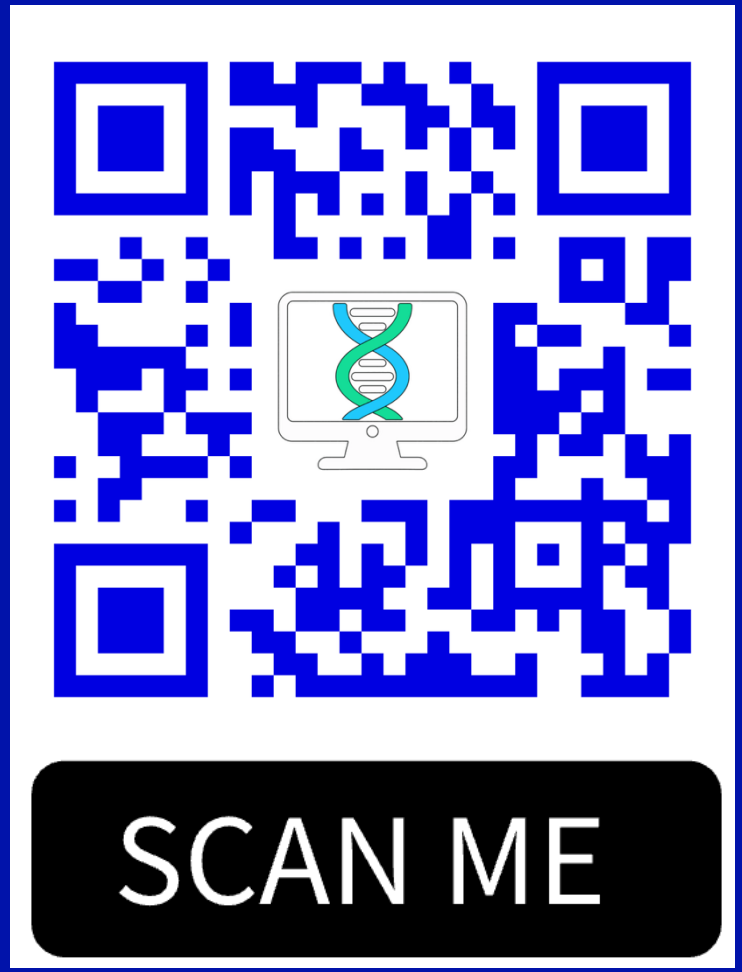
Introduction to deep learning

Bioinformatic tools

Corentin Maslard

November 02, 2026

Université de Neuchâtel
Faculty of Science



Quick poll: pe.app/cmaslard6

One question from Session 6 before we move on.

Nothing is graded today.

Take your time, try things, ask questions. You will meet artificial intelligence, use it and judge it during your whole career.

What is intelligence?

What is intelligence?

