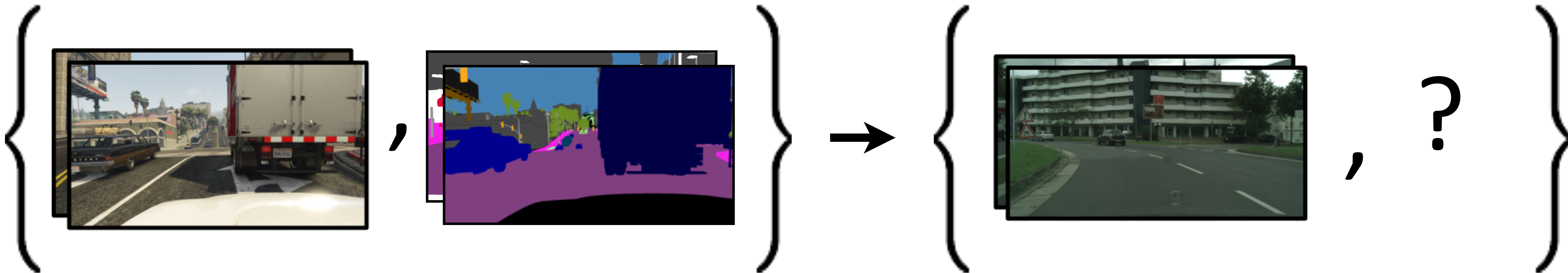




# CyCADA: Cycle-Consistent Adversarial Domain Adaptation

Source domain

Target domain



[Hoffman, Tzeng, Park, Zhu, Isola, Saenko, Darrell, Efros, arXiv 2017]

