

Computational Analysis of Big Data

Week 4

Machine Learning 1

What is Machine Learning ● ○ ○

Model fitting ○ ○ ○

Model evaluation ○ ○ ○ ○

Code example ○

Canonical example

What is Machine Learning ● ○ ○

Model fitting ○ ○ ○

Model evaluation ○ ○ ○ ○

Code example ○

Canonical example



Canonical example



[]

Dog



[1 1 0 1 0 0 0]

[]

Data point

fluffy
sad looking
showing teeth
ears down
tail between legs
is retriever
growling

Canonical example



[]

Dog



fluffy
sad looking
showing teeth
ears down
tail between legs
is retriever
growling
[0]

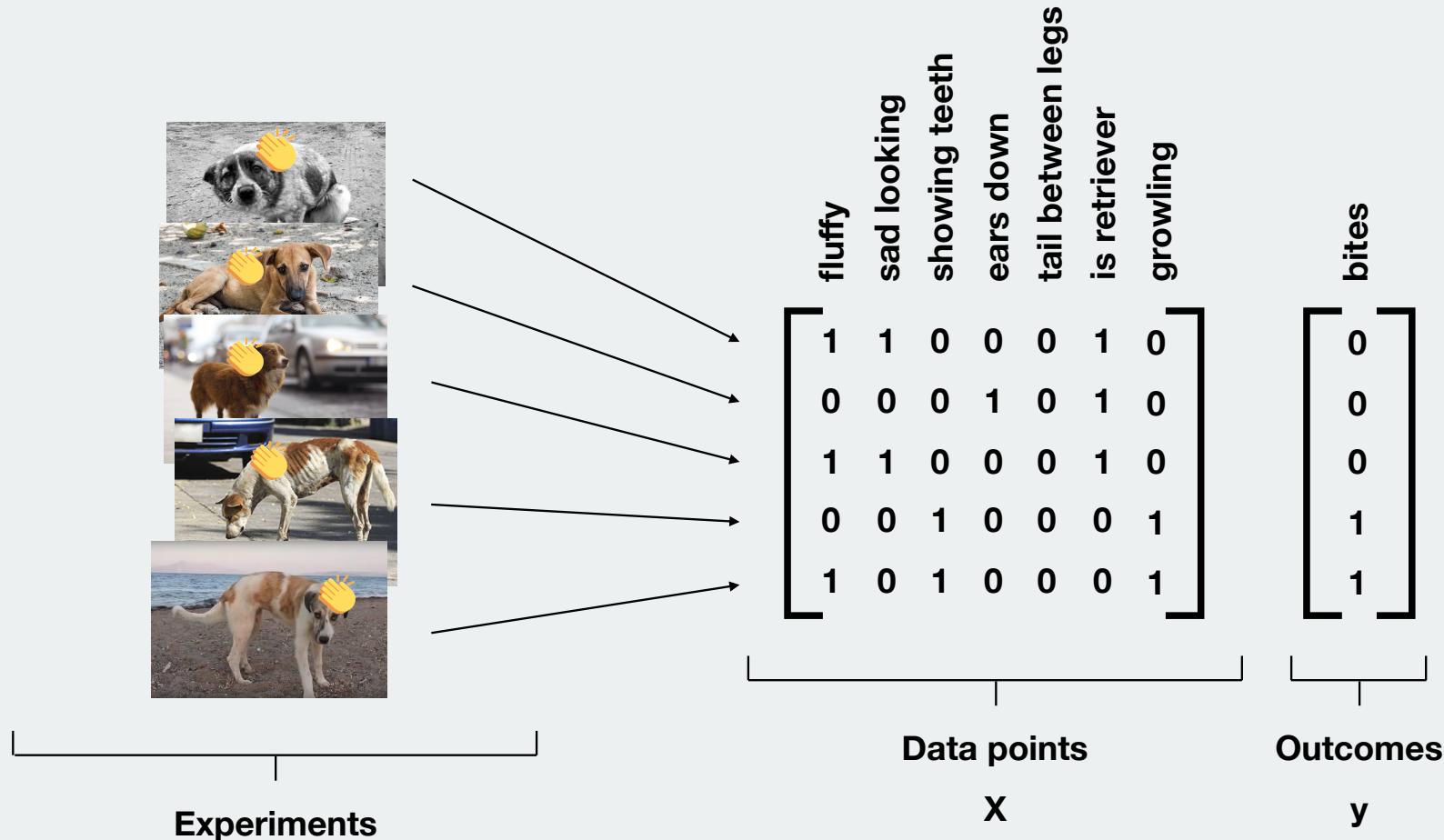
[]

Data point

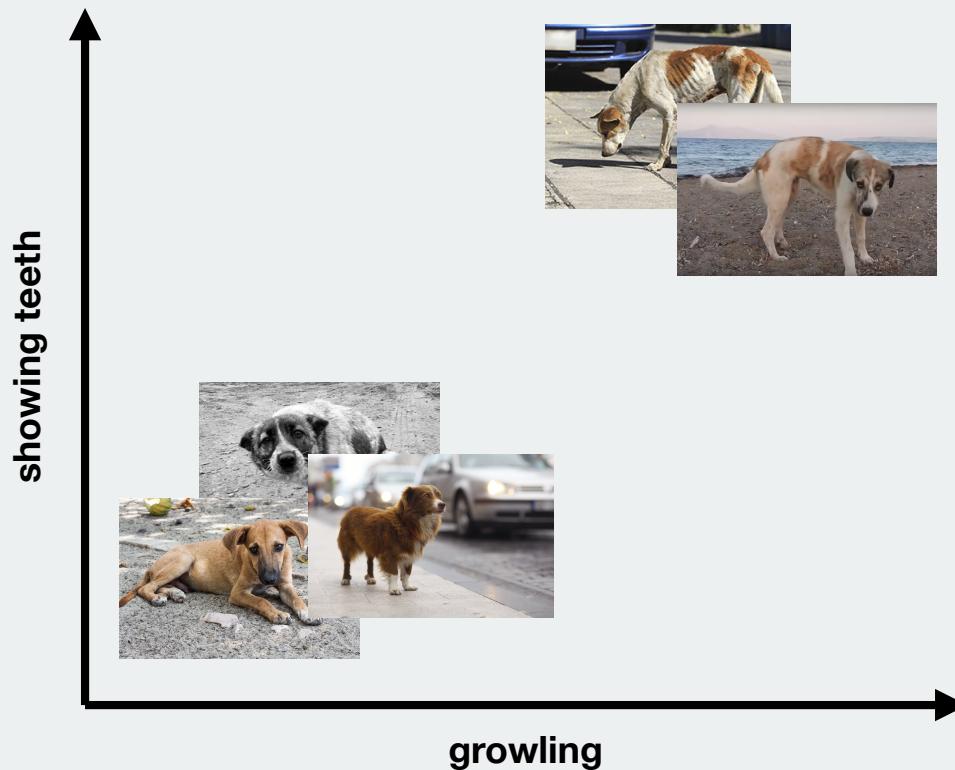
[]