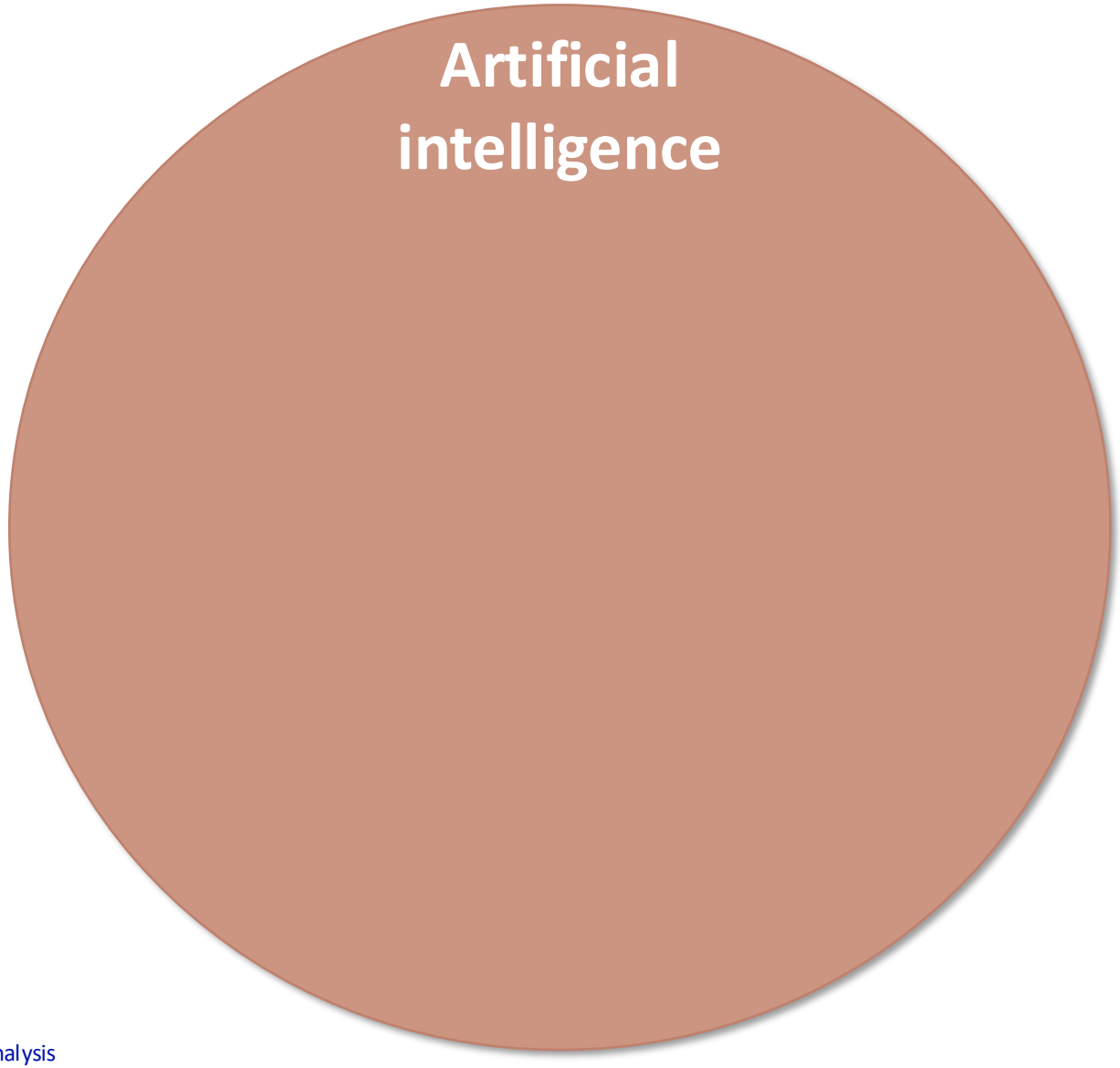
https://www.cleverbot.com/

What is intelligence?