# CycleGAN



Training data





# CycleGAN



Training data



CycleGAN



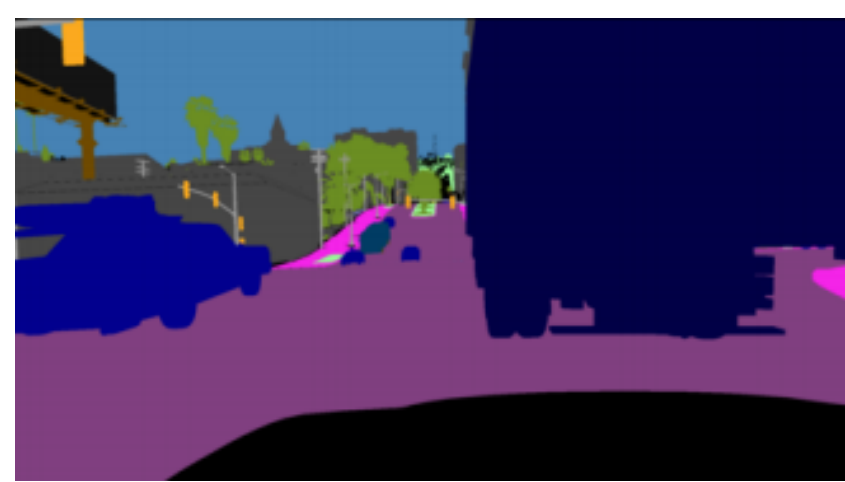
FCN



Training data

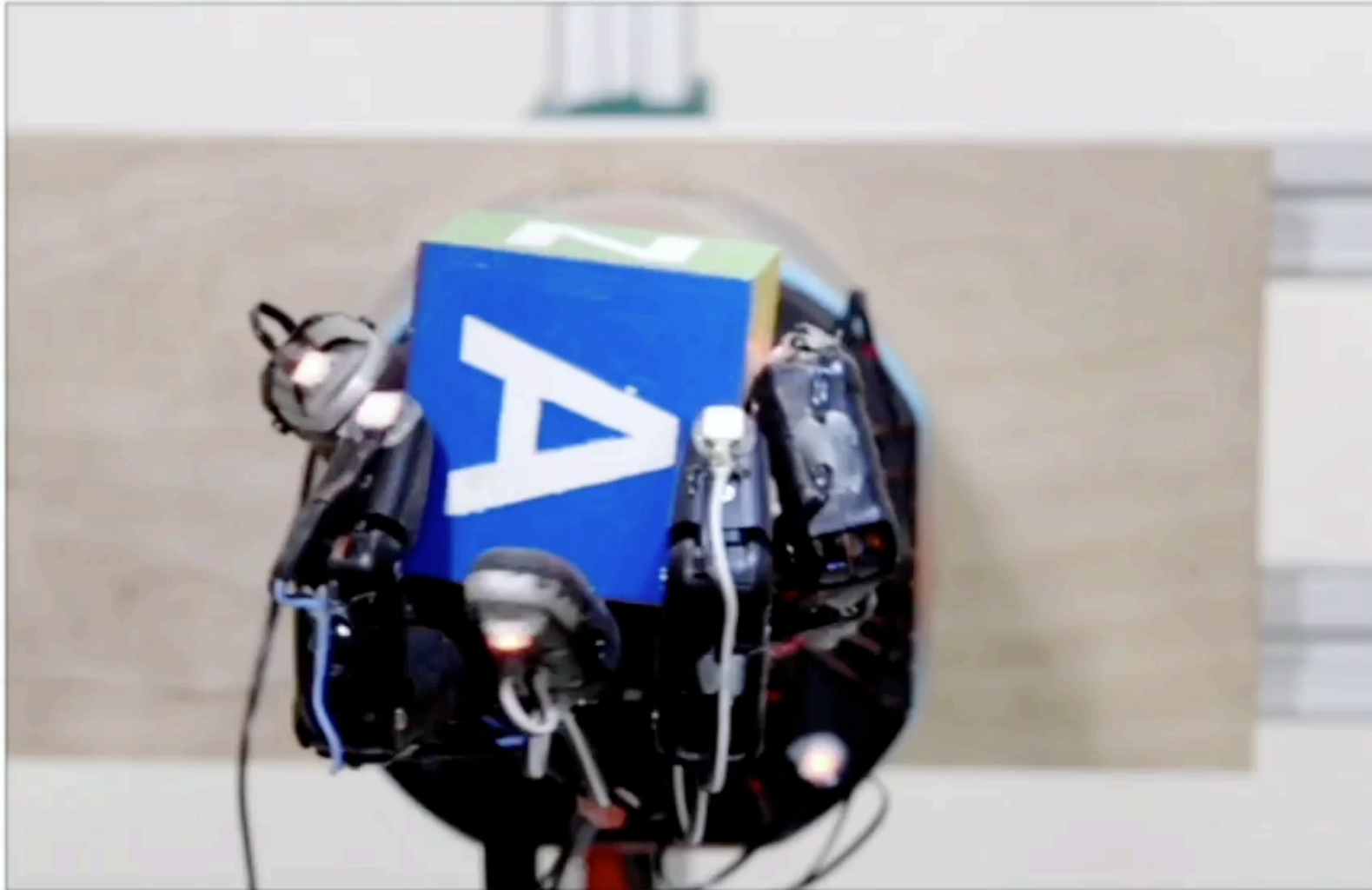


,

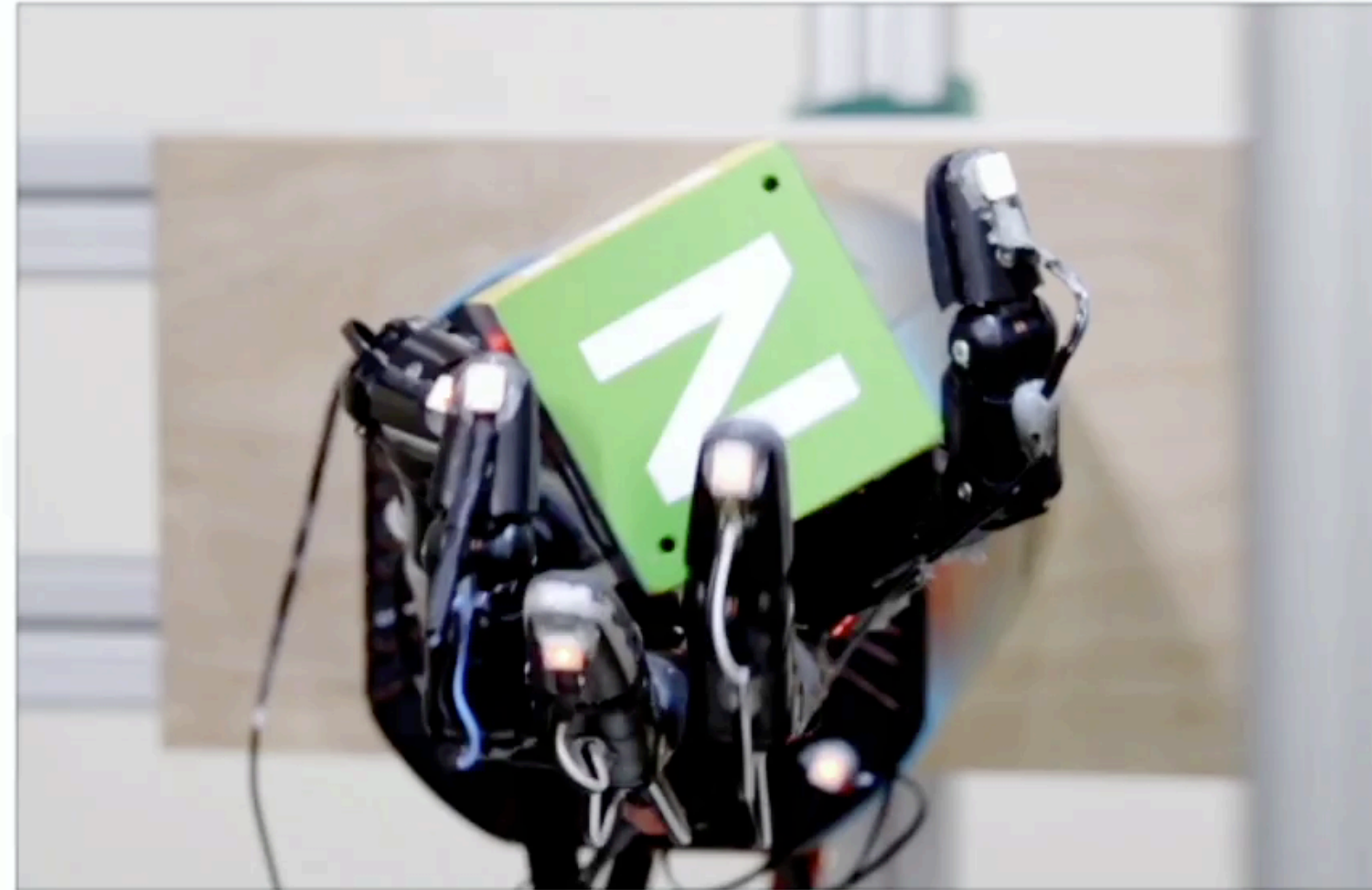




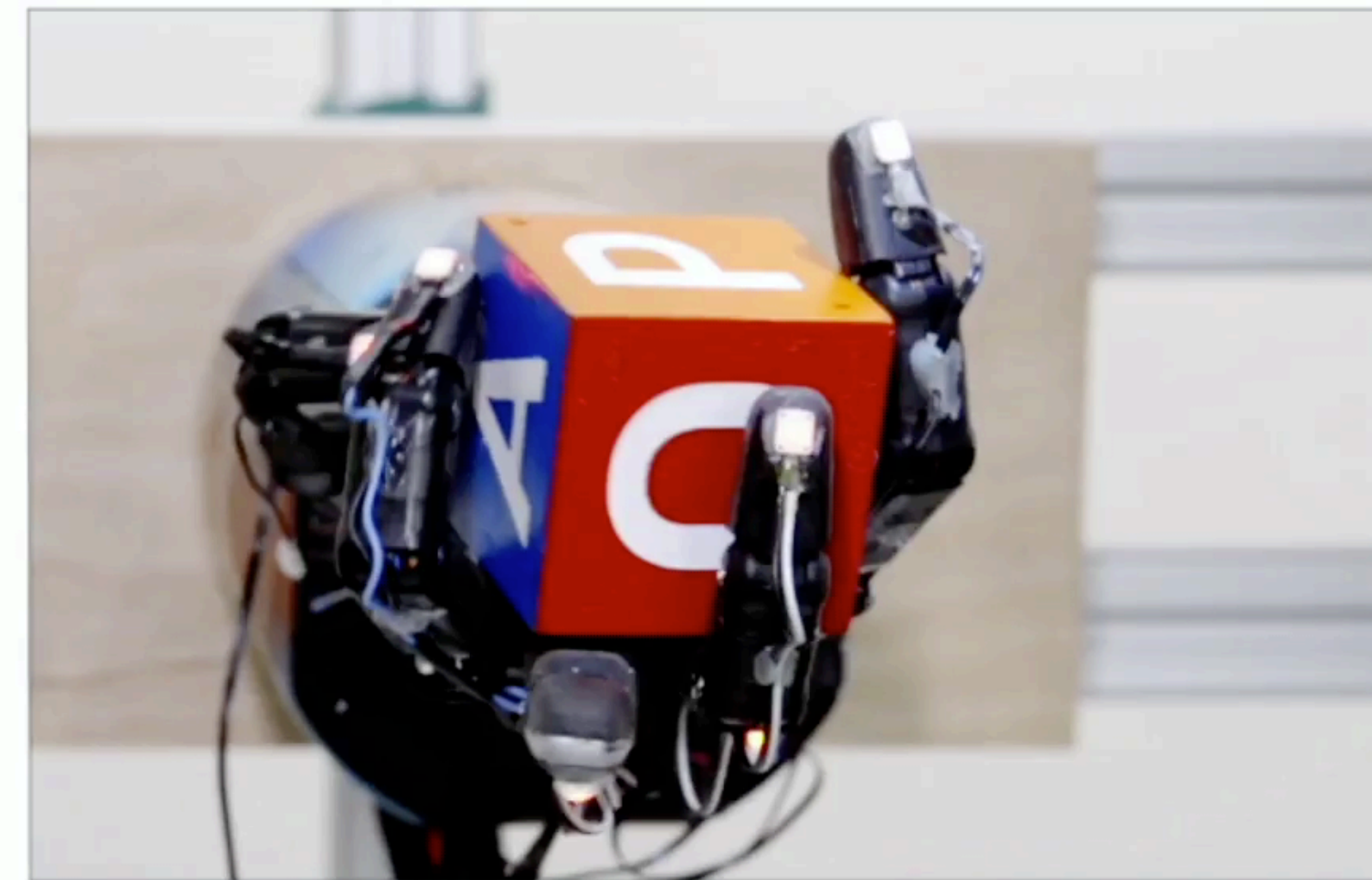