Outcome

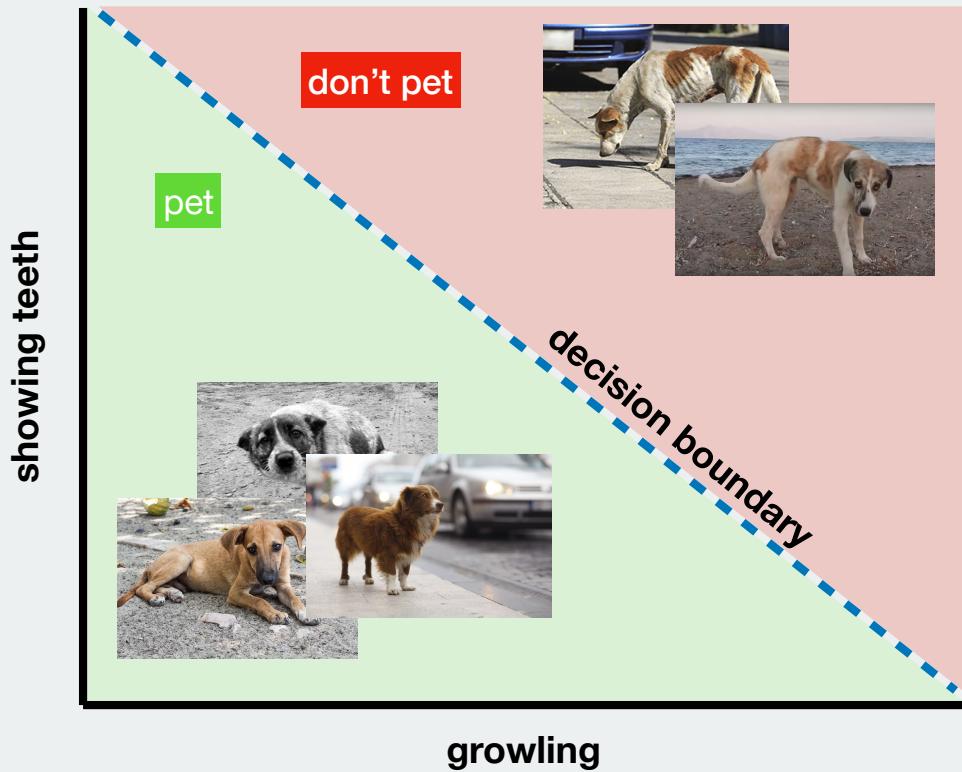
Canonical example



Canonical example



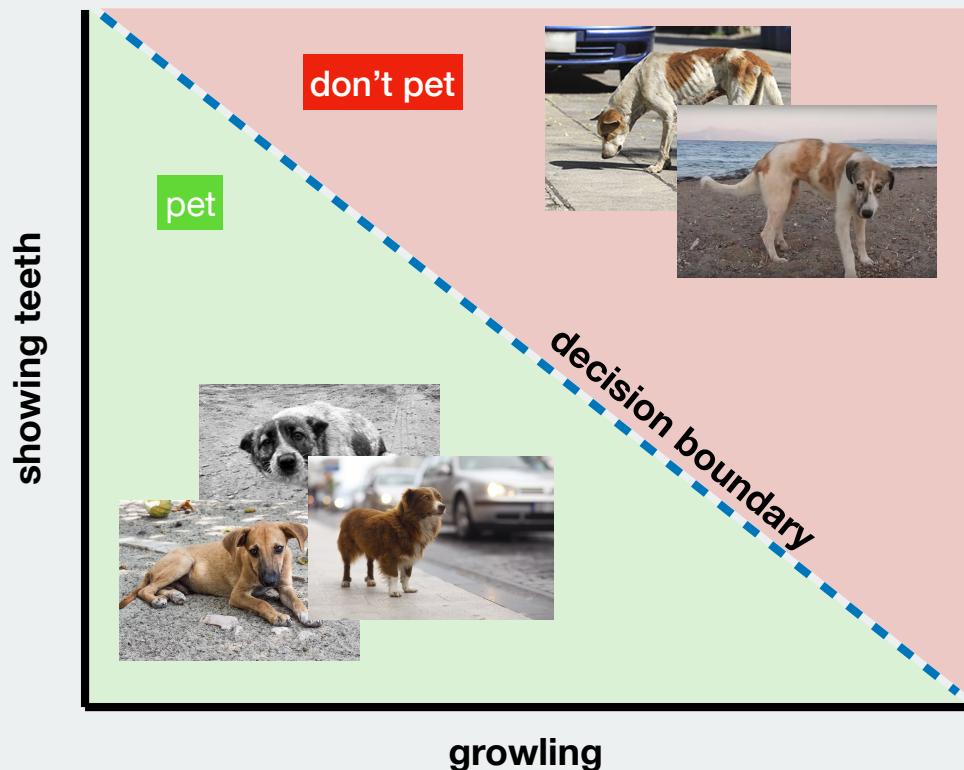
Canonical example



Canonical example

Supervised Machine Learning

When the input data has outcome labels



What is Machine Learning ● ● ○

Model fitting ○ ○ ○

Model evaluation ○ ○ ○ ○

Code example ○

Types of machine learning

Types of machine learning

Supervised

When you **have** outcomes 

Unsupervised

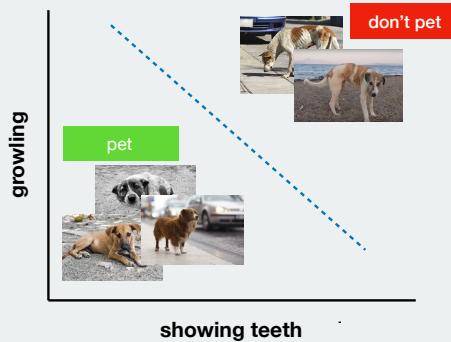
When you **don't have** outcomes 

Types of machine learning

Supervised

When you **have** outcomes 

- Classification



Unsupervised

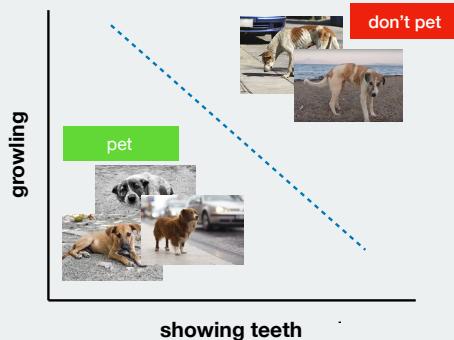
When you **don't have** outcomes 

Types of machine learning

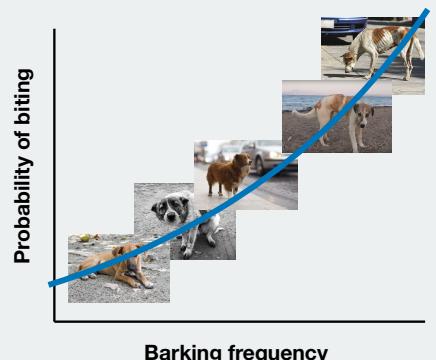
Supervised

When you **have** outcomes 🙌

- Classification



- Regression



Unsupervised

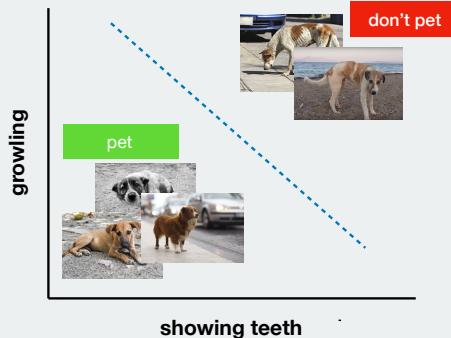
When you **don't have** outcomes 🚫

Types of machine learning

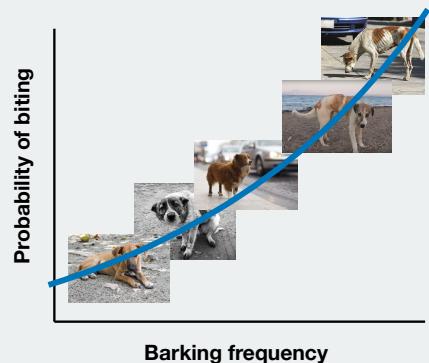
Supervised

When you **have** outcomes 🙌

- Classification



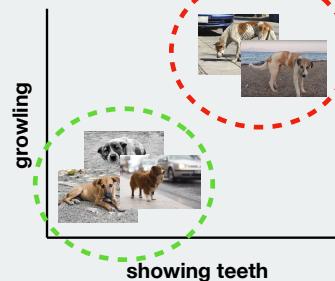
- Regression



Unsupervised

When you **don't have** outcomes 🚫

- Clustering

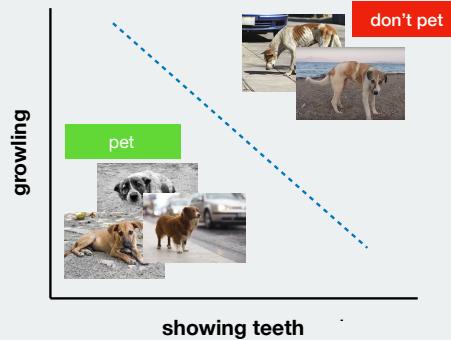


Types of machine learning

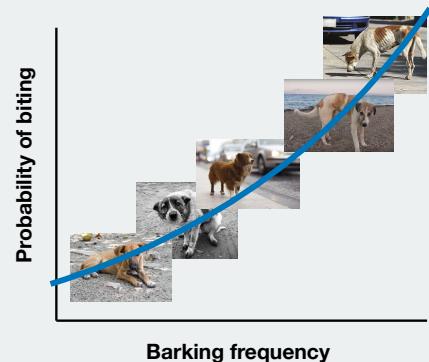
Supervised

When you **have** outcomes 🙌

- Classification



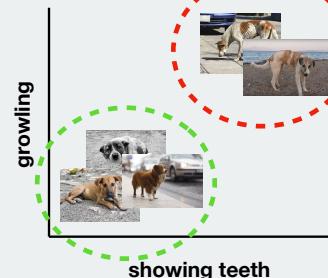
- Regression



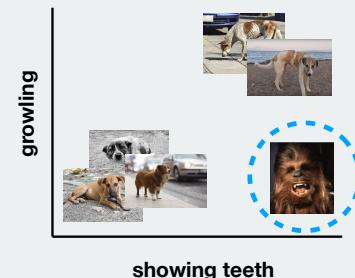
Unsupervised

When you **don't have** outcomes 🚫

- Clustering



- Outlier detection

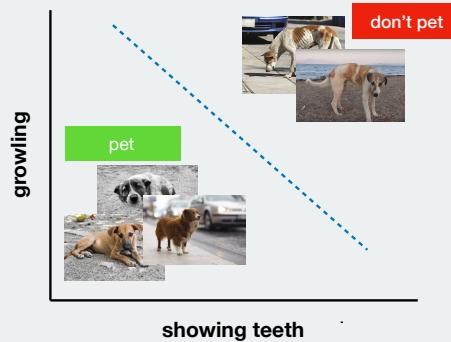


Types of machine learning

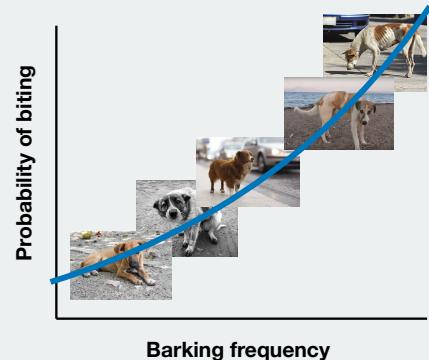
Supervised

When you **have** outcomes 🙌

- Classification



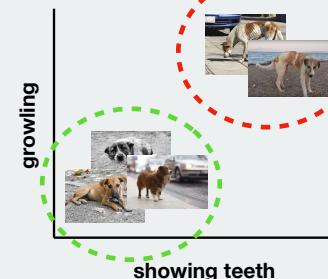
- Regression



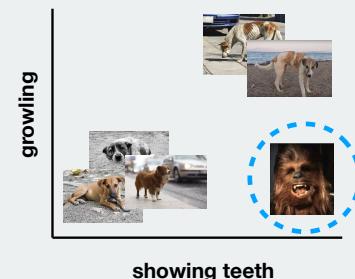
Unsupervised

When you **don't have** outcomes 🚫

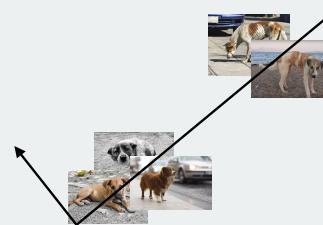
- Clustering



- Outlier detection



- Latent variable analysis

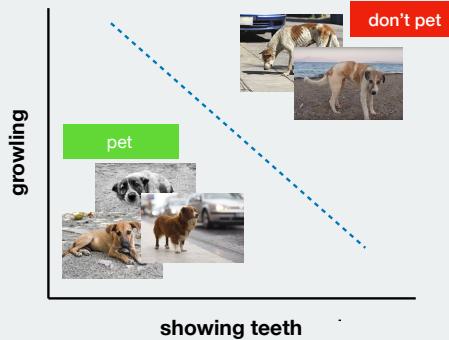


Types of machine learning

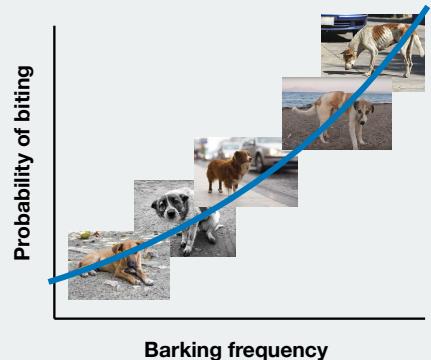
Supervised

When you **have** outcomes 🙌

- Classification



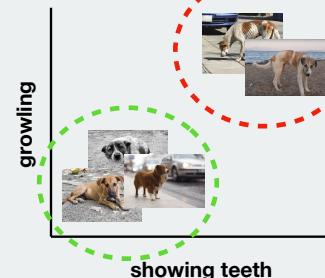
- Regression



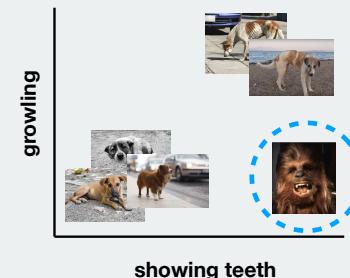
Unsupervised

When you **don't have** outcomes ✎

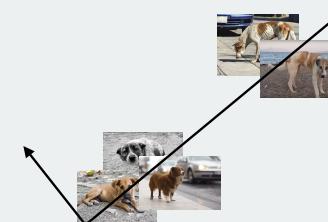
- Clustering



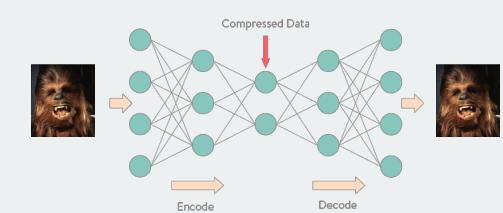
- Outlier detection



- Latent variable analysis

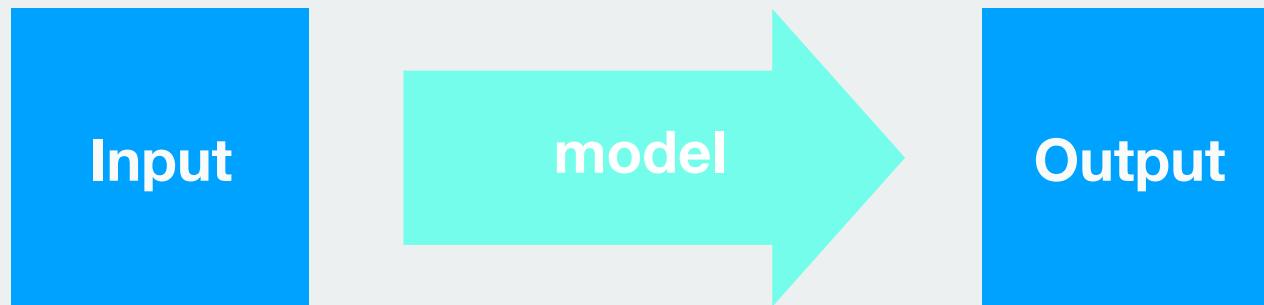


- Auto encoding



In a nutshell

You want the **model** that
best fits your **data**



What is Machine Learning ● ● ●

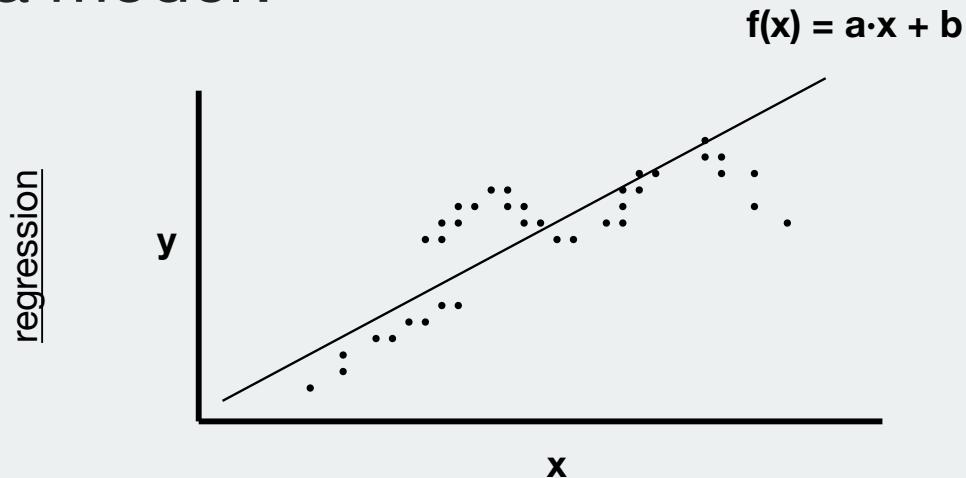
Model fitting ● ○ ○

Model evaluation ○ ○ ○ ○

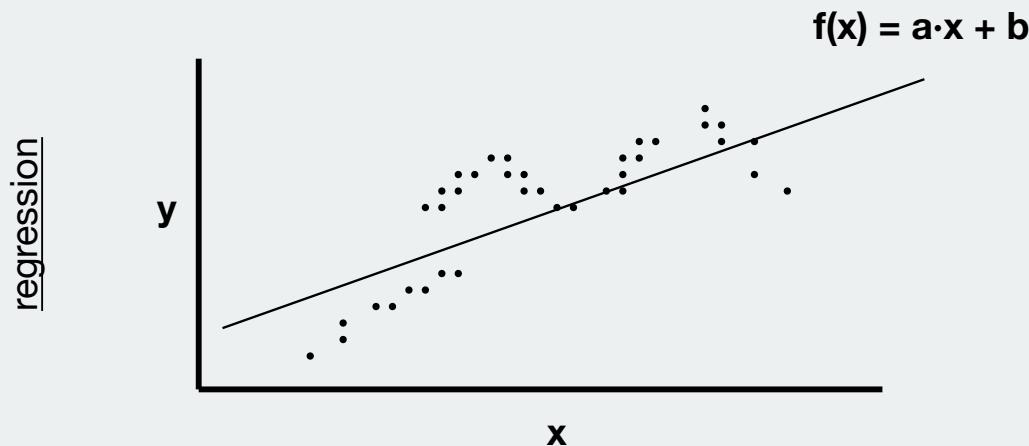
Code example ○

What is a model?

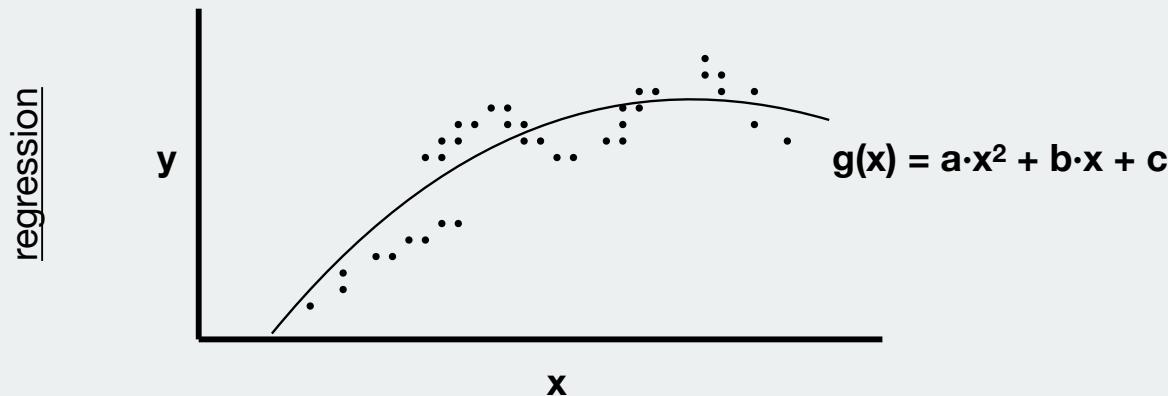
What is a model?