https://www.cleverbot.com/

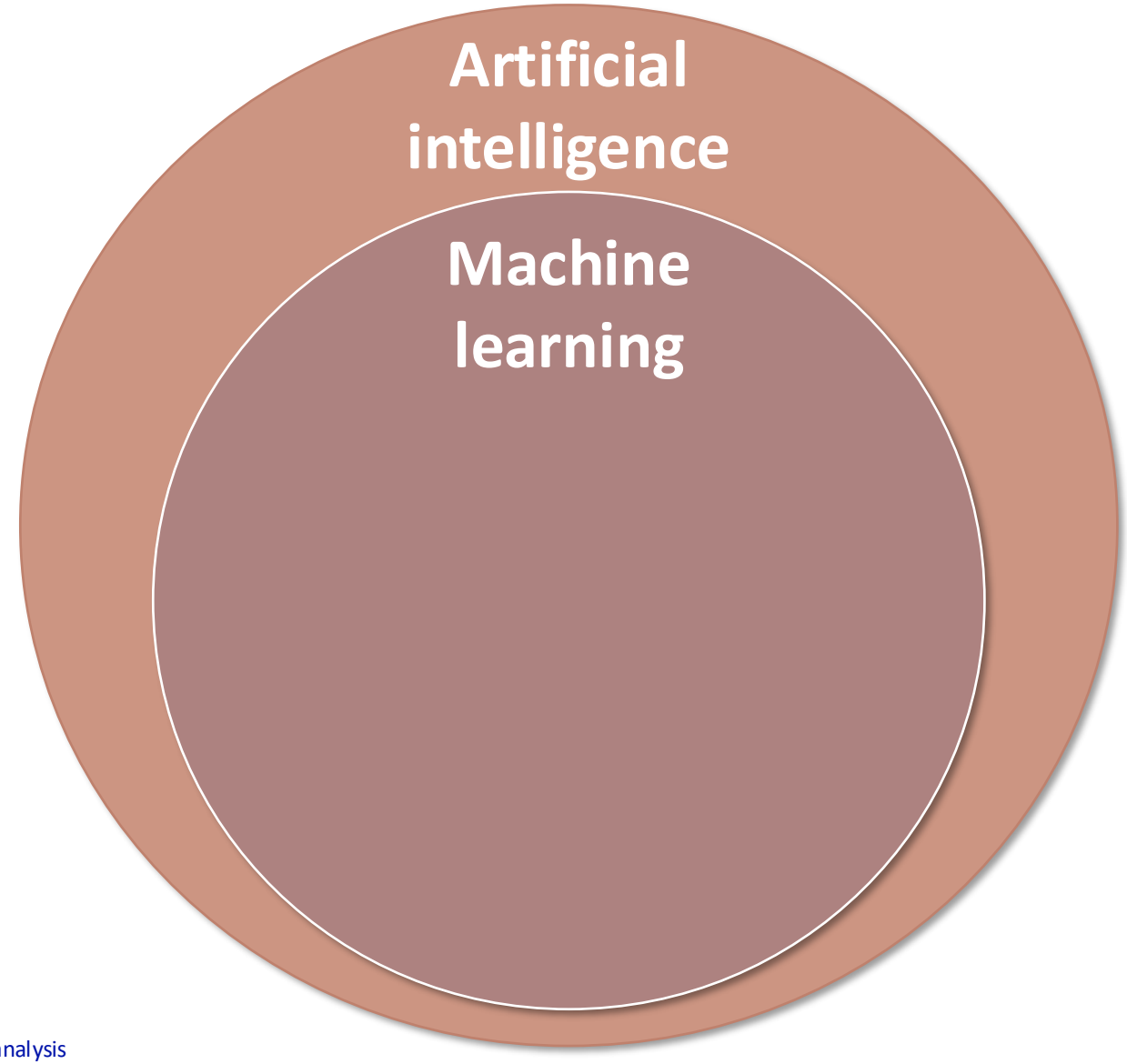
It adapts to the context and just picks words at random

AI, machine learning, deep learning, what are we talking about?



A science whose goal is to enable machines to perform tasks that humans accomplish using their intelligence.