# OpenAI Dactyl



**FINGER PIVOTING**

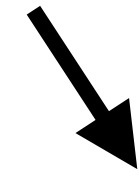


**SLIDING**



**FINGER GAITING**

*source domain*



*target domain*

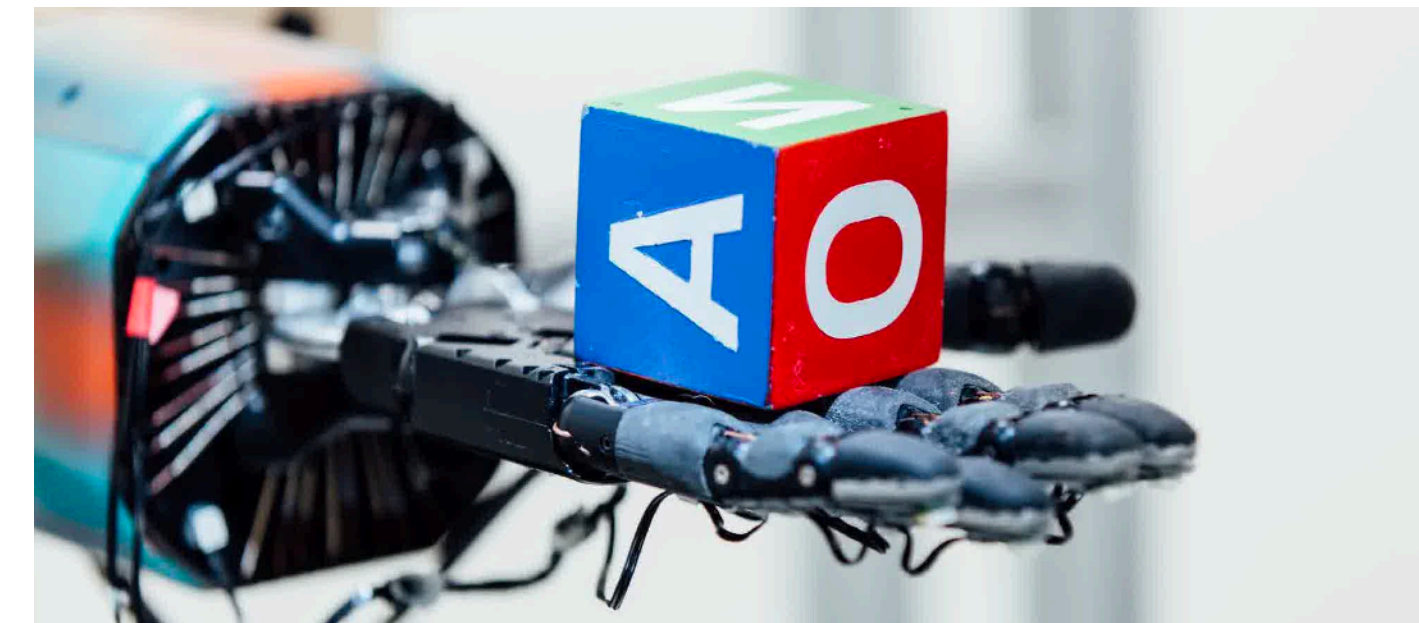
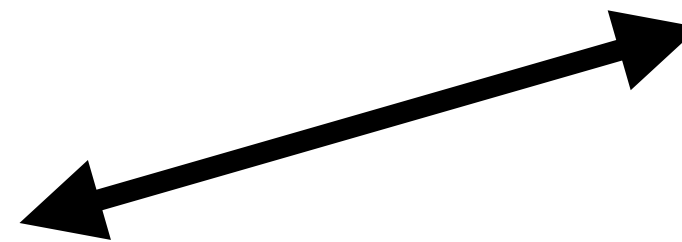
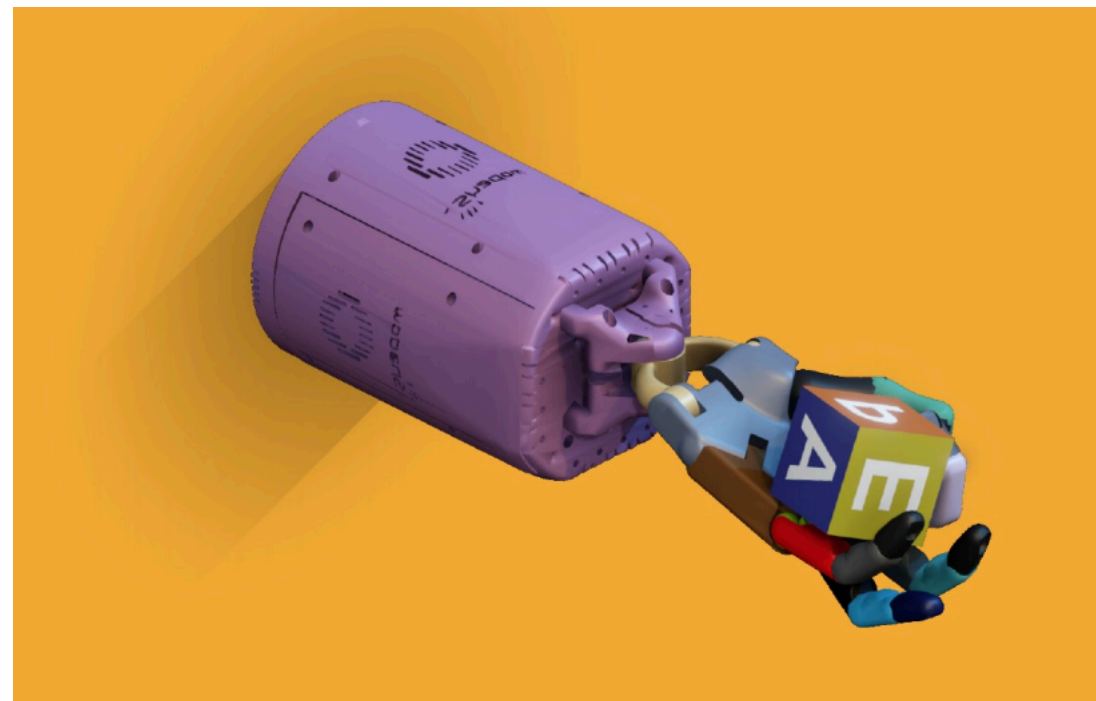
(where we actual use our model)