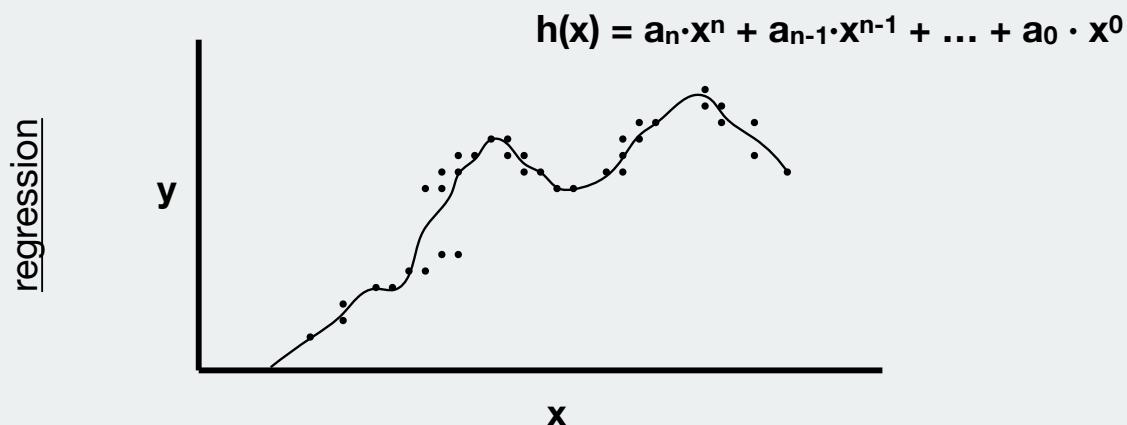
What is a model?



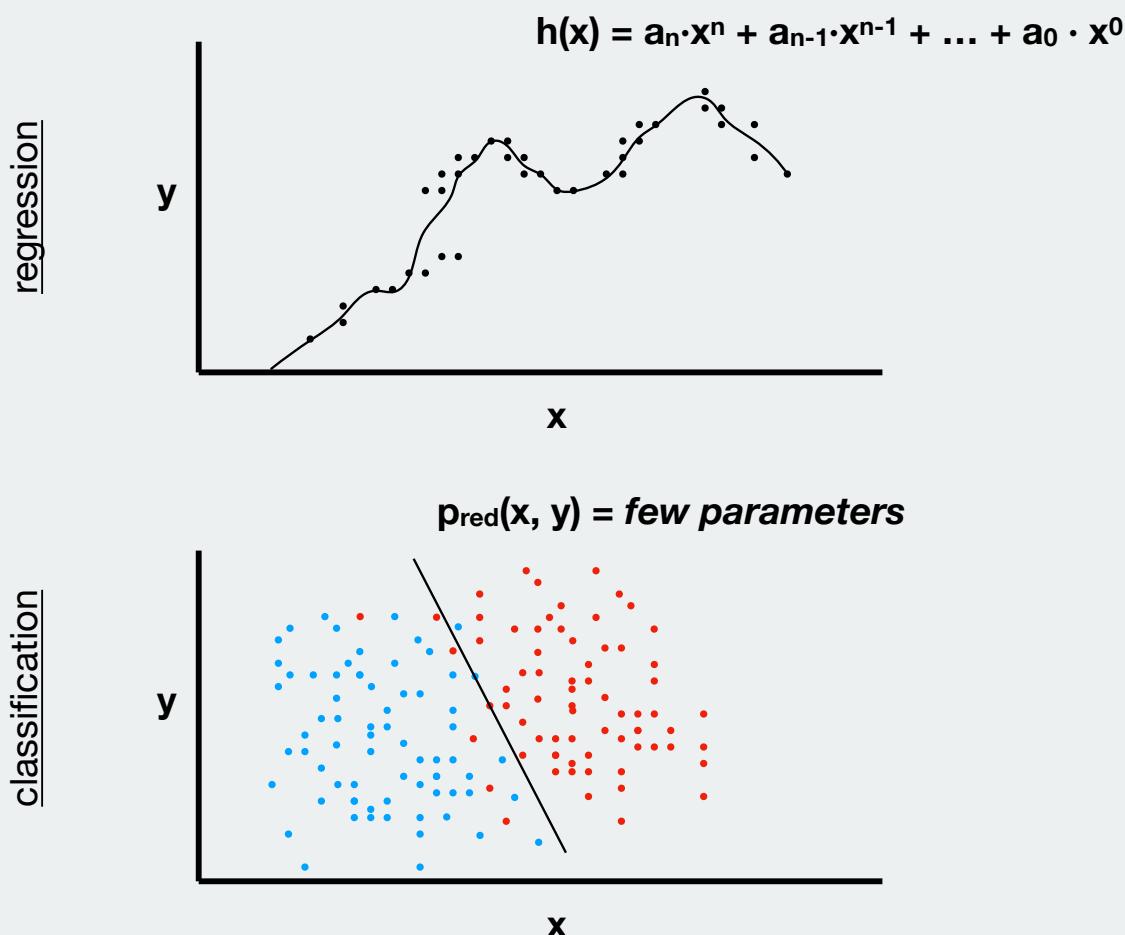
What is a model?



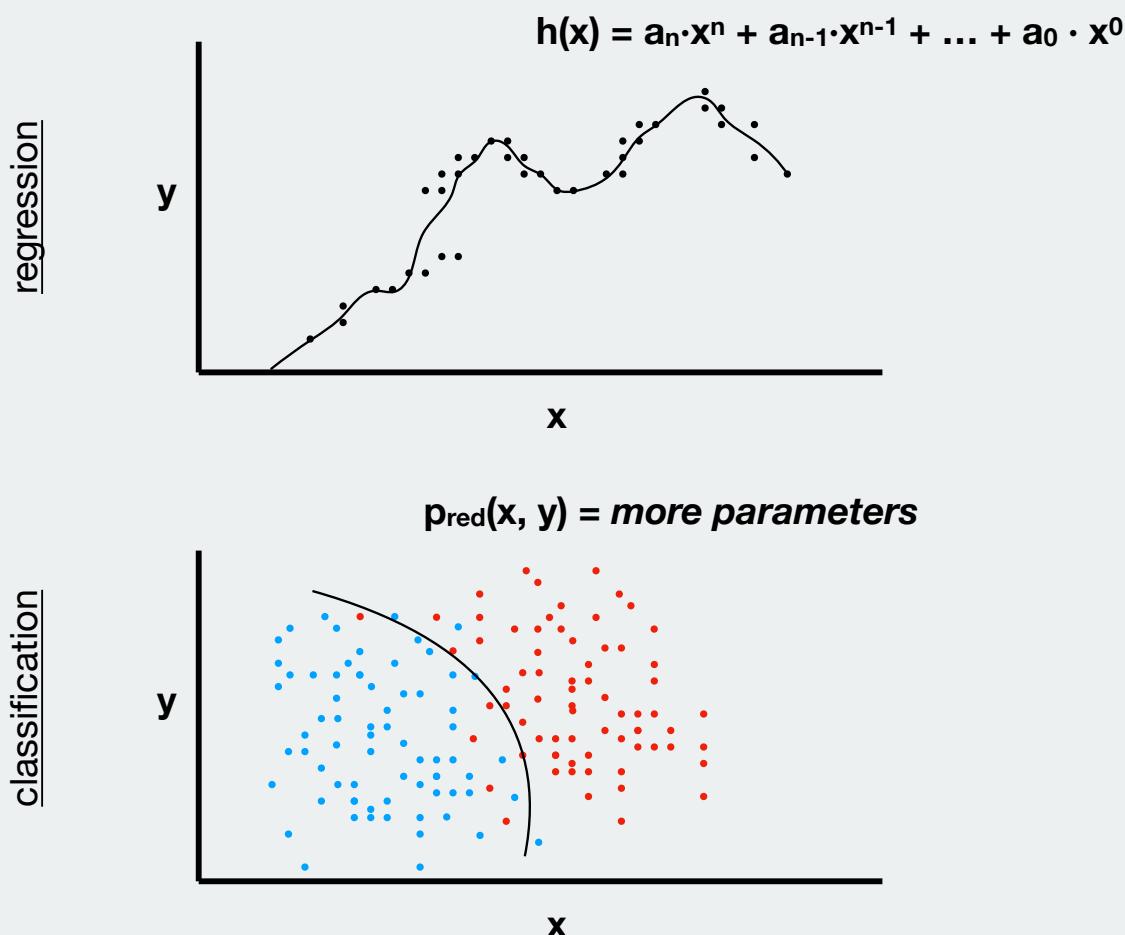
What is a model?



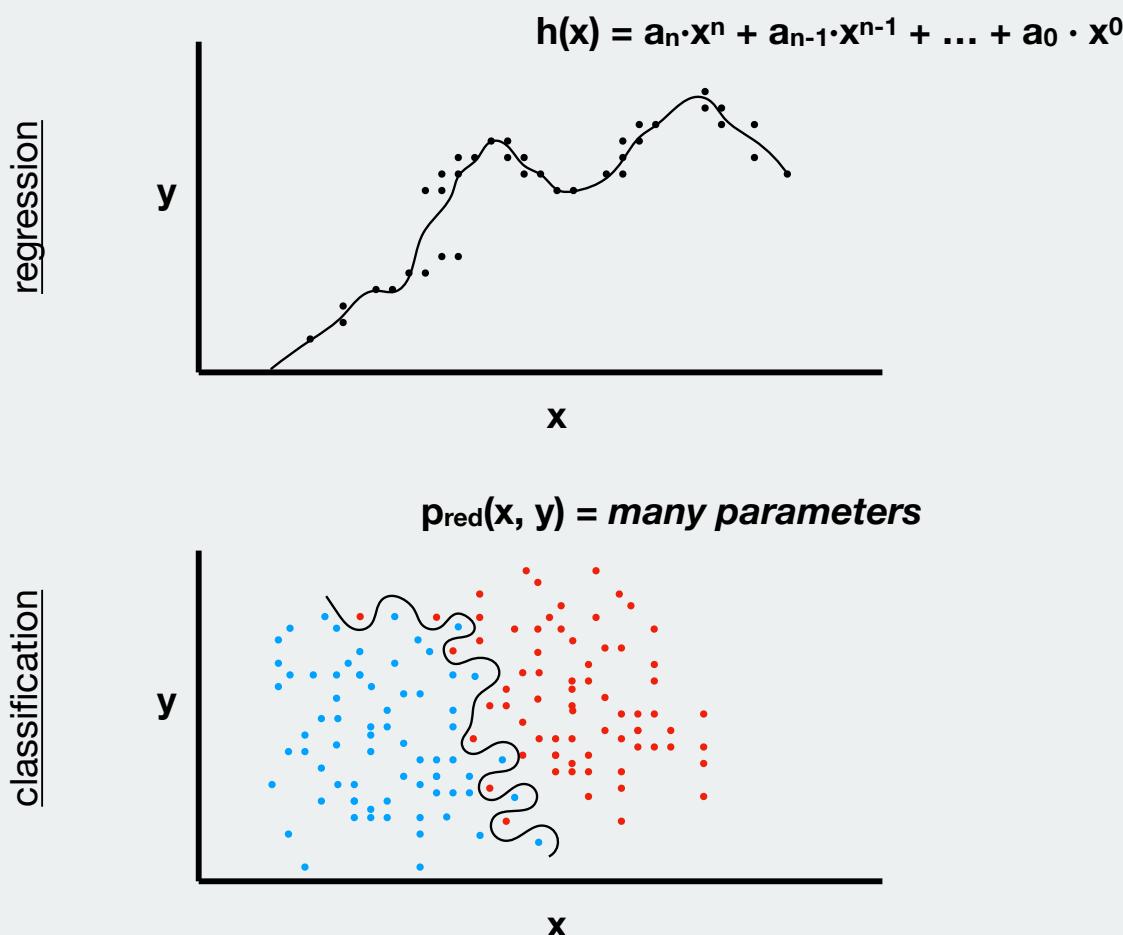
What is a model?



What is a model?



What is a model?



What is a model?