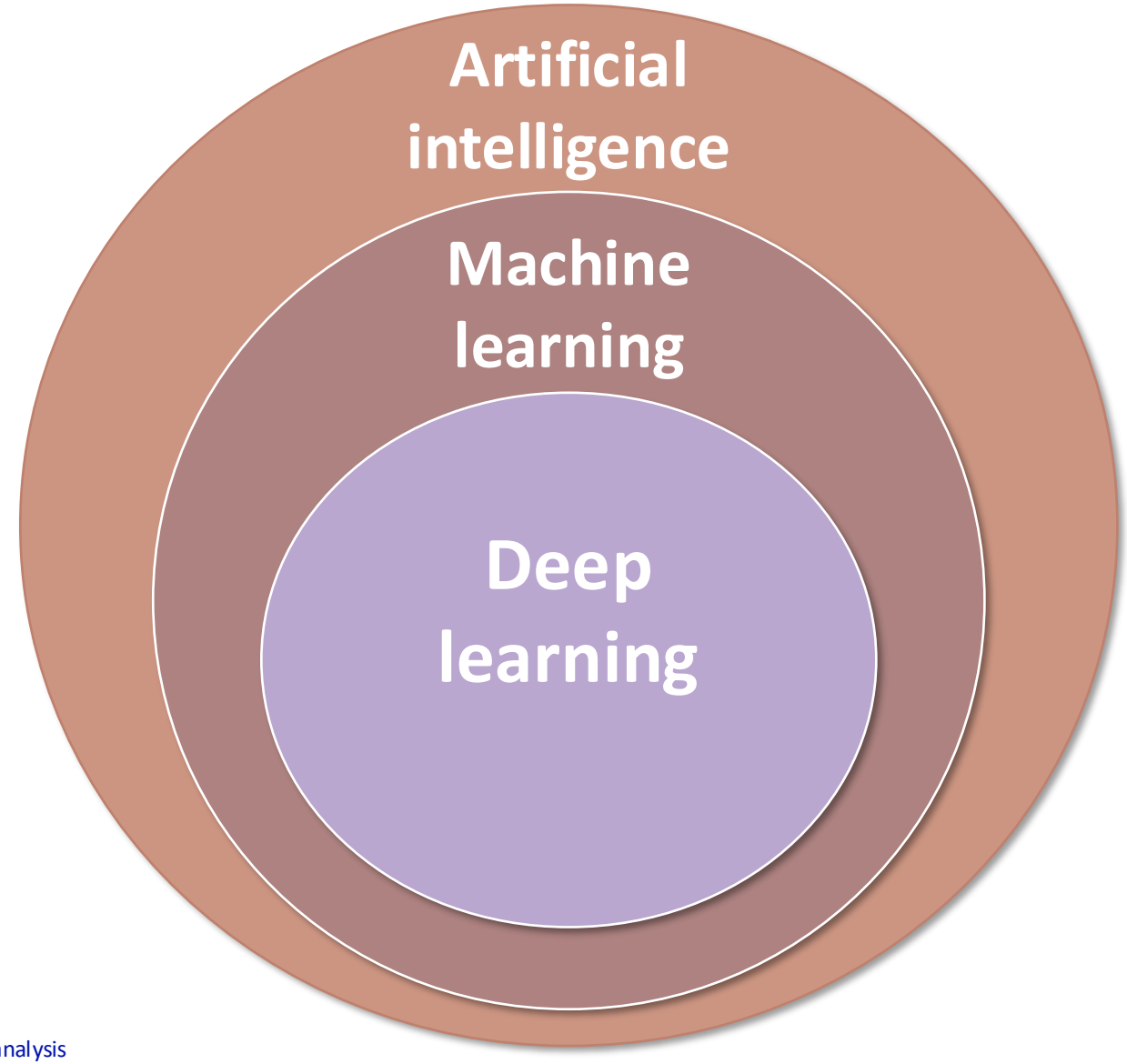
AI, machine learning, deep learning, what are we talking about?



Machine learning (ML) is a branch of artificial intelligence that enables computers to learn **without being explicitly programmed**.(1959)

Machine learning works on the basis of **examples**. It uses algorithms to statistically analyze data and **identify patterns**. These patterns are then used to predict results, gain a better understanding of the processes that generate this data, or make decisions.