**Domain gap** between  $p_{\text{source}}$  and  $p_{\text{target}}$  will cause us to fail to generalize.

Space of images

Source data

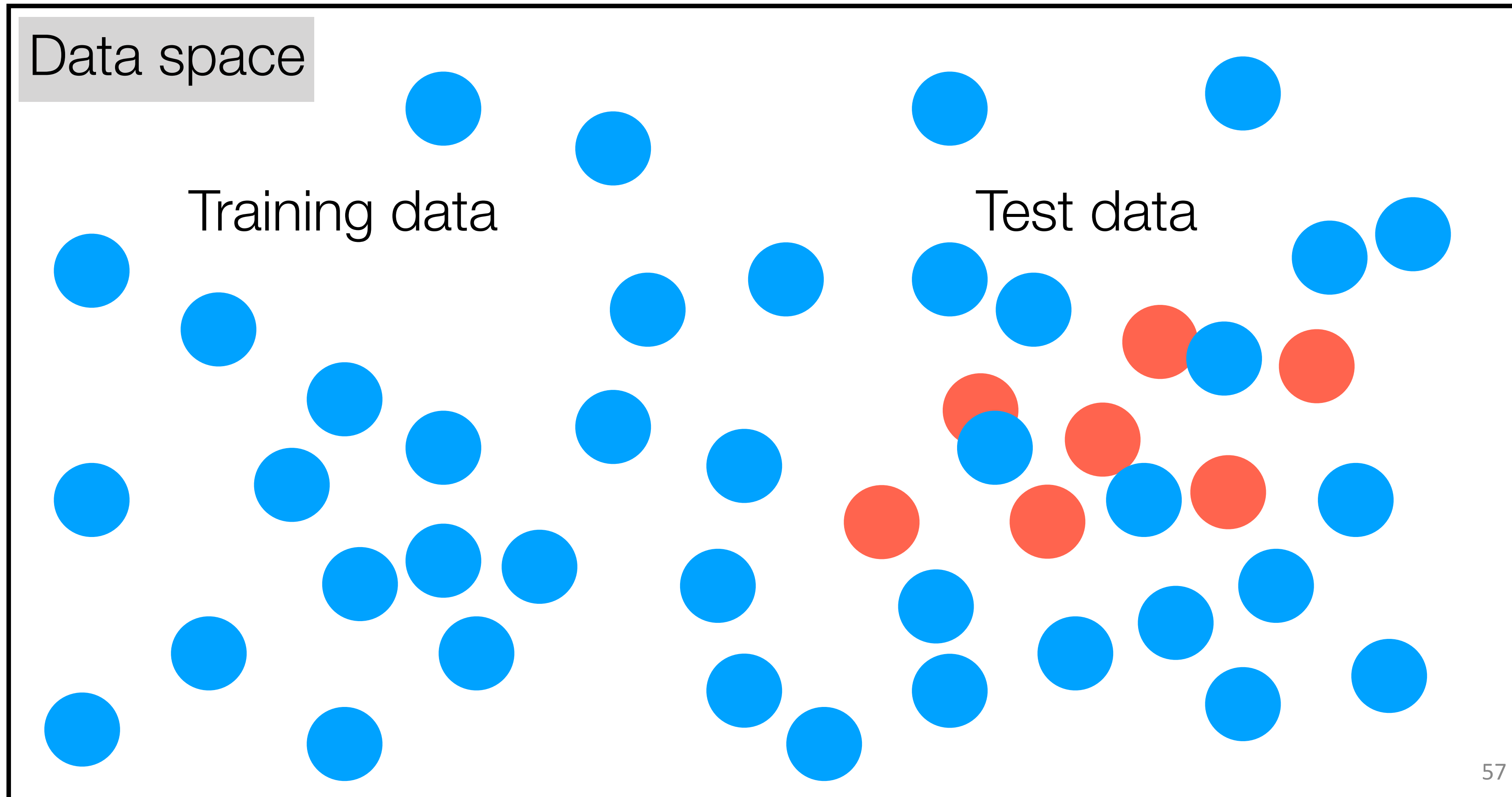


Target data

56

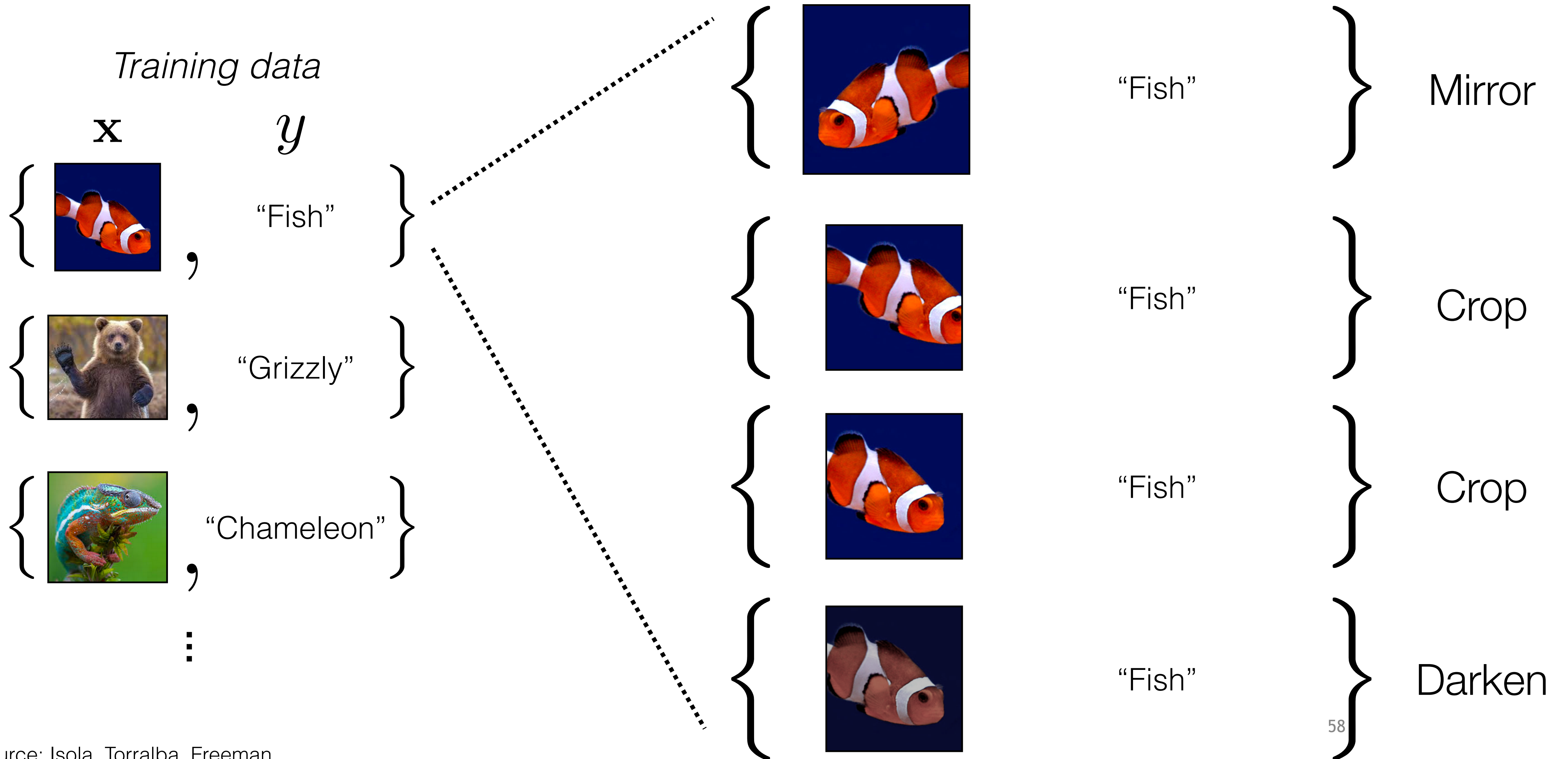


Idea #2: train on randomly perturbed data, so that test set just looks like another random perturbation