**THE BEST WAY TO
EXPLAIN OVERTFITTING**

imgflip.com

ADDTXT.COM

What is Machine Learning ● ● ●

Model fitting ● ● ○

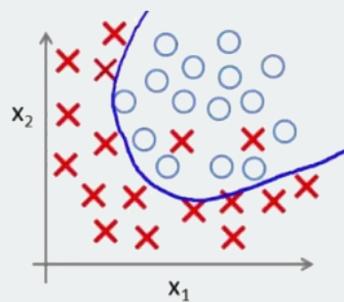
Model evaluation ○ ○ ○ ○

Code example ○

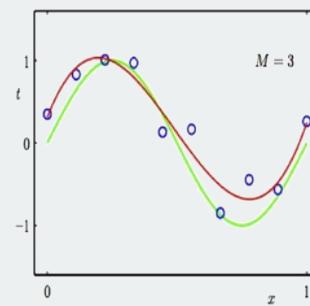
Underfitting and overfitting

Underfitting and overfitting

Classification



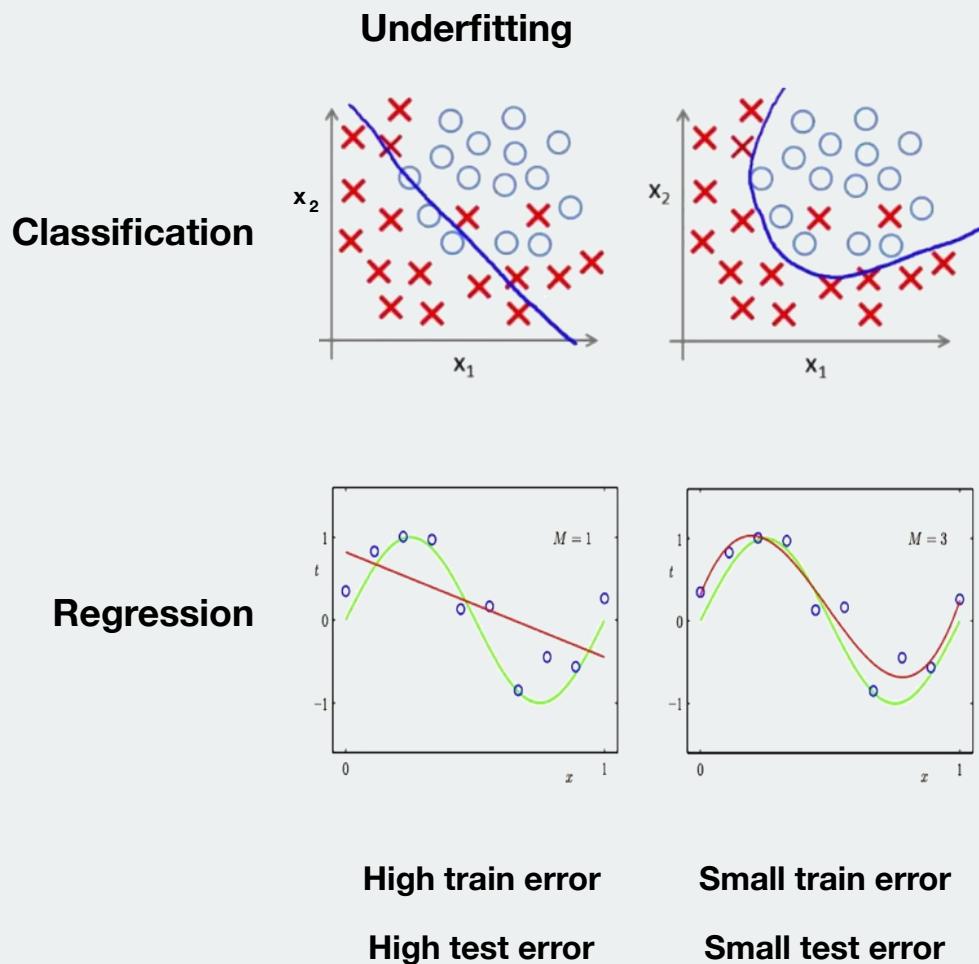
Regression



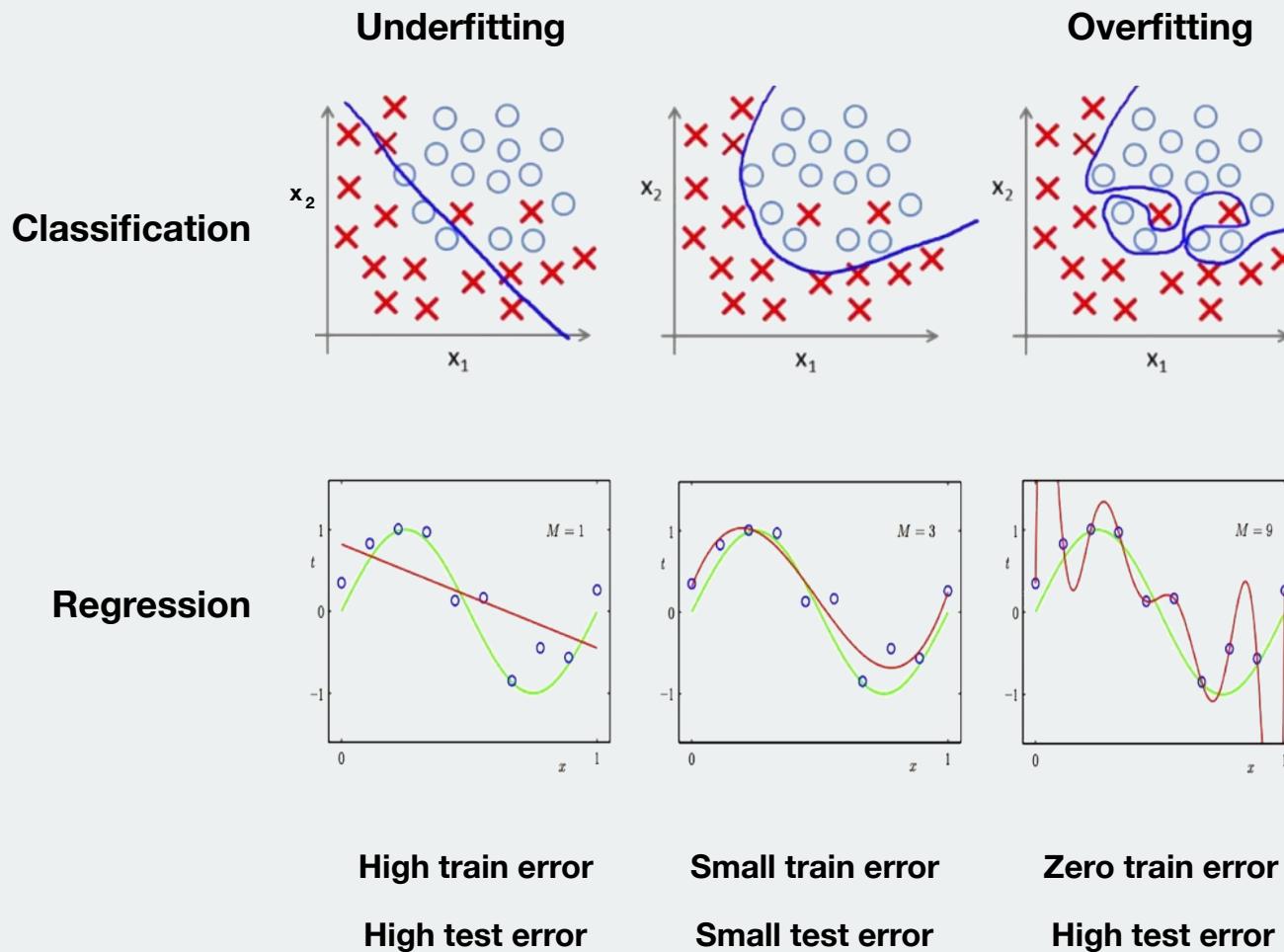
Small train error

Small test error

Underfitting and overfitting



Underfitting and overfitting



What is Machine Learning ● ● ●

Model fitting ● ● ○

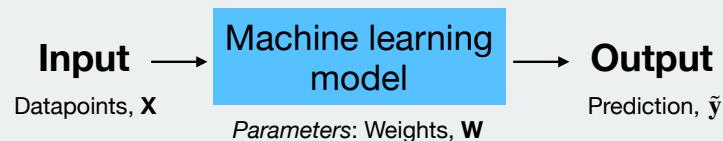
Model evaluation ○ ○ ○ ○

Code example ○

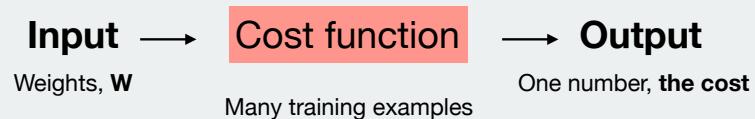
Finding model parameters

Finding model parameters

(1) The model

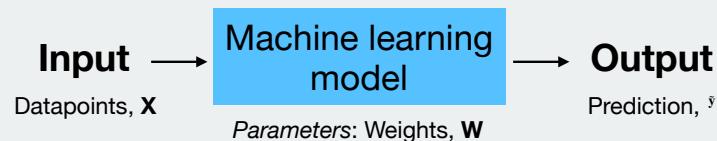


(2) Its performance



Finding model parameters

(1) The model



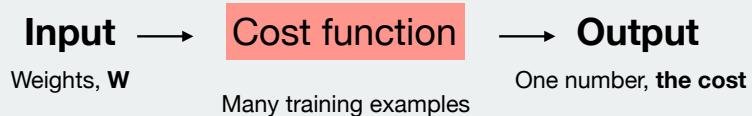
predicted from \mathbf{X} given \mathbf{W}

$$\hat{\mathbf{y}} = \begin{bmatrix} 0.96 \\ 0.10 \\ 0.04 \\ \dots \\ 0.70 \\ 0.02 \\ 0.99 \end{bmatrix}$$

true

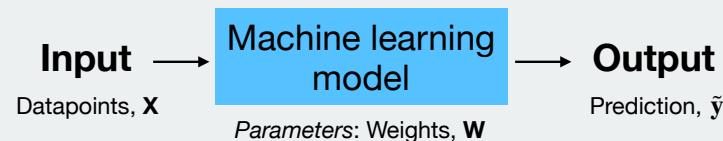
$$\mathbf{y} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \dots \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

(2) Its performance

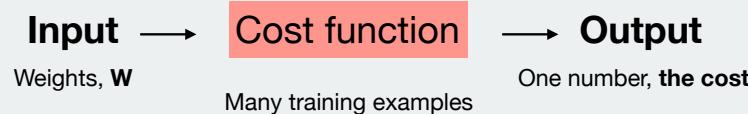


Finding model parameters

(1) The model



(2) Its performance



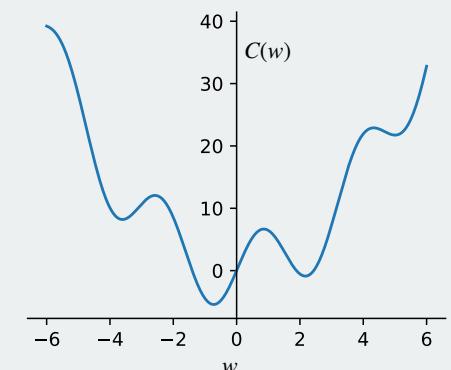
predicted from \mathbf{X} given \mathbf{W}

$$\tilde{\mathbf{y}} = \begin{bmatrix} 0.96 \\ 0.10 \\ 0.04 \\ \dots \\ 0.70 \\ 0.02 \\ 0.99 \end{bmatrix}$$

true

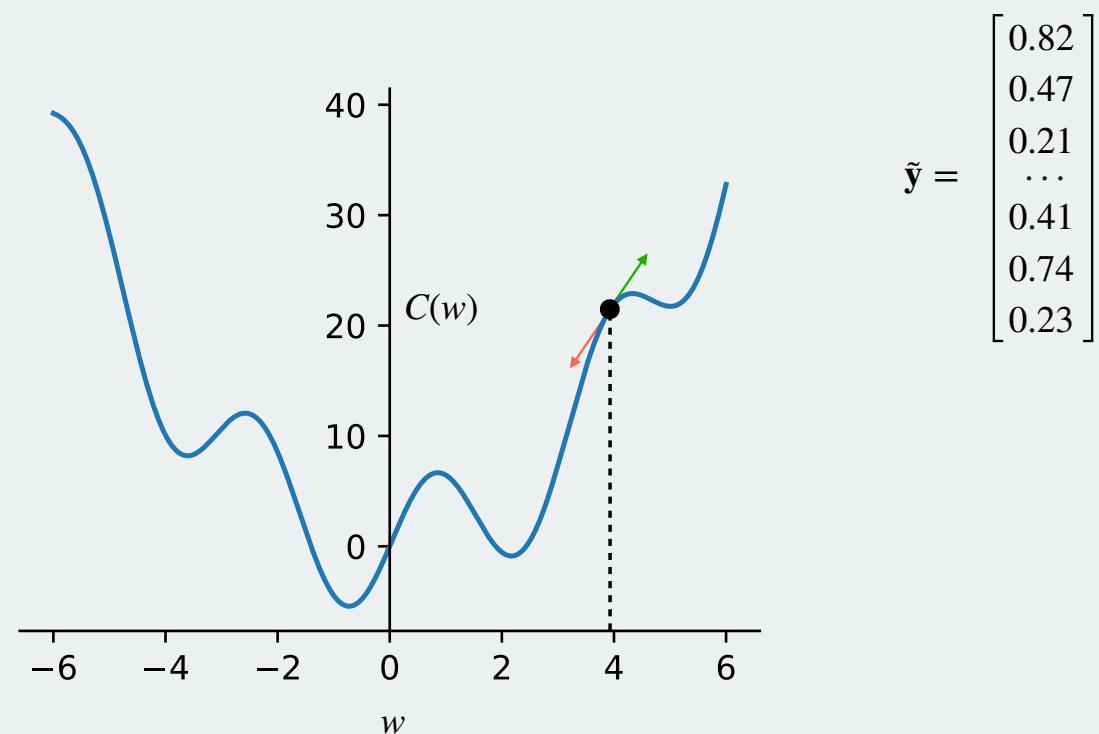
$$\mathbf{y} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \dots \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{aligned} C(\mathbf{W}) &= \frac{1}{N} \sum_i (\tilde{y}_i - y_i)^2 \\ &= (0.96 - 1)^2 \\ &\quad + (0.10 - 0)^2 \\ &\quad + (0.04 - 0)^2 \\ &\quad + \dots \\ &\quad + (0.70 - 1)^2 \\ &\quad + (0.02 - 0)^2 \\ &\quad + (0.99 - 1)^2 \end{aligned}$$