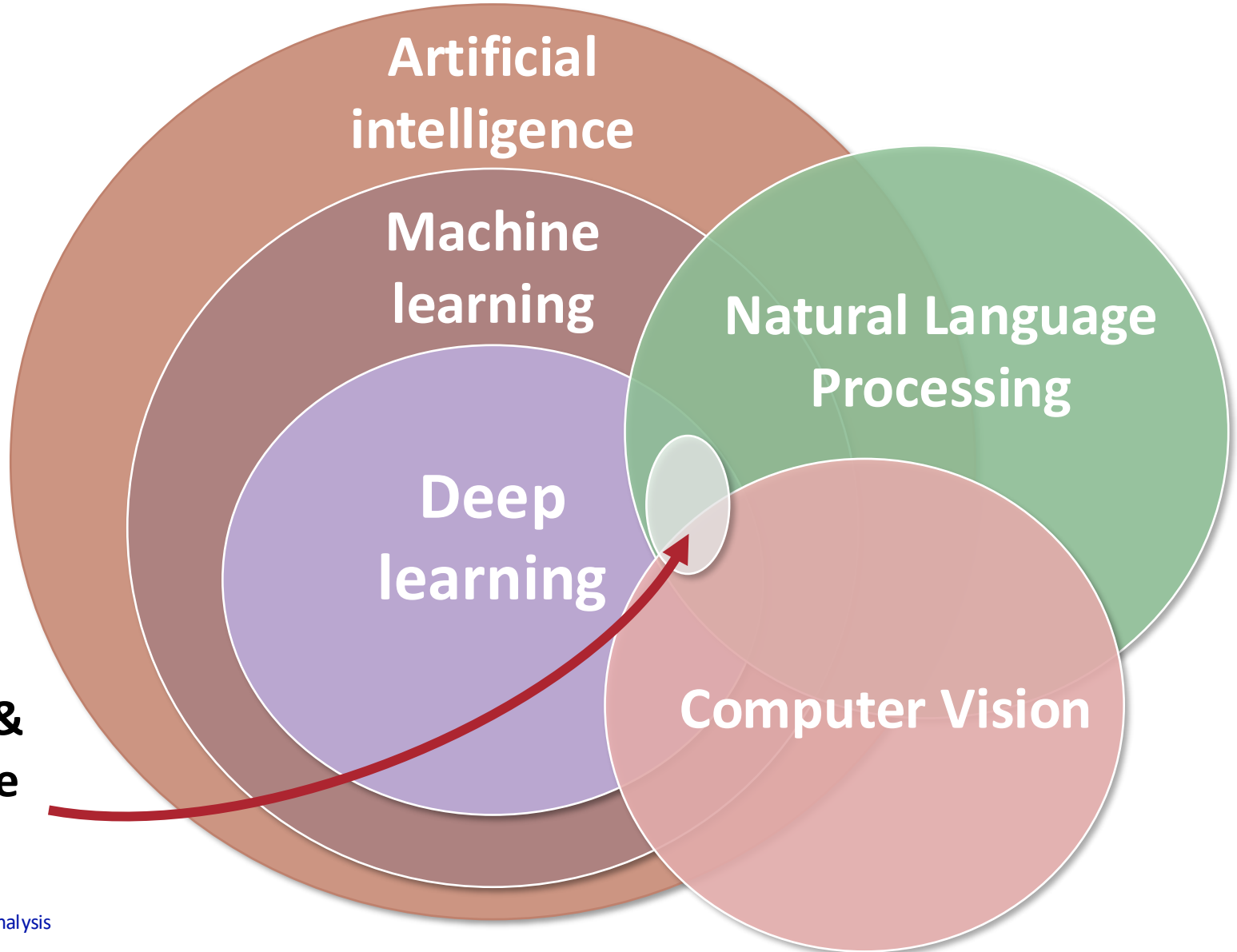
AI, machine learning, deep learning, what are we talking about?



Deep learning (DL) is a subset of machine learning. It uses **artificial neural networks** (ANNs), algorithms inspired by the functioning of the human brain, imitating the way neurons send signals to each other. These neurons are organized into interconnected layers with a certain level of depth. Each level of depth helps to optimize and refine the accuracy of the results.

Deep learning algorithms are well suited to solving complex problems. They **require a large volume of data** and therefore a great amount of computing power to process it.

AI, machine learning, deep learning, what are we talking about?