This is called **domain randomization** or **data augmentation**

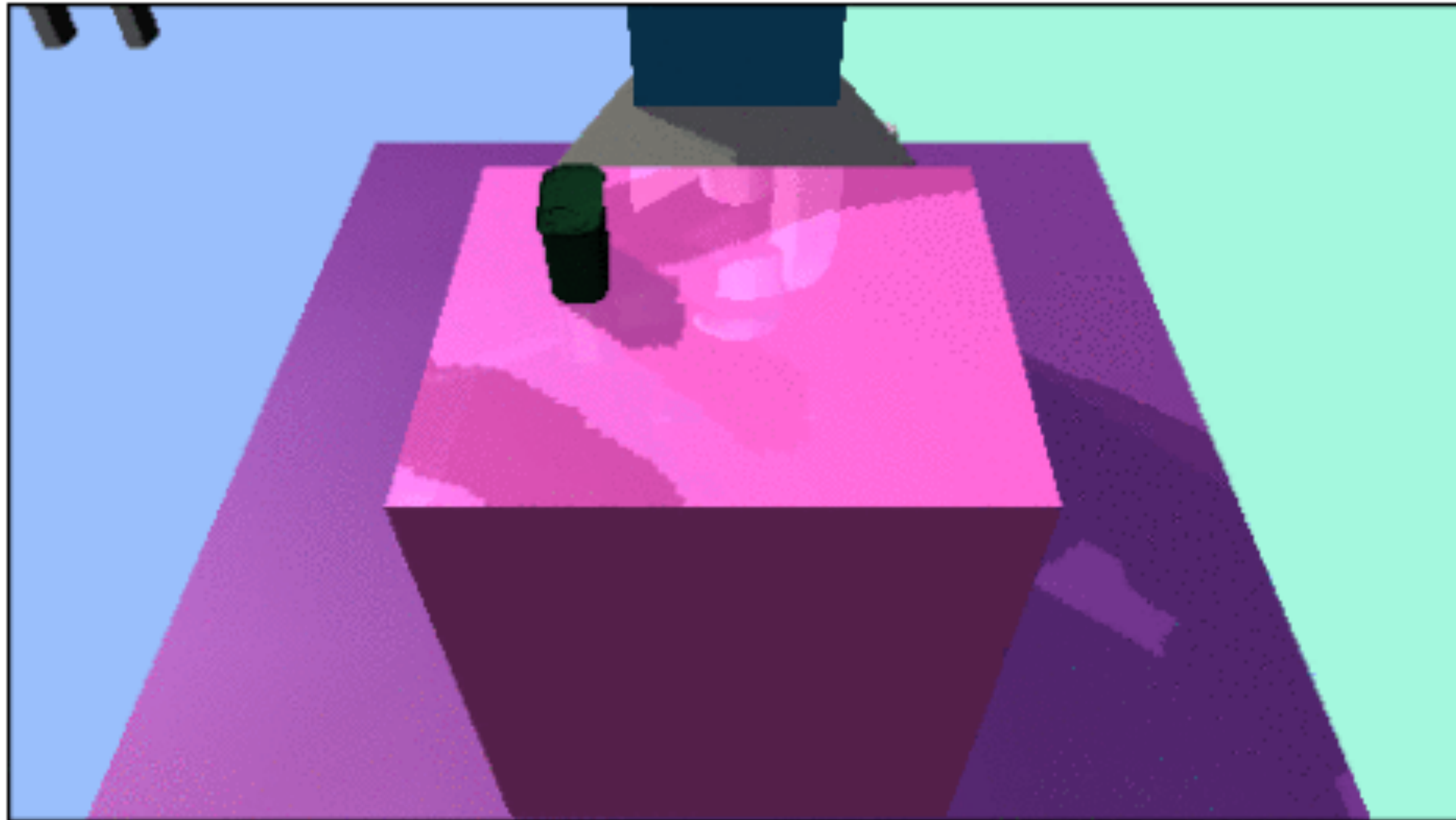
# Data augmentation



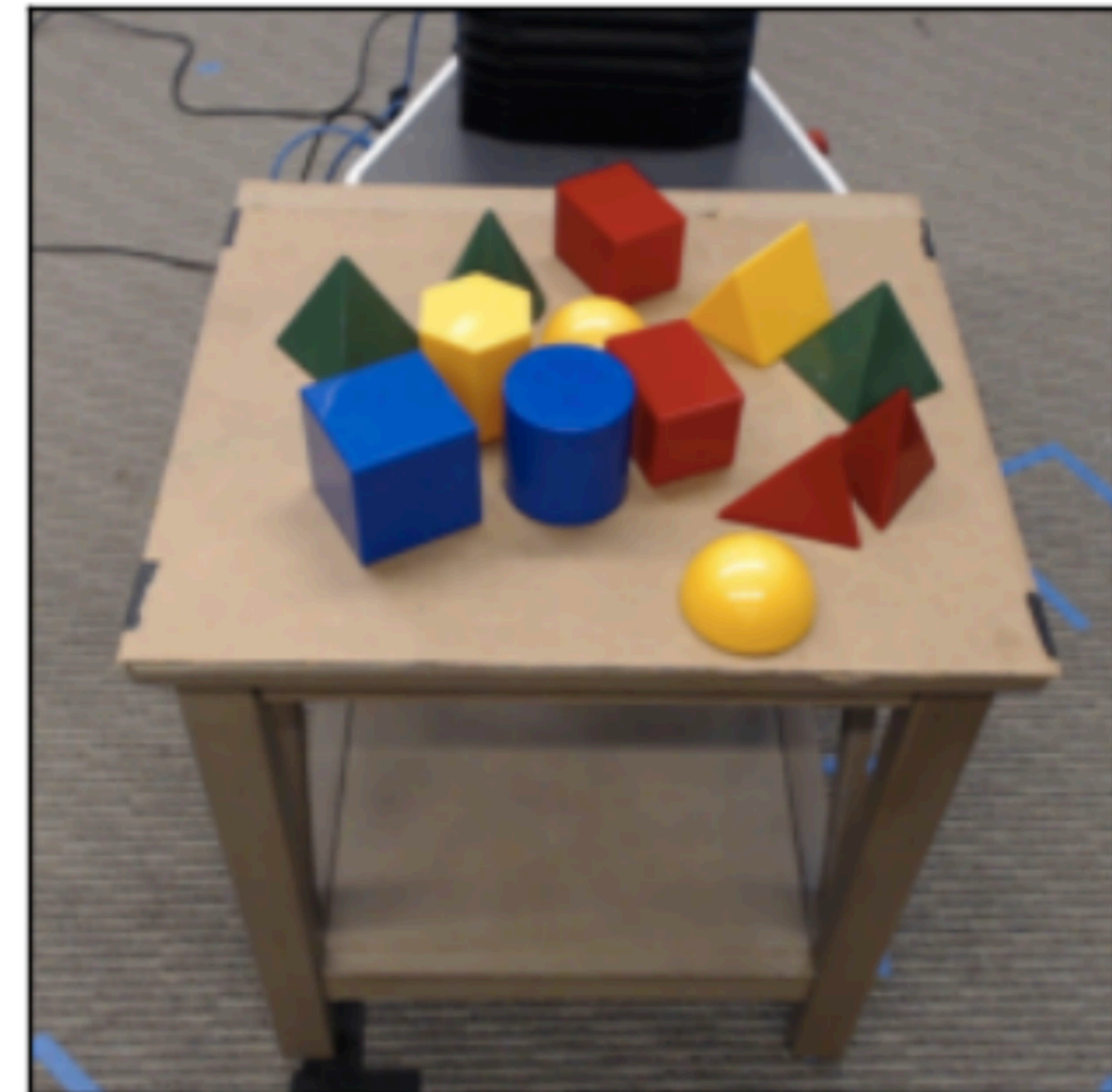


# Domain randomization

Training data



Test data



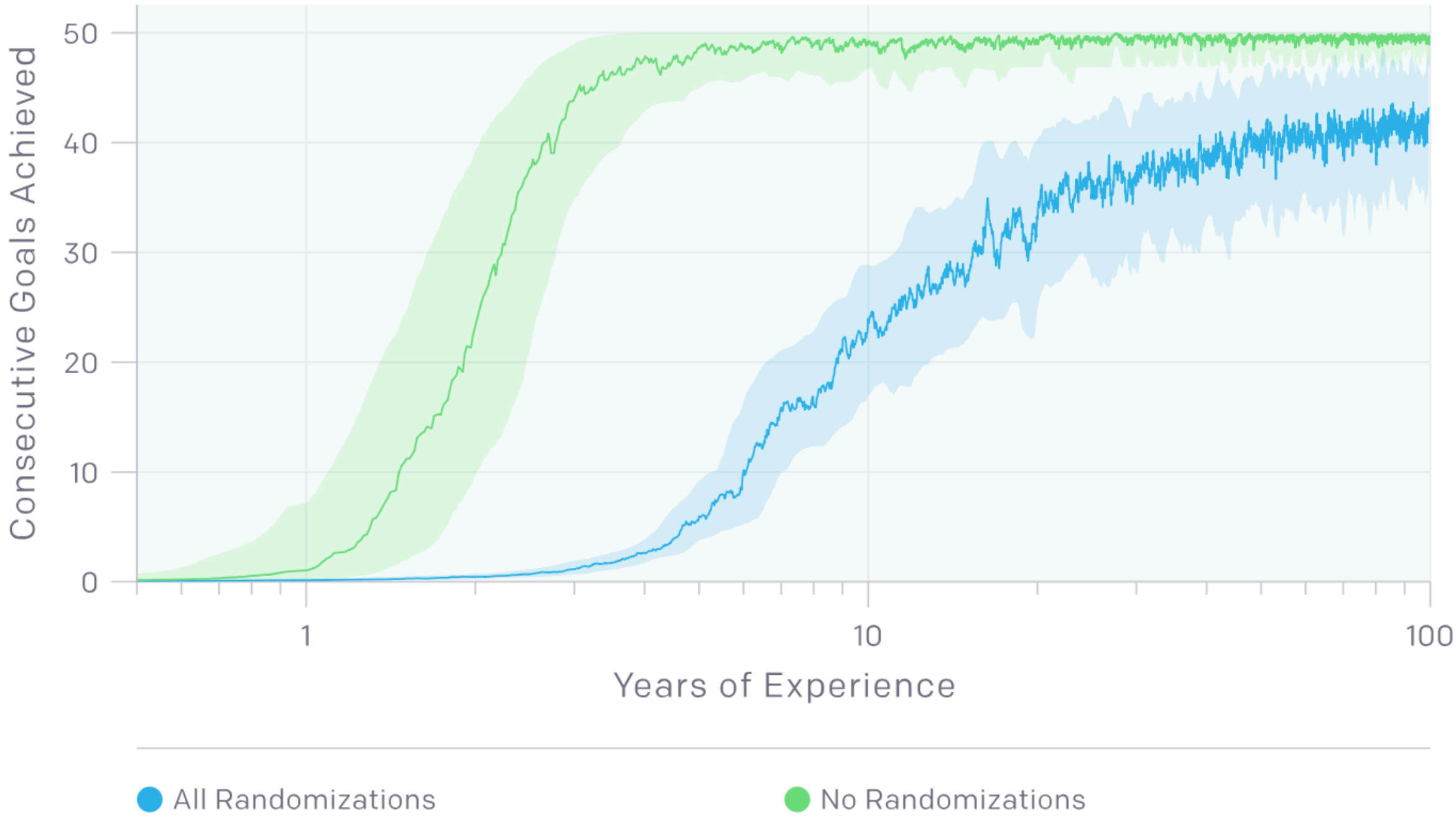
[Sadeghi & Levine 2016]

Above example is from [Tobin et al. 2017]



Table 1: Ranges of physics parameter randomizations.