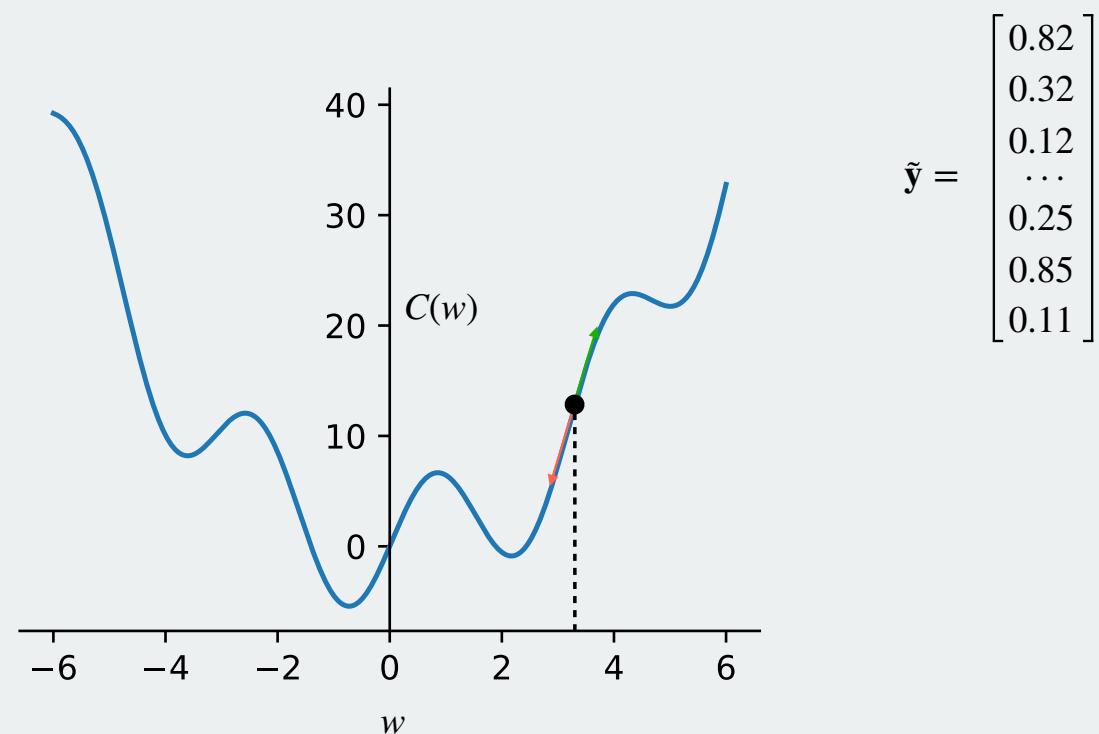
Finding model parameters

> Gradient Descent



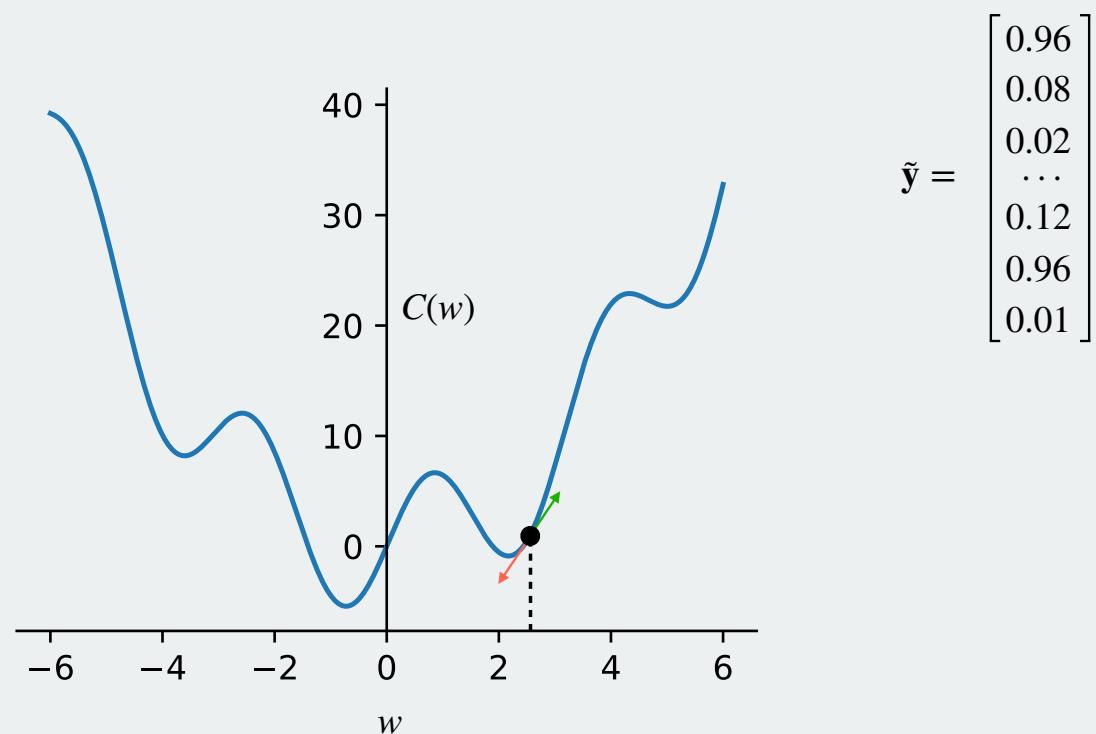
Finding model parameters

> Gradient Descent



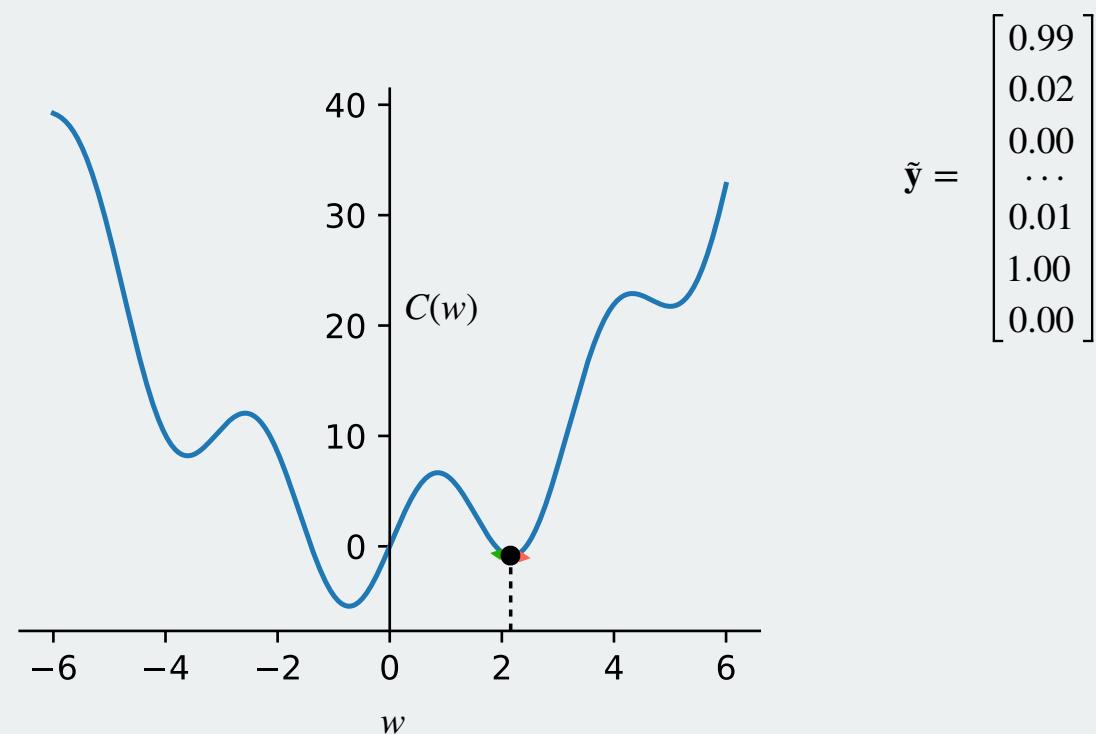
Finding model parameters

> Gradient Descent



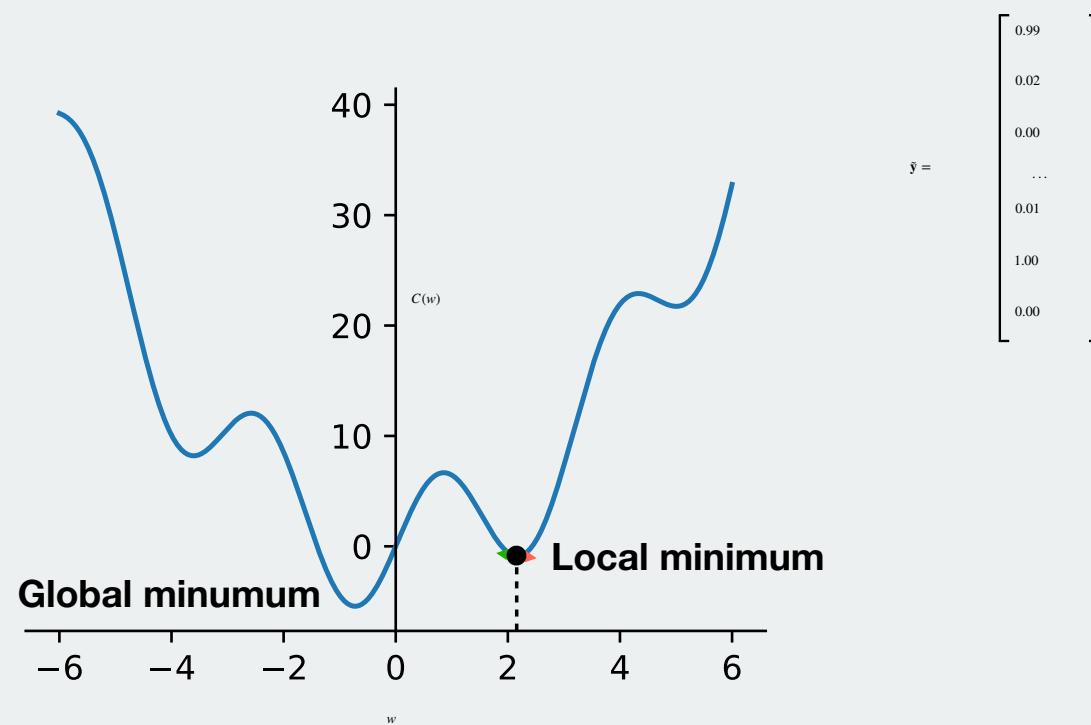
Finding model parameters

> Gradient Descent



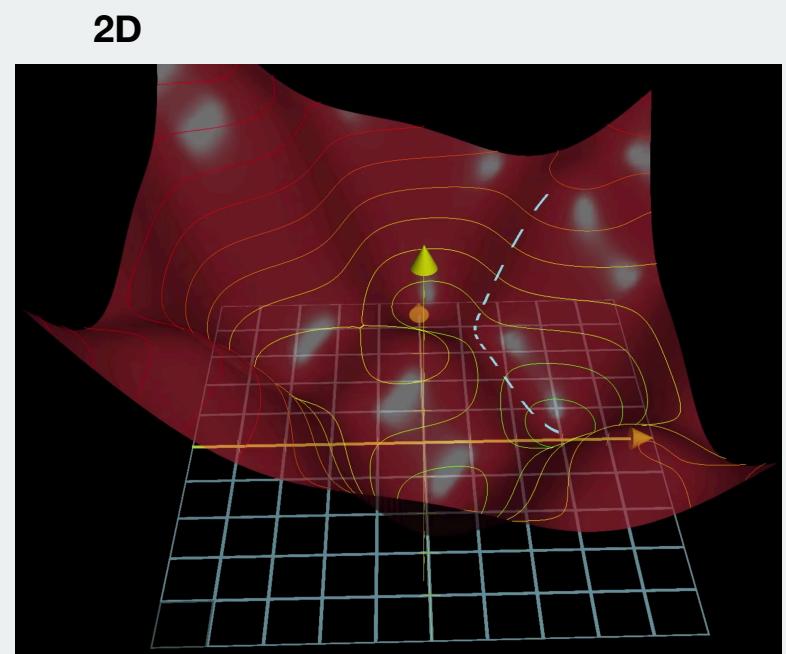
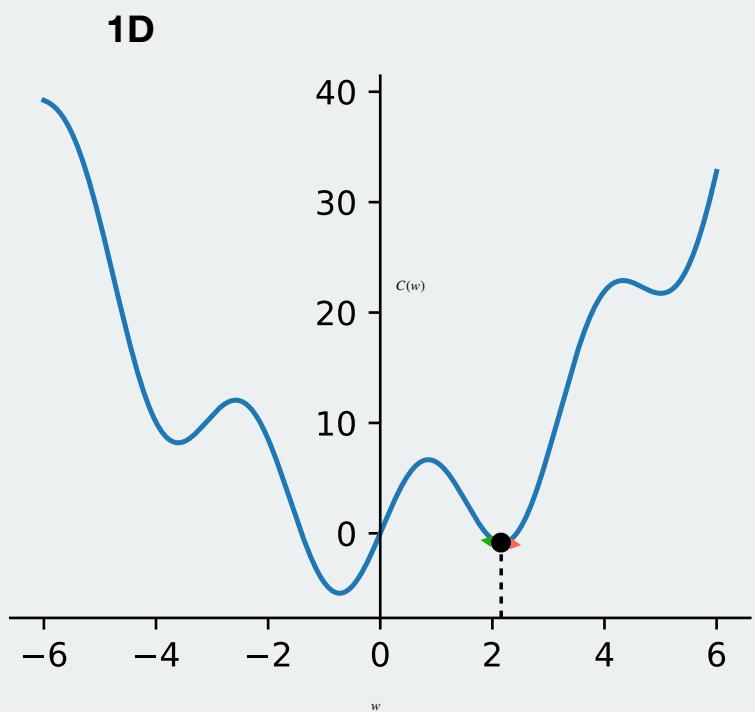
Finding model parameters