Generative AI & Large Language Models

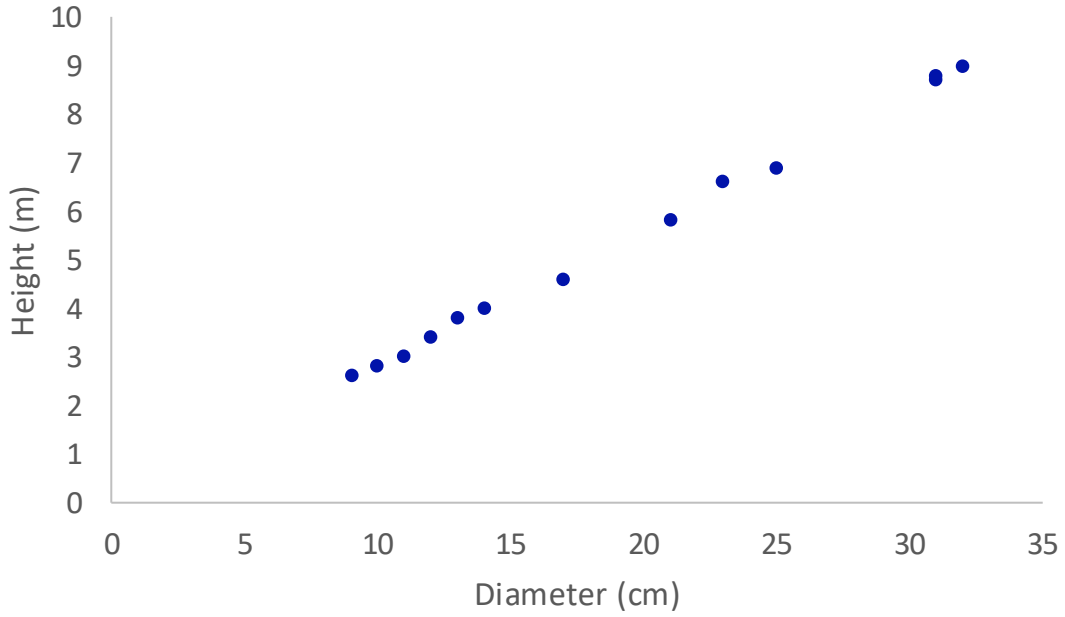
How does machine learning work? Generalization, predicting data...



Diameter (cm)	Height (m)
10	2,8
17	4,6
14	4,0
9	2,6
21	5,8
25	6,9
31	8,8
12	3,4
23	6,6
32	9
31	8,7
11	3
13	3,8

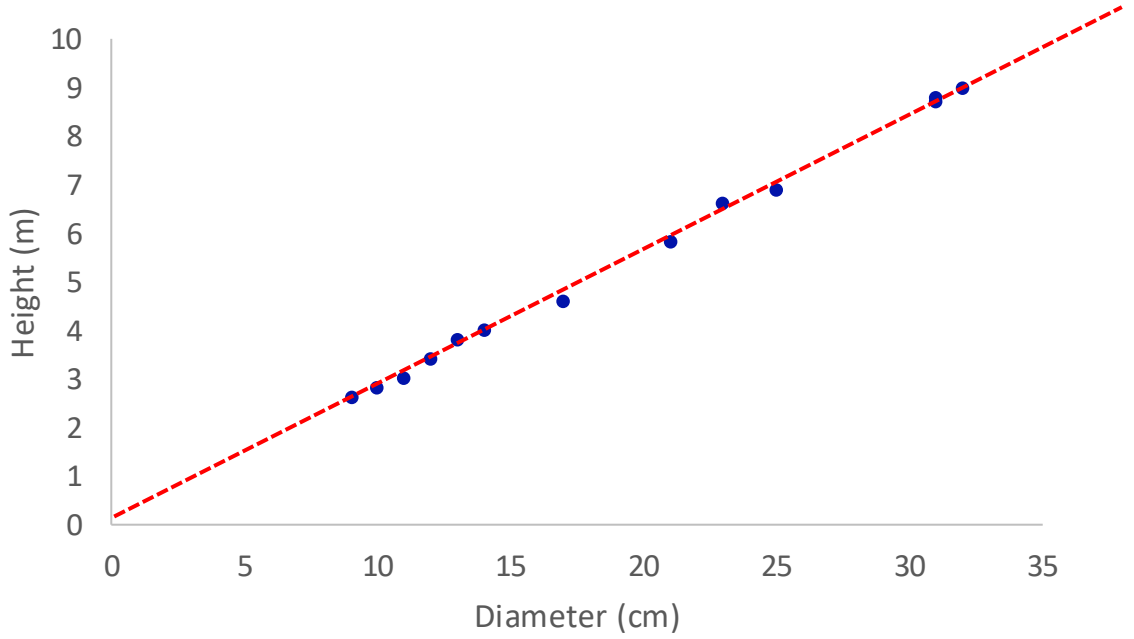
How does machine learning work? Generalization, predicting data...