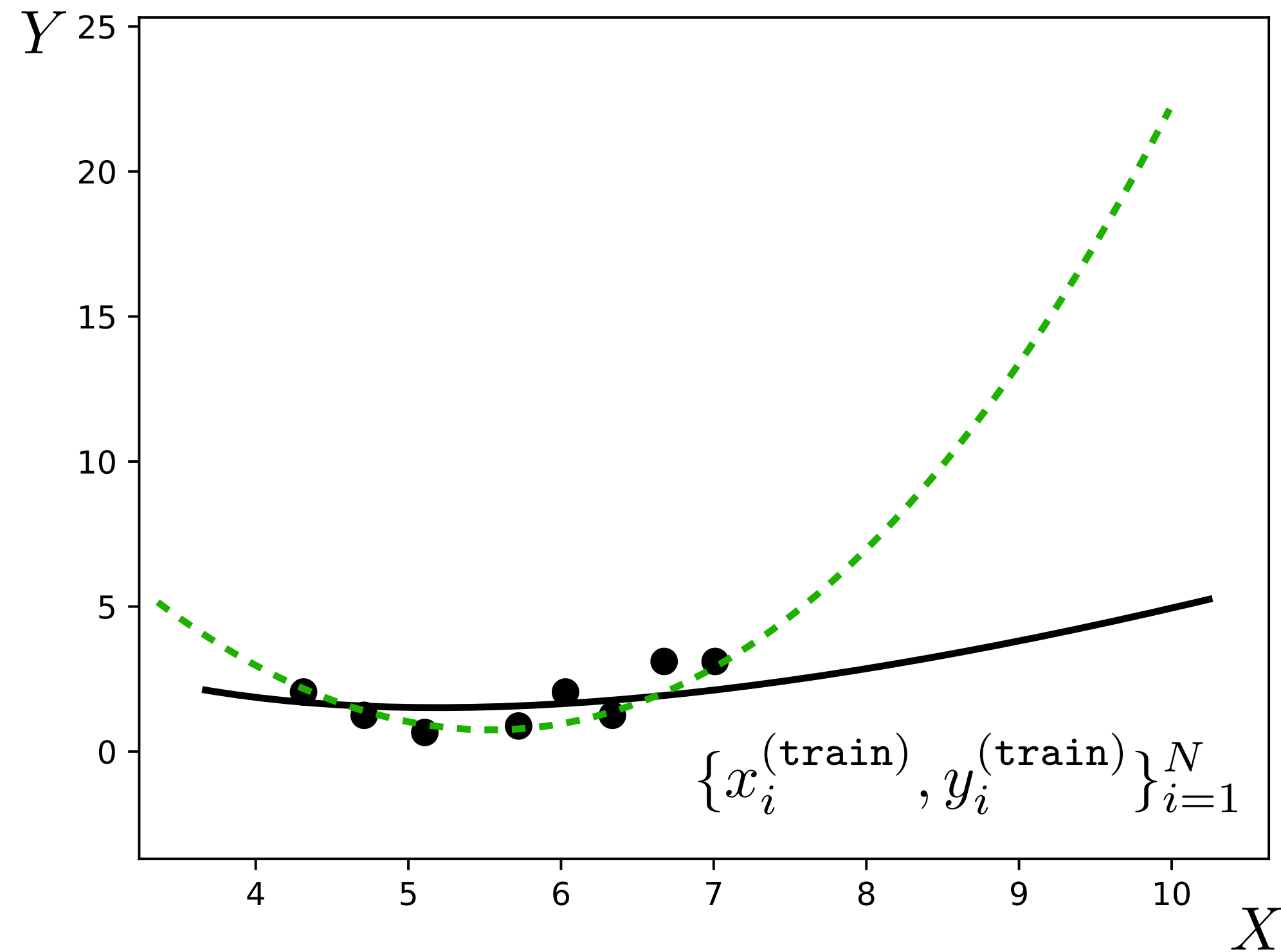
Parameter	Scaling factor range	Additive term range
object dimensions	$\text{uniform}([0.95, 1.05])$	
object and robot link masses	$\text{uniform}([0.5, 1.5])$	
surface friction coefficients	$\text{uniform}([0.7, 1.3])$	
robot joint damping coefficients	$\text{loguniform}([0.3, 3.0])$	
actuator force gains (P term)	$\text{loguniform}([0.75, 1.5])$	
joint limits		$\mathcal{N}(0, 0.15) \text{ rad}$
gravity vector (each coordinate)		$\mathcal{N}(0, 0.4) \text{ m/s}^2$



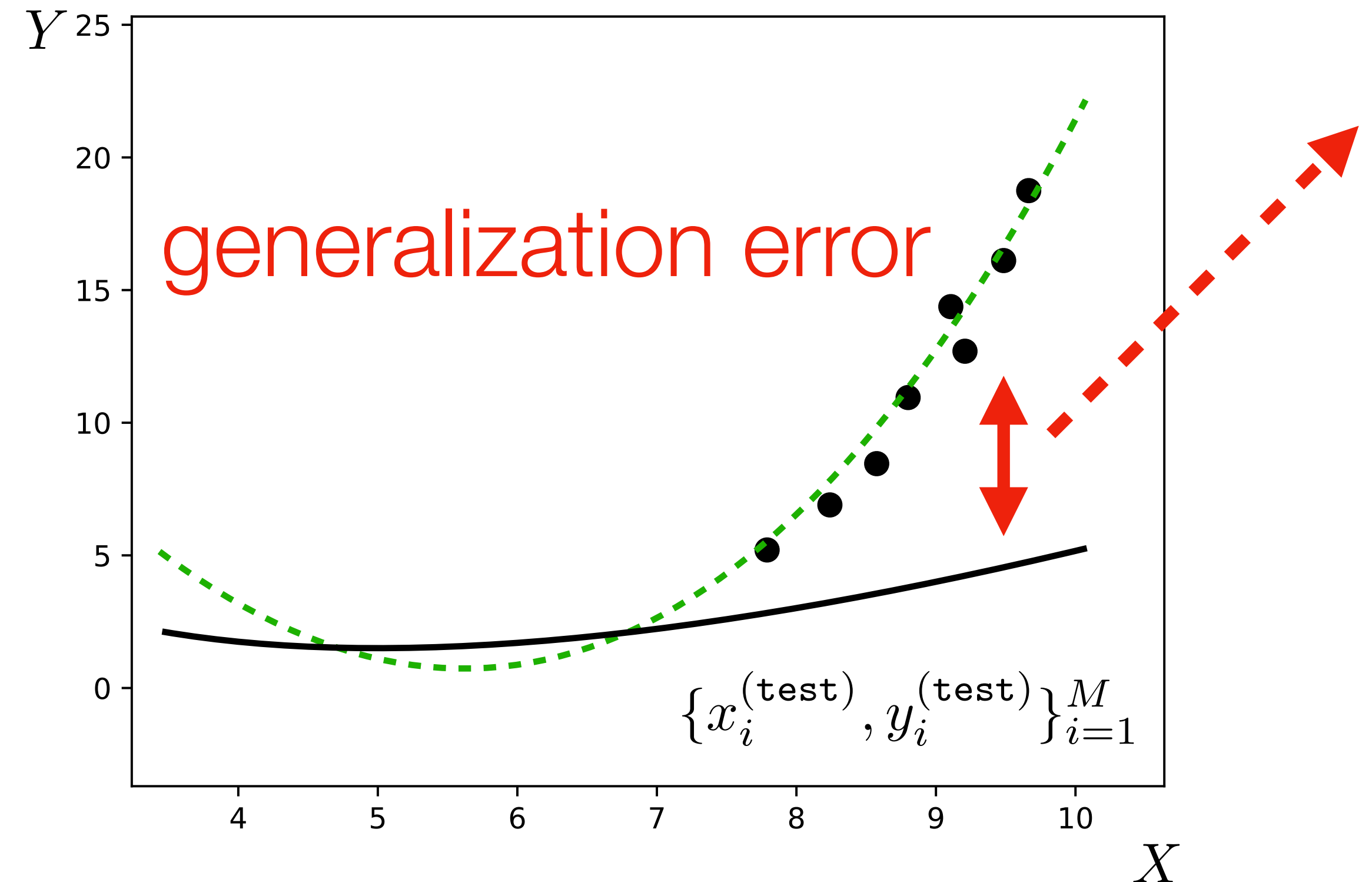


What if we go way outside of the training distribution?