> Gradient Descent



Finding model parameters

> Gradient Descent



What is Machine Learning ● ● ●

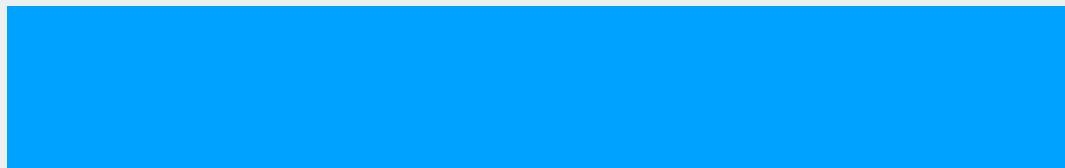
Model fitting ● ● ●

Model evaluation ● ○ ○ ○

Code example ○

Model evaluation

Model evaluation



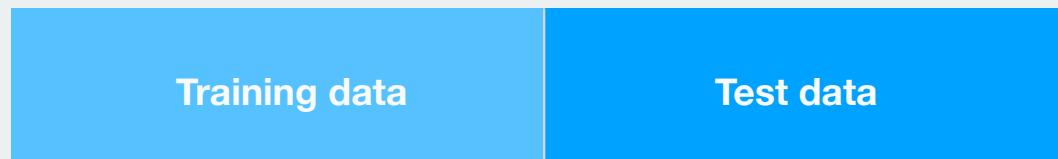
Data



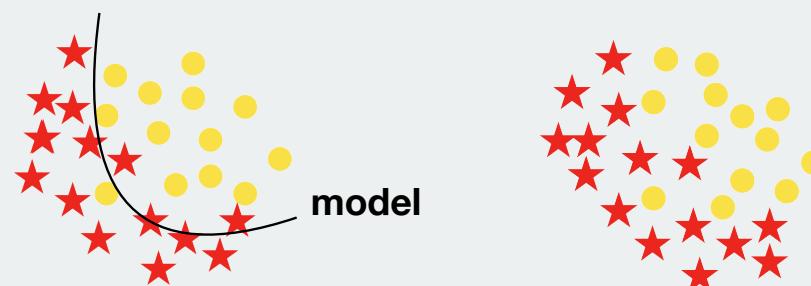
Model evaluation



Model evaluation

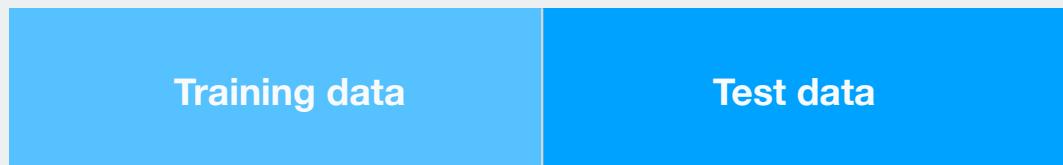


Data

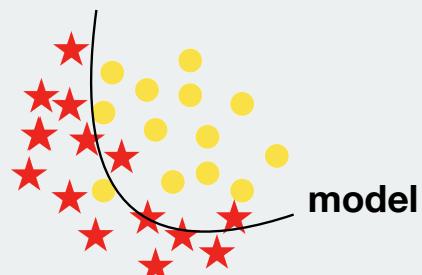


training error: $3 / 27 = \textbf{0.11}$

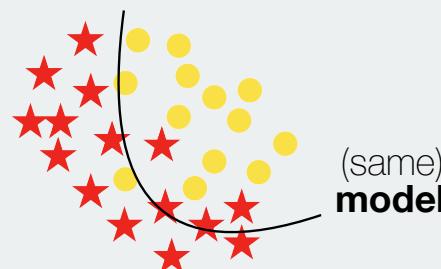
Model evaluation



Data



training error: $3 / 27 = \mathbf{0.11}$



test error: $5 / 27 = \mathbf{0.18}$

What is Machine Learning ● ● ●

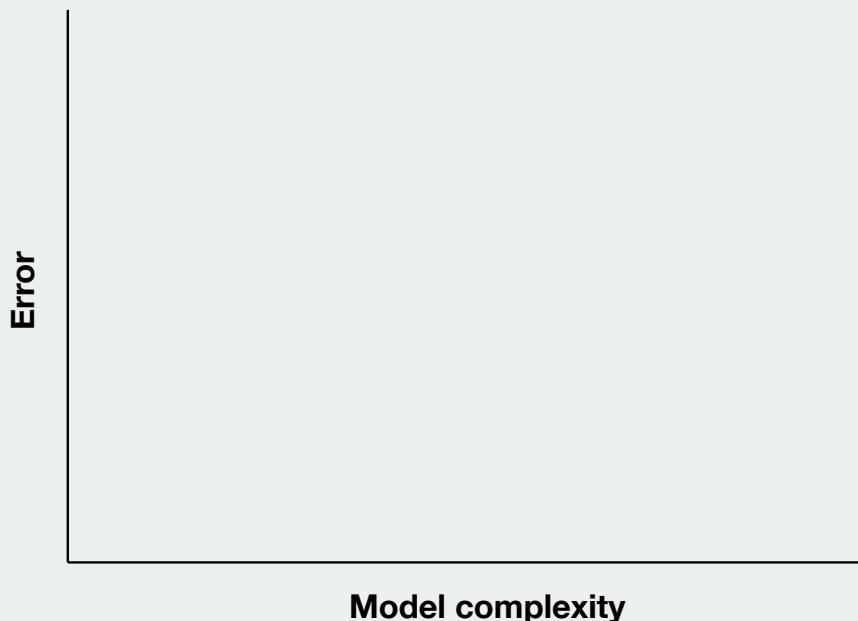
Model fitting ● ● ●

Model evaluation ● ○ ○ ○

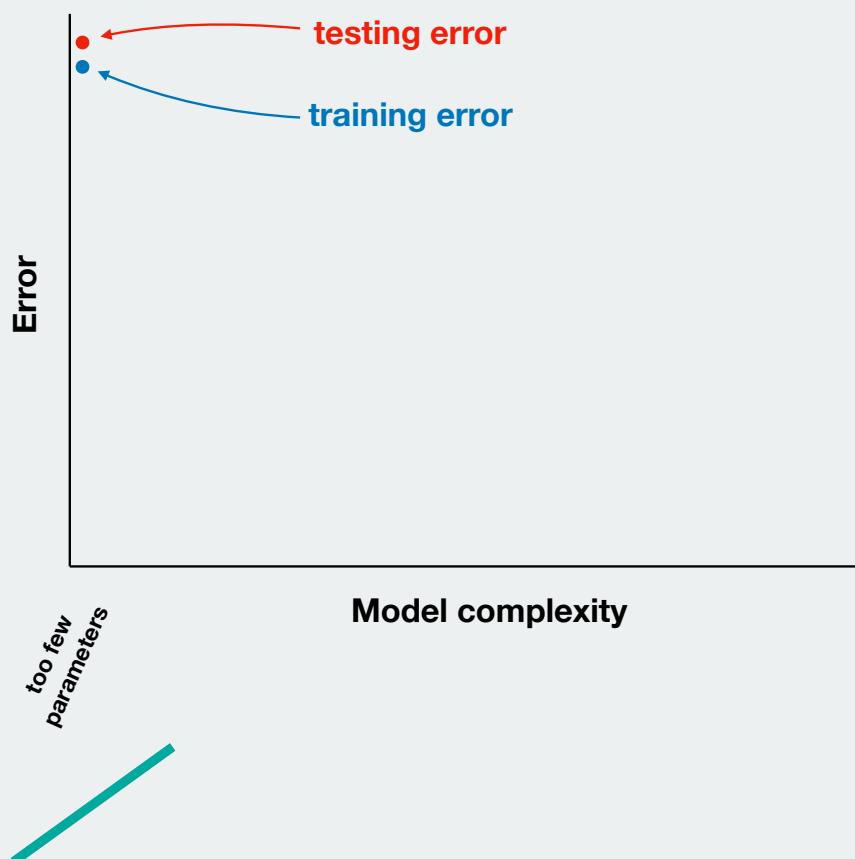
Code example ○

Avoiding underfitting and overfitting

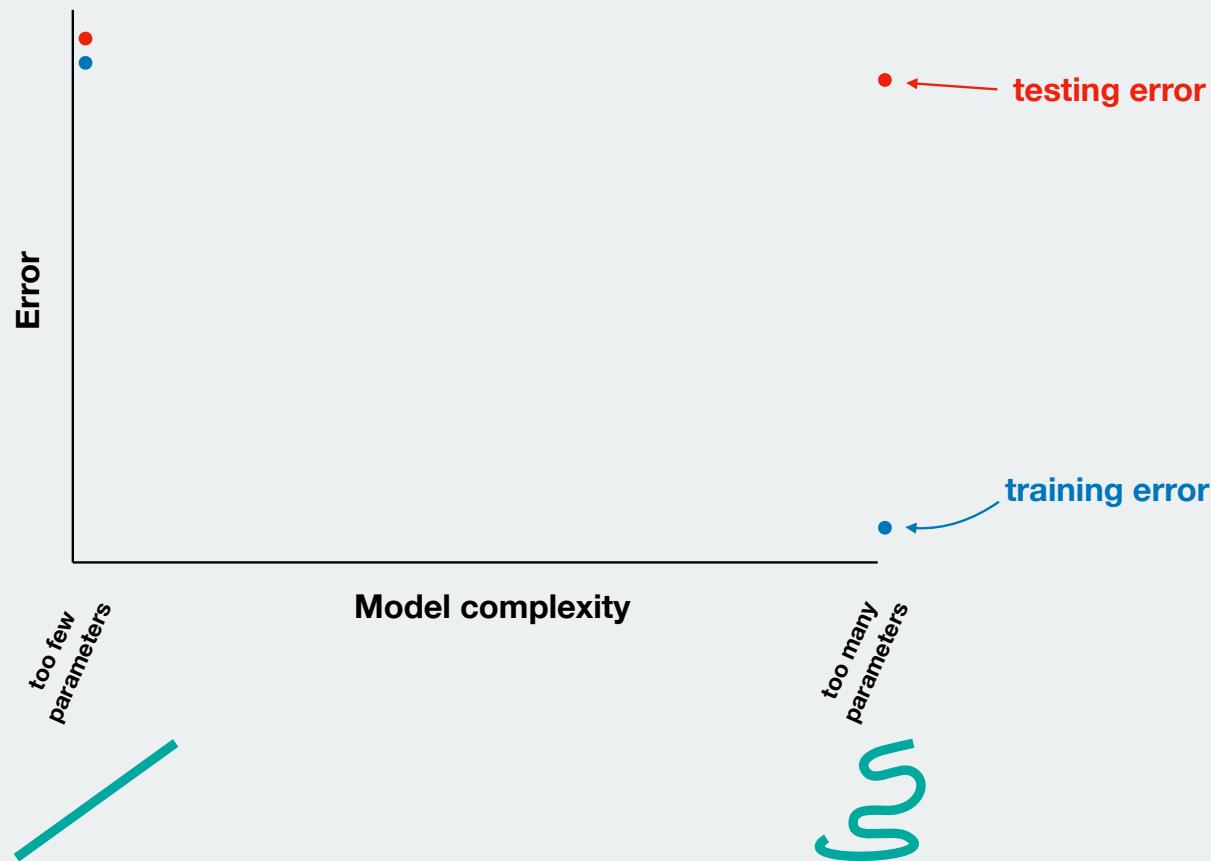
Avoiding underfitting and overfitting



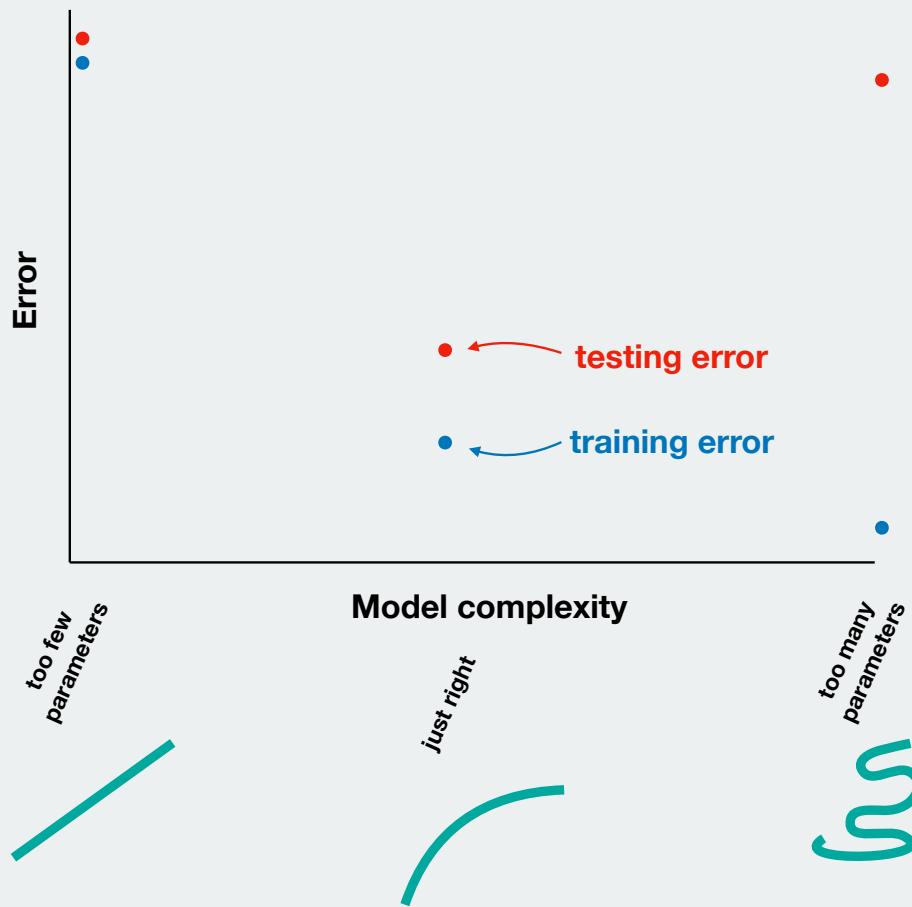
Avoiding underfitting and overfitting



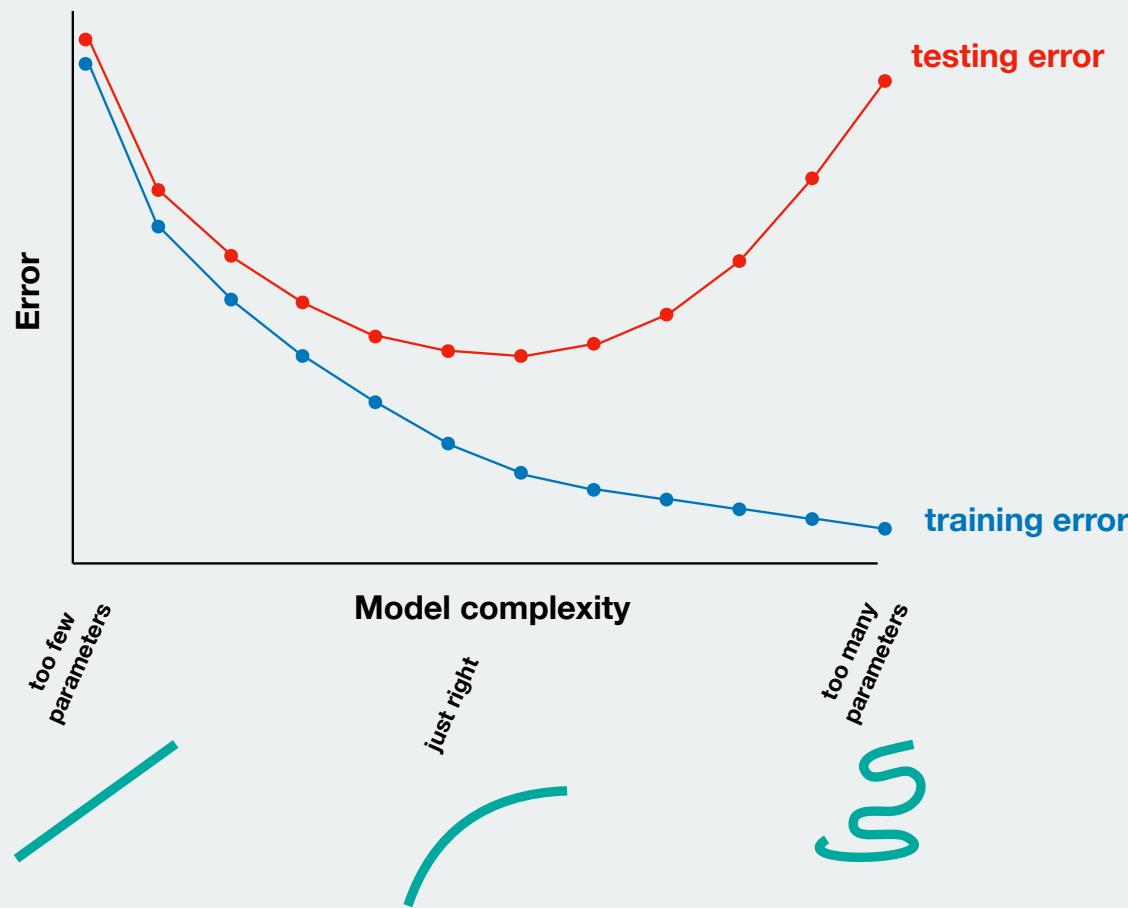
Avoiding underfitting and overfitting



Avoiding underfitting and overfitting



Avoiding underfitting and overfitting



What is Machine Learning ● ● ●

Model fitting ● ● ●

Model evaluation ● ● ○ ○

Code example ○

Cross validation

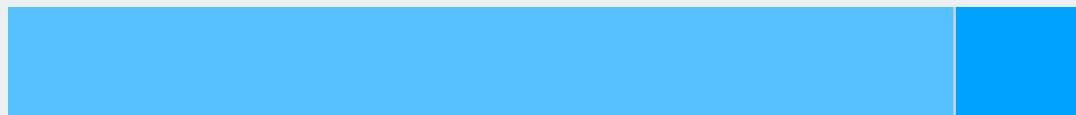
What is Machine Learning ● ● ●

Model fitting ● ● ●

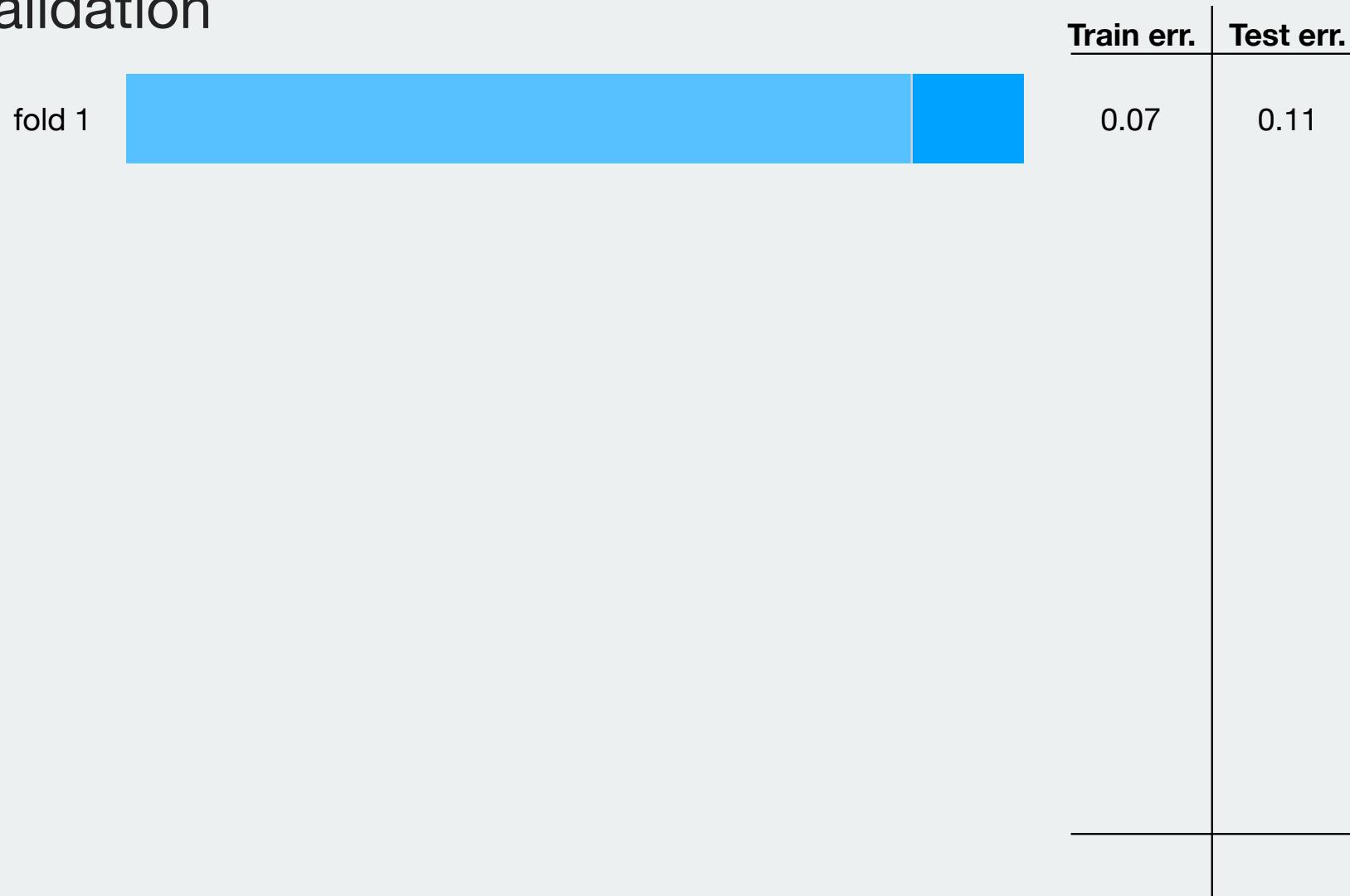
Model evaluation ● ● ○ ○

Code example ○

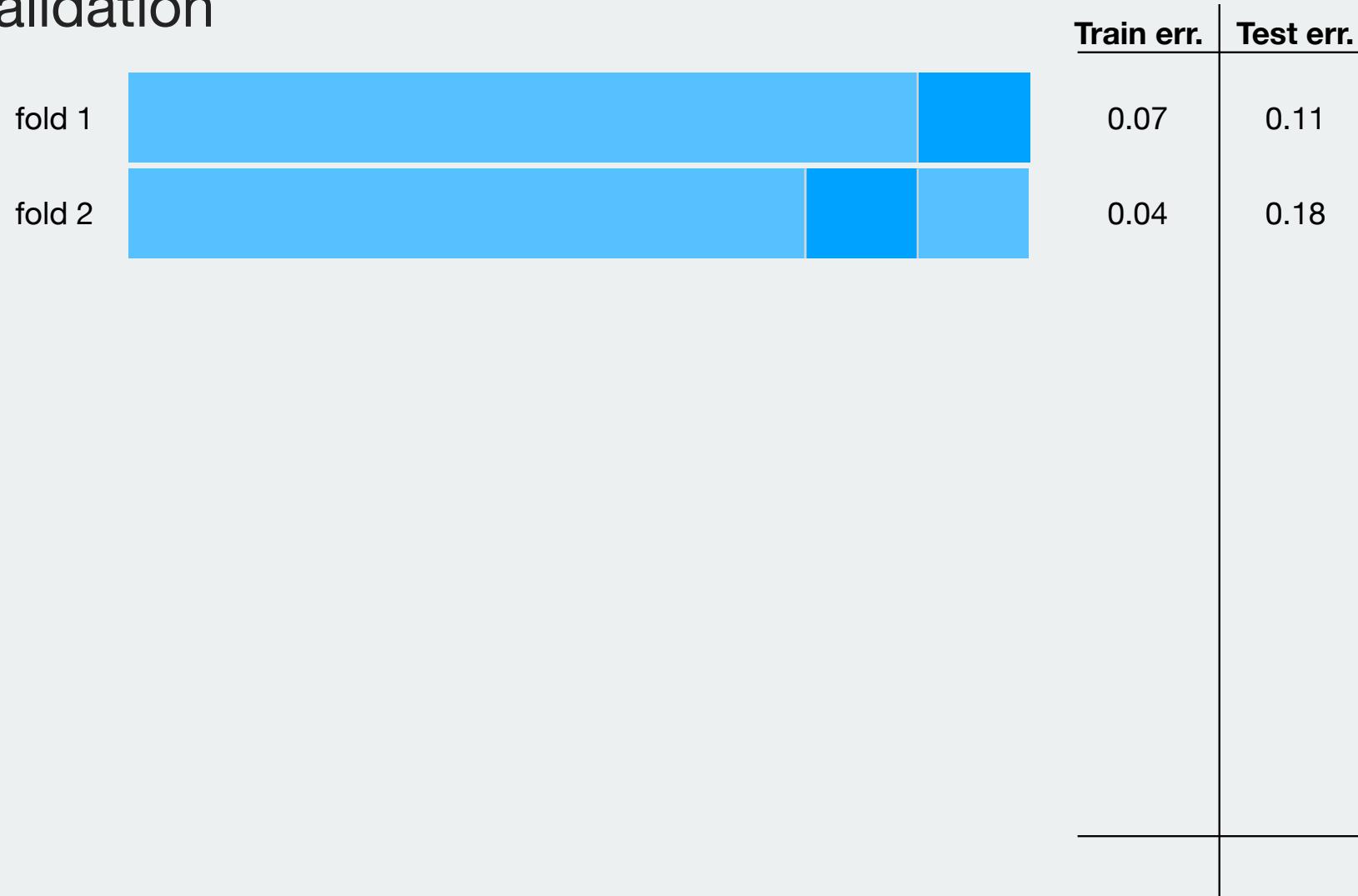
Cross validation



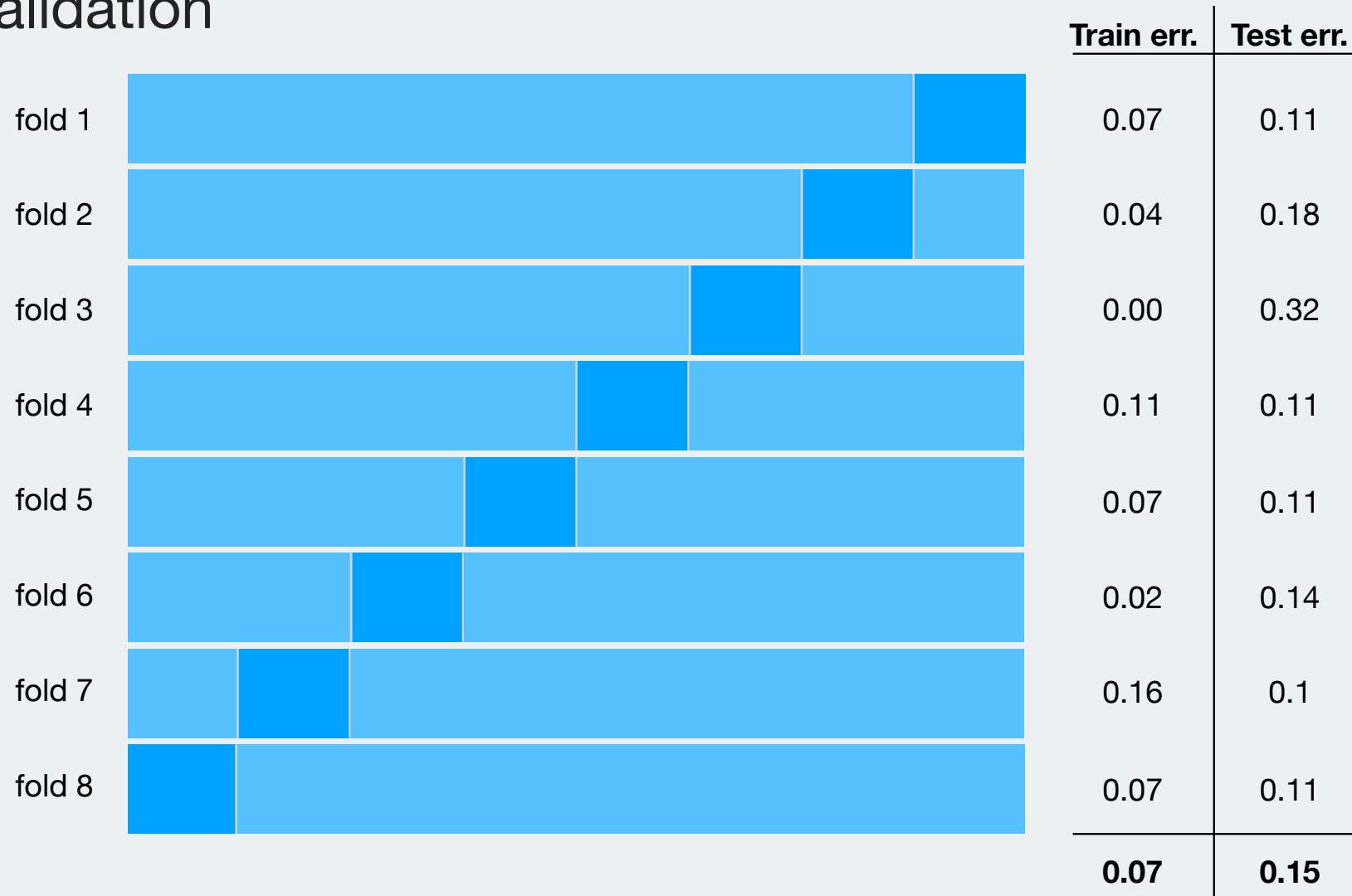
Cross validation



Cross validation



Cross validation



What is Machine Learning ● ● ●

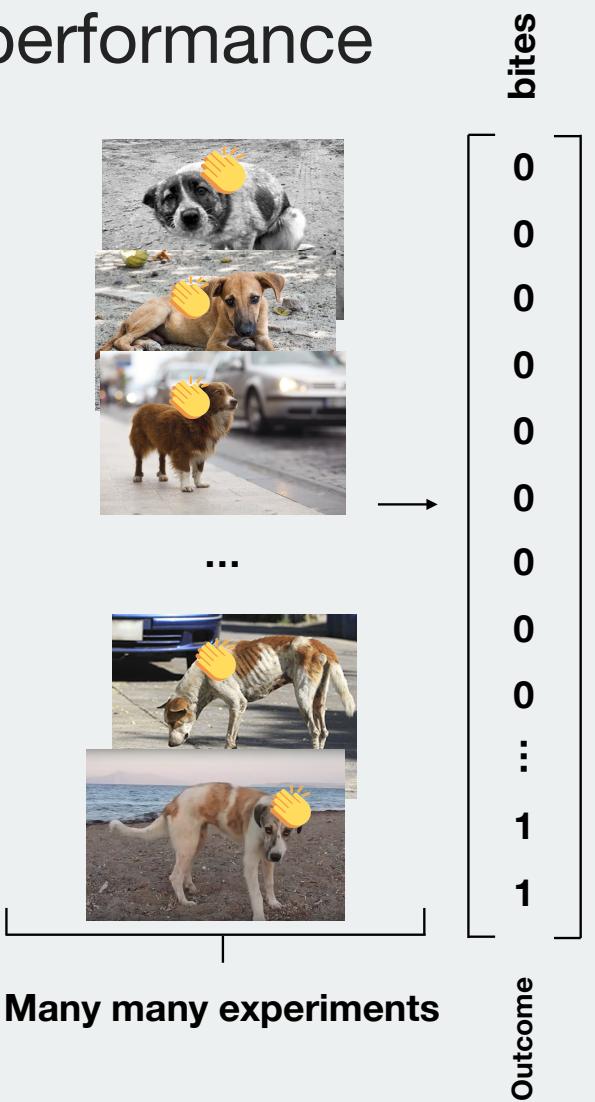