Diameter (cm)	Height (m)
10	2,8
17	4,6
14	4,0
9	2,6
21	5,8
25	6,9
31	8,8
12	3,4
23	6,6
32	9
31	8,7
11	3
13	3,8



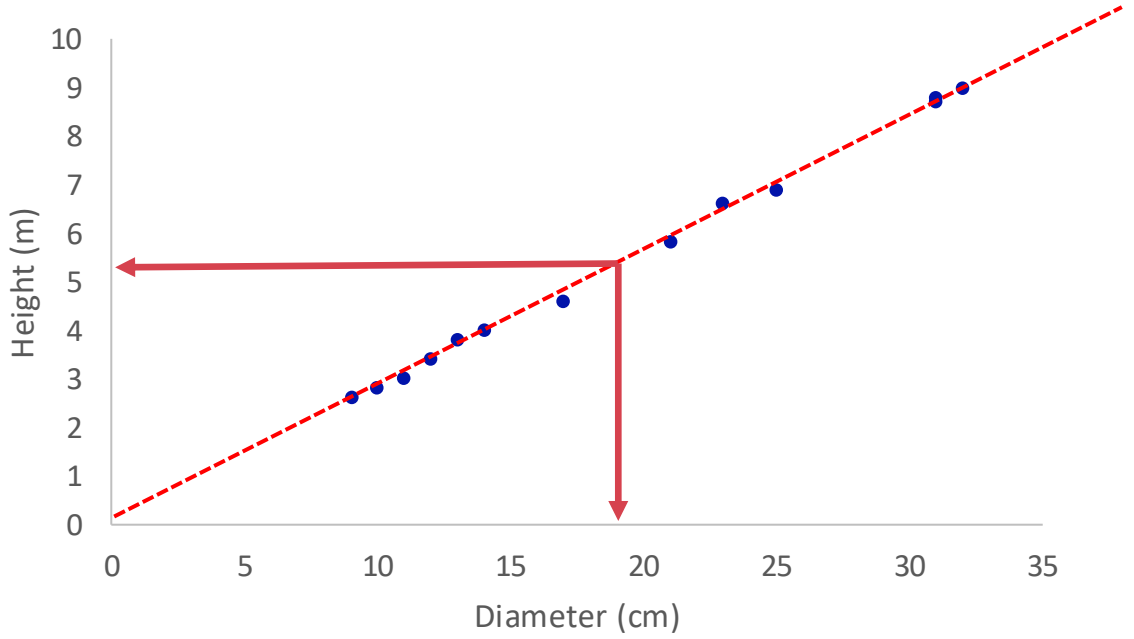
How does machine learning work? Generalization, predicting data...

Diameter (cm)	Height (m)
10	2,8
17	4,6
14	4,0
9	2,6
21	5,8
25	6,9
31	8,8
12	3,4
23	6,6
32	9
31	8,7
11	3
13	3,8



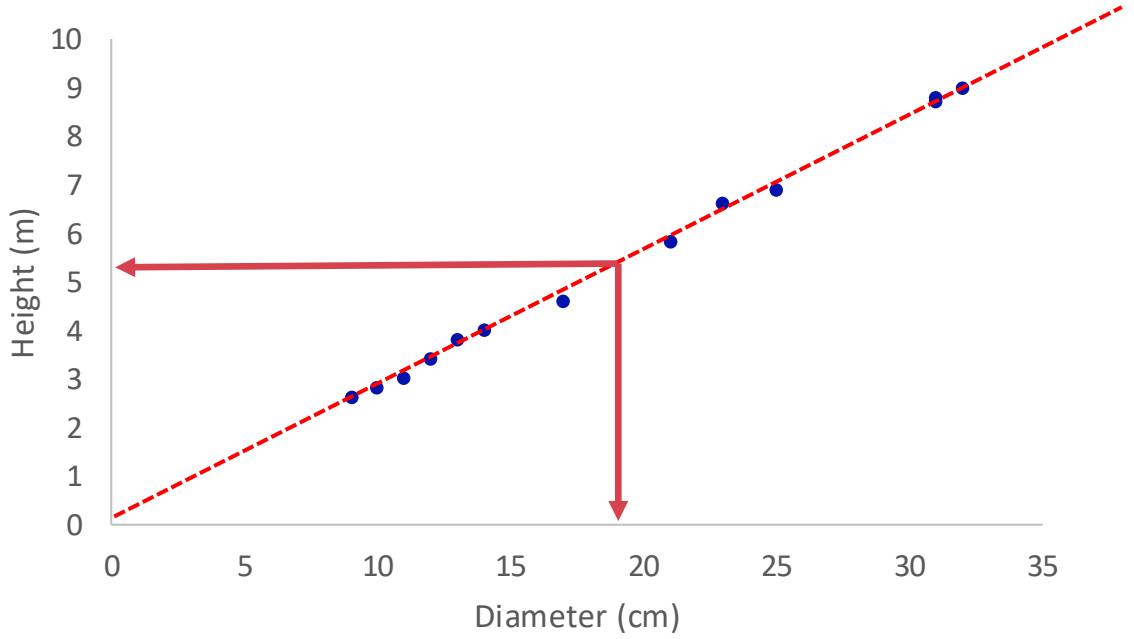
How does machine learning work? Generalization, predicting data...