Training data



Test data

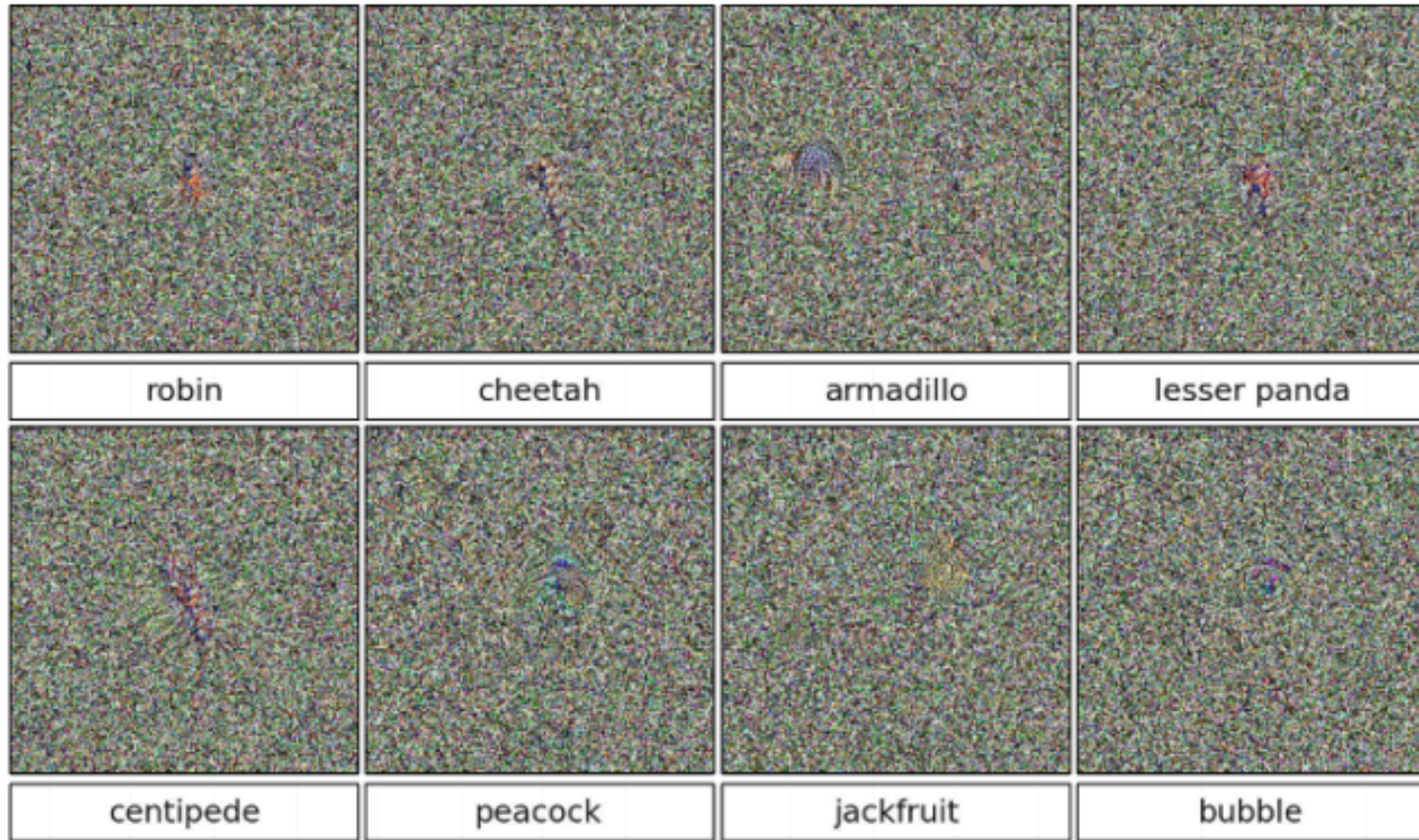


Our training data did not cover the part of the distribution that was tested  
**(biased data)**



# “Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images”

[Nguyen, Yosinski, and Clune, CVPR 2015]