Model fitting ● ● ●

Model evaluation ● ● ● ○

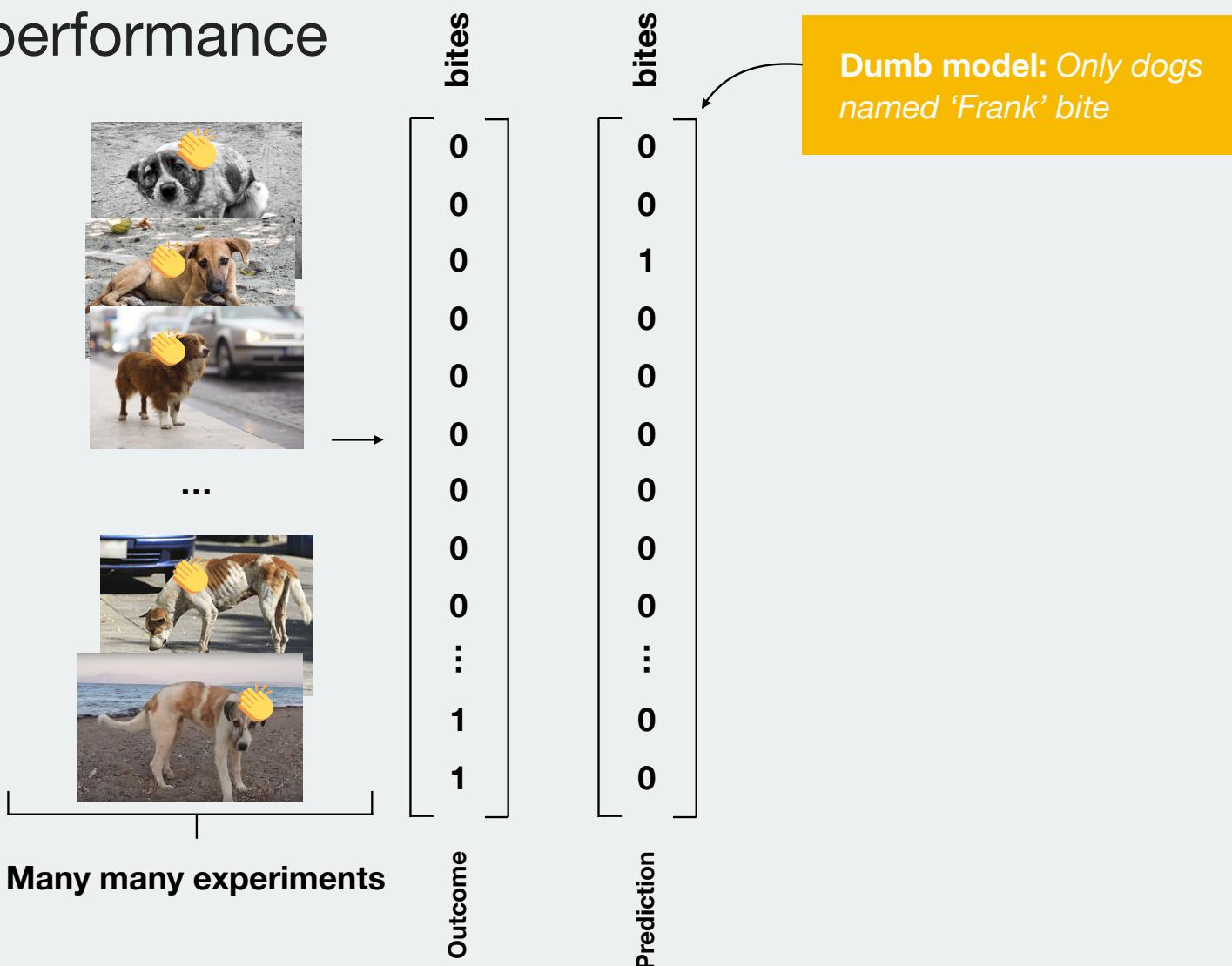
Code example ○

Model performance

Model performance



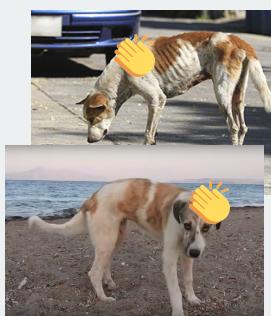
Model performance



Model performance



...



Many many experiments



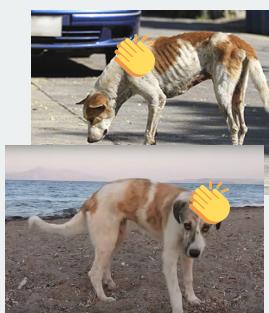
Dumb model: Only dogs named 'Frank' bite

Dogs named Frank: 1%
Dogs that bite: 2%
Frank & bites: 0.02%

Model performance



...

**Many many experiments****bites**

0
0
0
0
0
...
1
1

Outcome**bites**

0
0
1
0
0
0
0
0
0
0
0
0
0
0
...
0
0
0

Prediction

Dumb model: Only dogs named 'Frank' bite

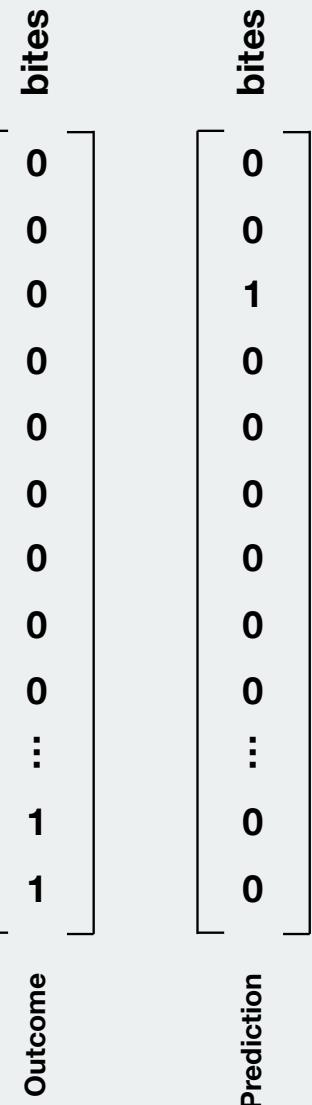
Accuracy:

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{200 + 970200}{1000000} = 97\%$$

high!

Dogs named Frank: 1%
Dogs that bite: 2%
Frank & bites: 0.02%

Model performance



Dumb model: Only dogs named 'Frank' bite

Accuracy:

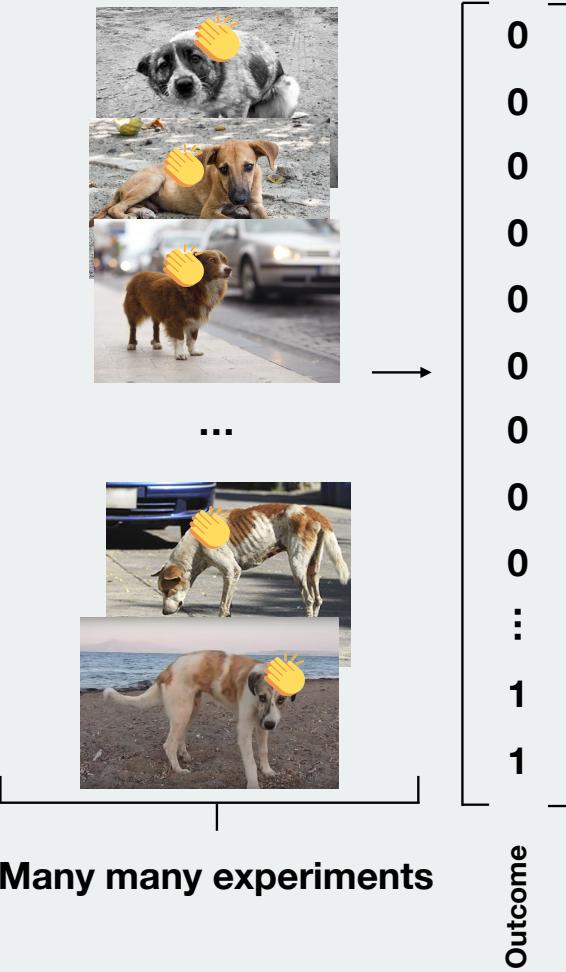
$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{200 + 970200}{1000000} = 97\%$$

high!

Confusion matrix:

		Outcome	
		Bite	No bite
Prediction	Bite	TP: 200	FP: 9800
	No bite	FN: 19800	TN: 970200

Model performance



Dumb model: Only dogs named 'Frank' bite

Dogs named Frank: 1%
Dogs that bite: 2%
Frank & bites: 0.02%

Accuracy:

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{200 + 970200}{1000000} = 97\%$$

high!

Confusion matrix:

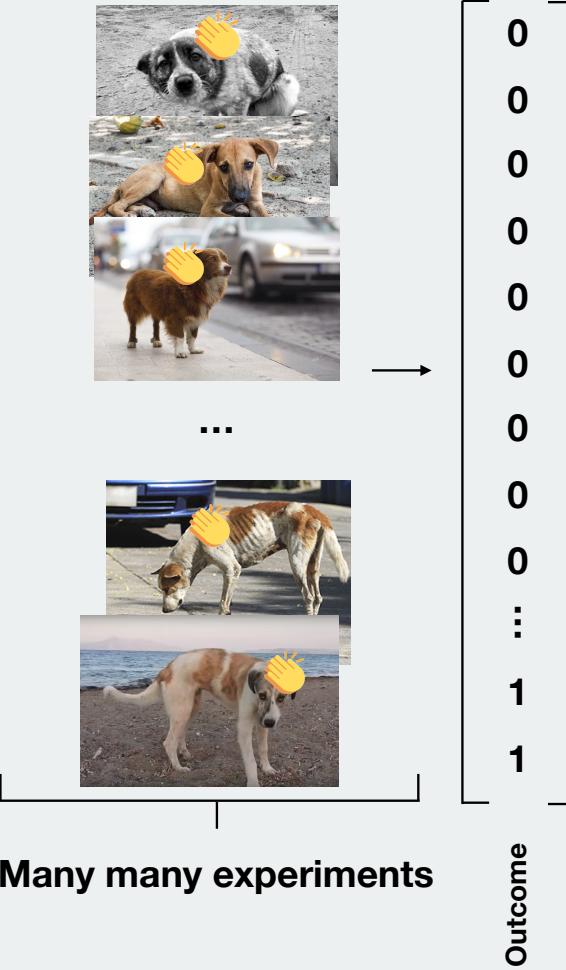
		Outcome	
		Bite	No bite
Prediction	Bite	TP: 200	FP: 9800
	No bite	FN: 19800	TN: 970200

Precision & recall

$$\text{precision} = \frac{TP}{TP + FP} = \frac{200}{200 + 9800} = 2\%$$

$$\text{recall} = \frac{TP}{TP + FN} = \frac{200}{200 + 19800} = 1\%$$

Model performance