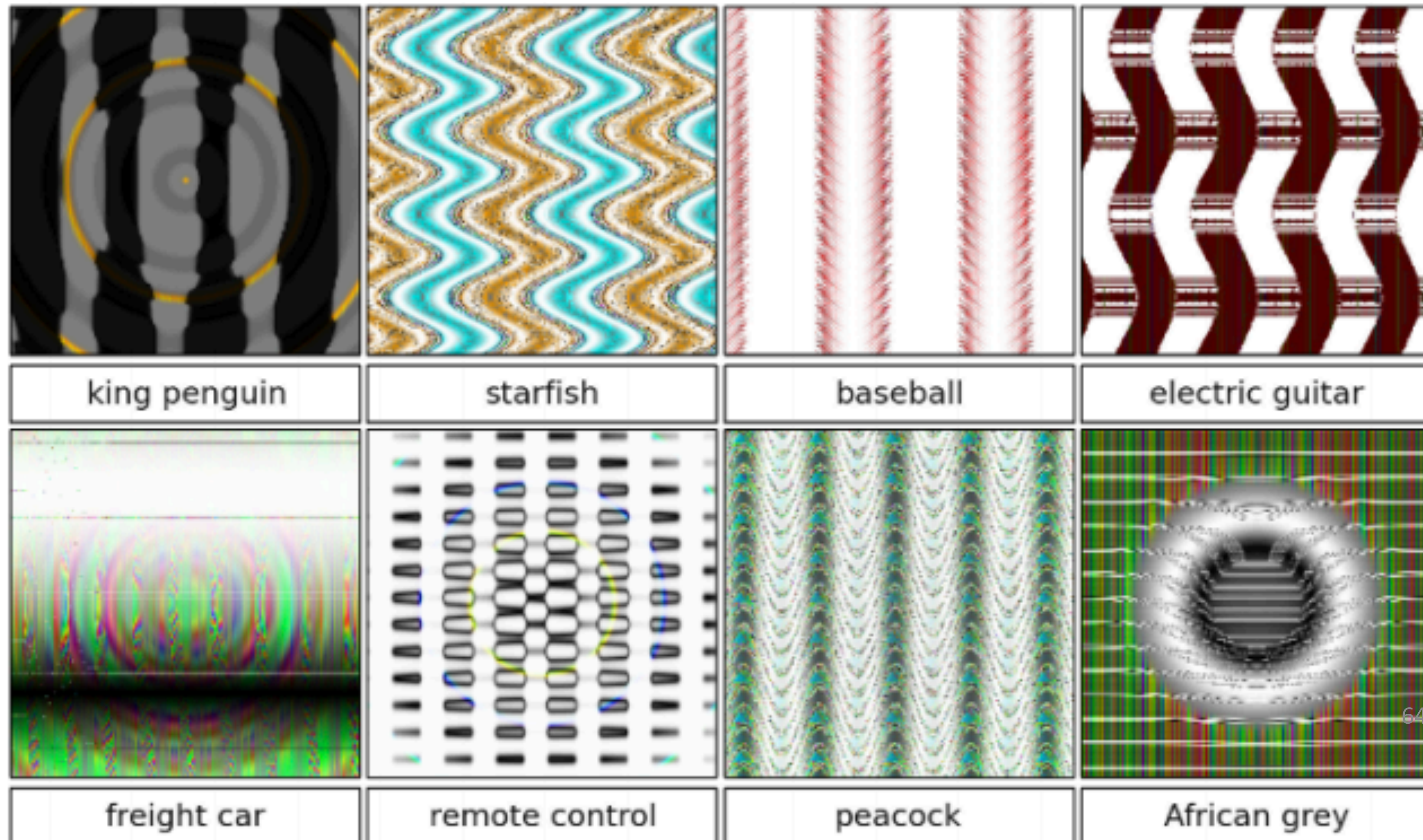


63



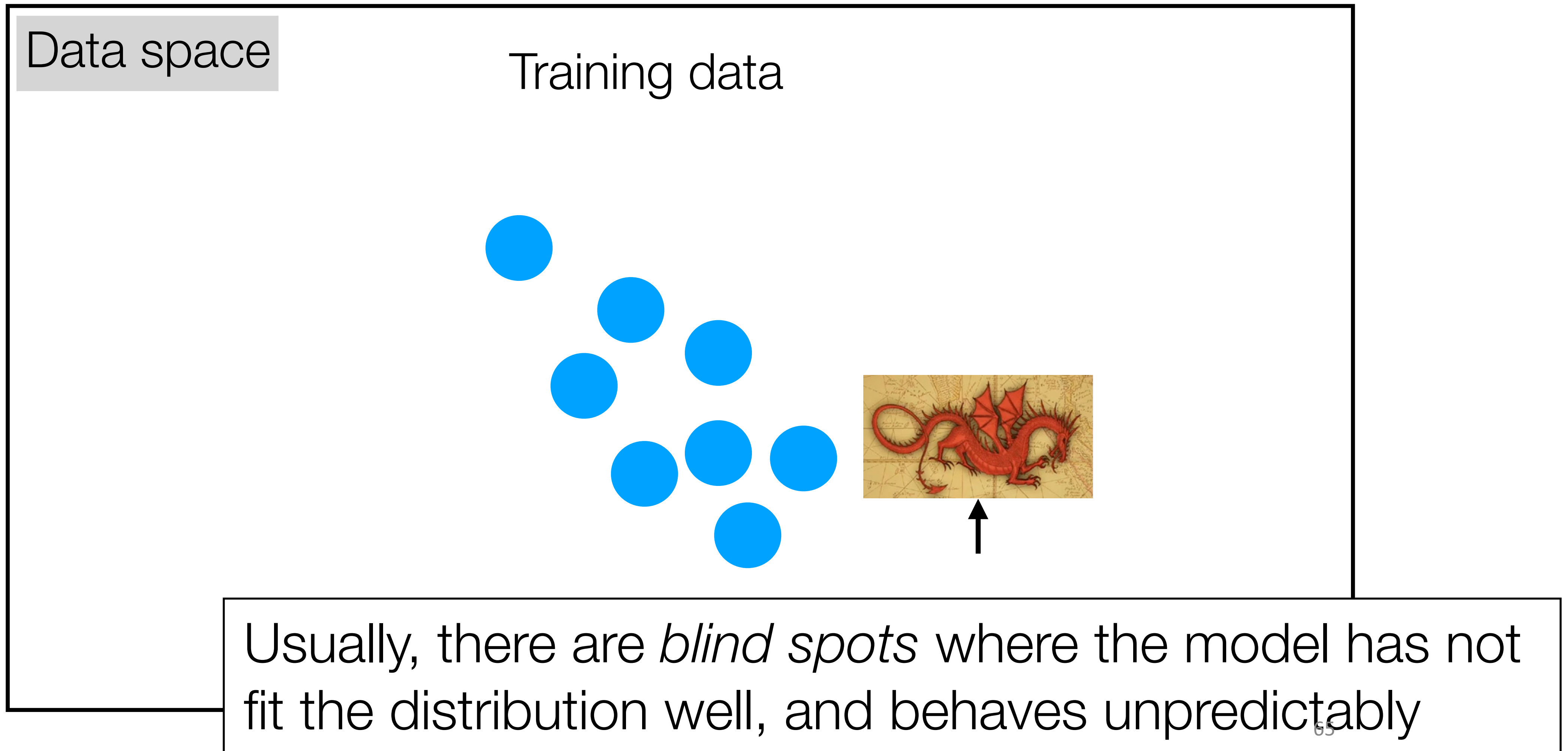
# “Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images”

[Nguyen, Yosinski, and Clune, CVPR 2015]





# Weirdness of high-dimensional space:

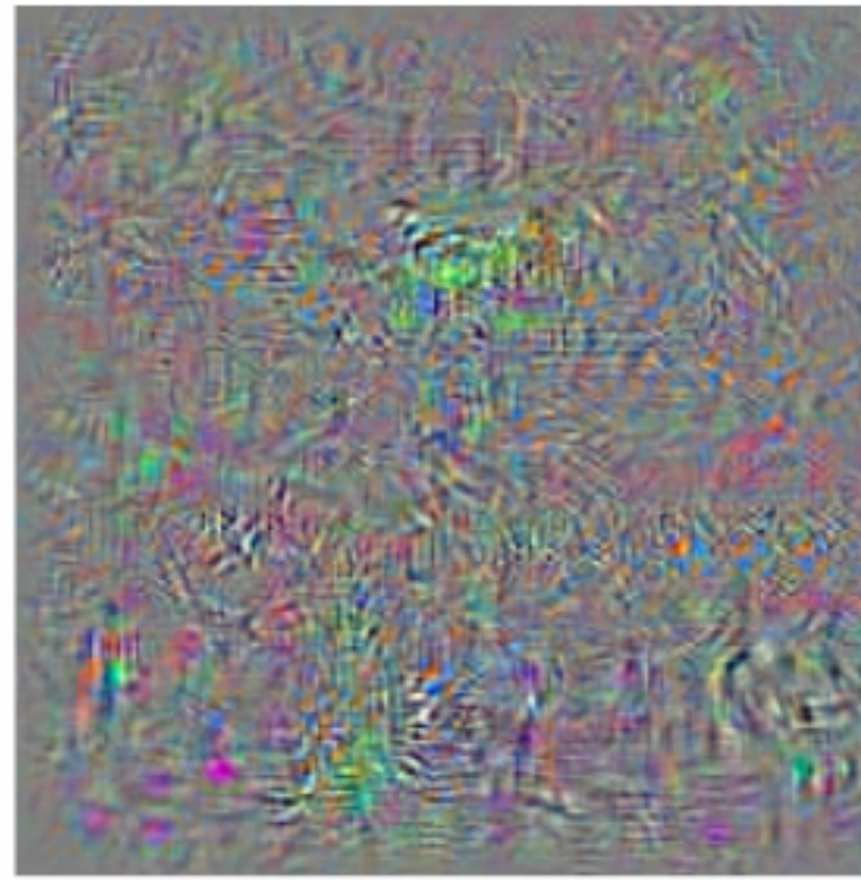


# Adversarial noise

$\mathbf{x}$



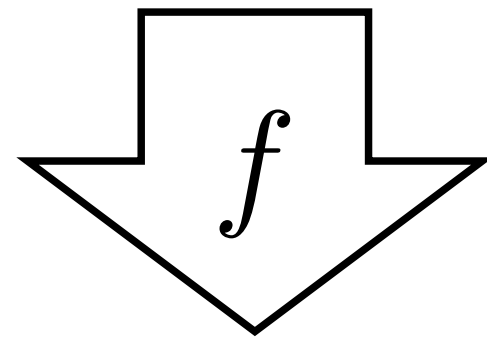
$\mathbf{r}$



+

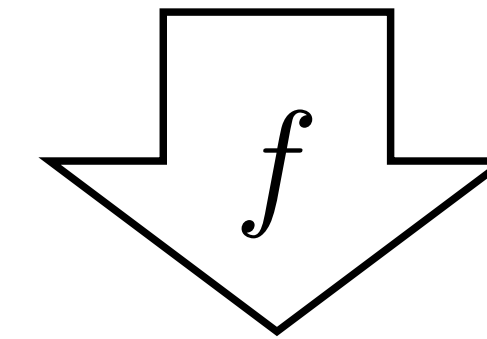
=

$\mathbf{x} + \mathbf{r}$



$y$

“School bus”



“Ostrich”

$$\arg \max_{\mathbf{r}} p(y = \text{ostrich} | \mathbf{x} + \mathbf{r}) \quad \text{subject to} \quad \|\mathbf{r}\| < \epsilon$$

[“Intriguing properties of neural networks”, Szegedy et al. 2014]



# Anything to worry about?

“NO Need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles”, Lu et al. 2017



(Early) 2017's attacks fail on physical objects, since they are optimized to attack a single view!



# Anything to worry about?

Later in 2017...

“Synthesizing Robust Adversarial Examples”, Athalye, Engstrom, Ilyas, Kwok, 2017

3D-printed **turtle** model classified as **rifle** from most viewpoints





# Adversarial examples

- Current deep models have bad **worst-case performance**
- Can be exploited by an adversary
- Few guarantees, can't fully trust what the model's output



# Mission-critical computer vision systems



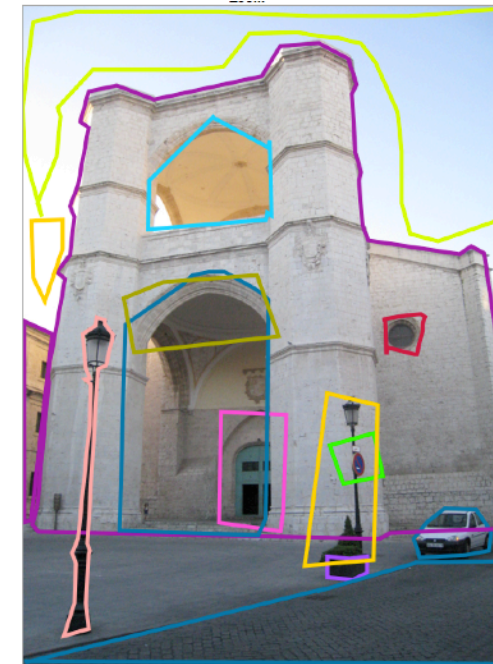


# Some things to worry about...

- Misinformation



- Our datasets are often poorly labeled



- And usually biased (overrepresent certain categories)



- ML methods perform beautifully on laboratory data, but often generalize poorly to real-world data



- Can have negative social consequences

Project office hours until 2pm