Dumb model: Only dogs named 'Frank' bite

Dogs named Frank: 1%
Dogs that bite: 2%
Frank & bites: 0.02%

Accuracy:

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{200 + 970200}{1000000} = 97\%$$

high!

Confusion matrix:

		Outcome	
		Bite	No bite
Prediction	Bite	TP: 200	FP: 9800
	No bite	FN: 19800	TN: 970200

Precision & recall

$$\text{precision} = \frac{TP}{TP + FP} = \frac{200}{200 + 9800} = 2\%$$

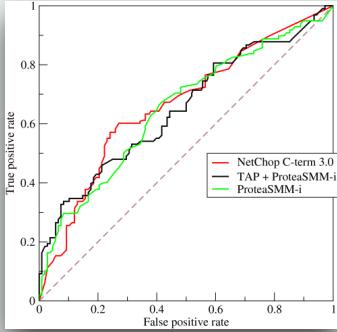
$$\text{recall} = \frac{TP}{TP + FN} = \frac{200}{200 + 19800} = 1\%$$

F1-score (harmonic mean of precision and recall):

$$F1 = \frac{2pr}{p+r} = \frac{2 \cdot 0.02 \cdot 0.01}{0.02 + 0.01} = .013$$

low!

Performance metrics summary

Metric	Description	Note
Accuracy	$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$ <p>“Fraction of classifications that are correct”</p>	Bad when classes are not balanced. Should always be compared with a balance baseline. Pretty easy to understand
Precision	$\text{precision} = \frac{TP}{TP + FP}$ <p>“What fraction of the dogs accused of biting, would actually bite?”</p>	Reveals a single useful aspect of a models performance
Recall	$\text{recall} = \frac{TP}{TP + FN}$ <p>“What fraction of dogs that would bite, were accused of biting?”</p>	Reveals a single useful aspect of a models performance
F1 score	$F1 = \frac{2pr}{p + r}$ <p>“Harmonic average of precision and recall”</p>	Mathematical construct = no intuitive interpretation “Honest” measure of performance
Receiver operating characteristic (ROC)	 <p>“Detection rate vs. false alarm rate for varying decision thresholds”</p>	Visual performance metric that gives an good impression of performance. If area under curve is > 0.5, classifier is better than random.