Diameter (cm)	Height (m)
10	2,8
17	4,6
14	4,0
9	2,6
21	5,8
25	6,9
31	8,8
12	3,4
23	6,6
32	9
31	8,7
11	3
13	3,8
19	?



How does machine learning work? Generalization, predicting data...

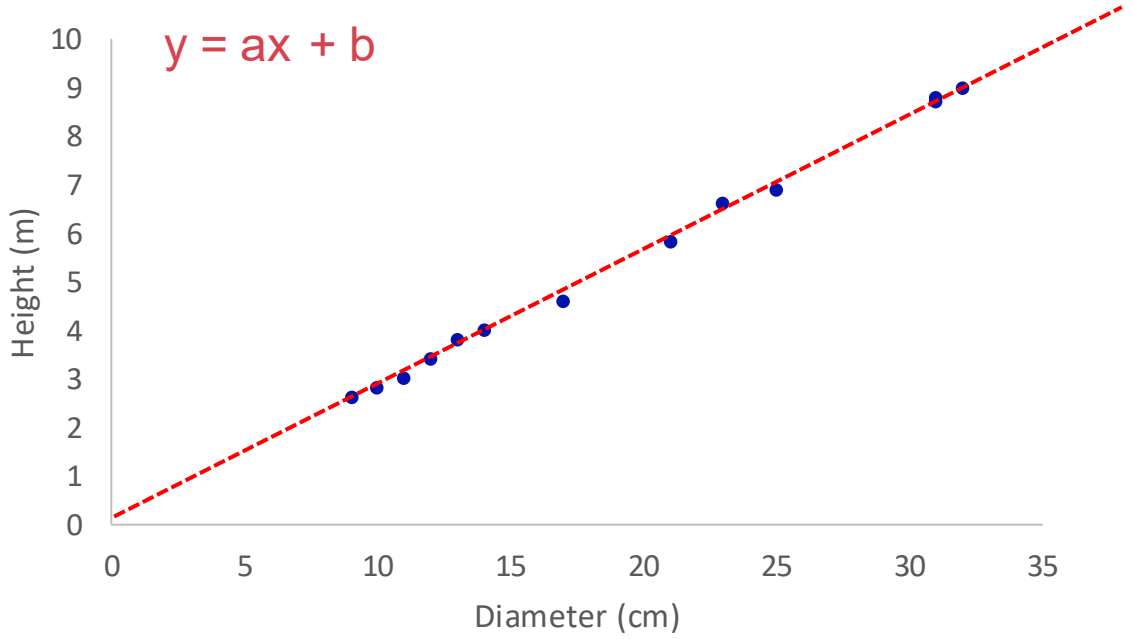
Diameter (cm)	Height (m)
10	2,8
17	4,6
14	4,0
9	2,6
21	5,8
25	6,9
31	8,8
12	3,4
23	6,6
32	9
31	8,7
11	3
13	3,8
19	?



Data (observation) → links between data → generalisation, prediction

How does machine learning work? Generalization, predicting data...

Diameter (cm)	Height (m)
10	2,8
17	4,6
14	4,0
9	2,6
21	5,8
25	6,9
31	8,8
12	3,4
23	6,6
32	9
31	8,7
11	3
13	3,8



a

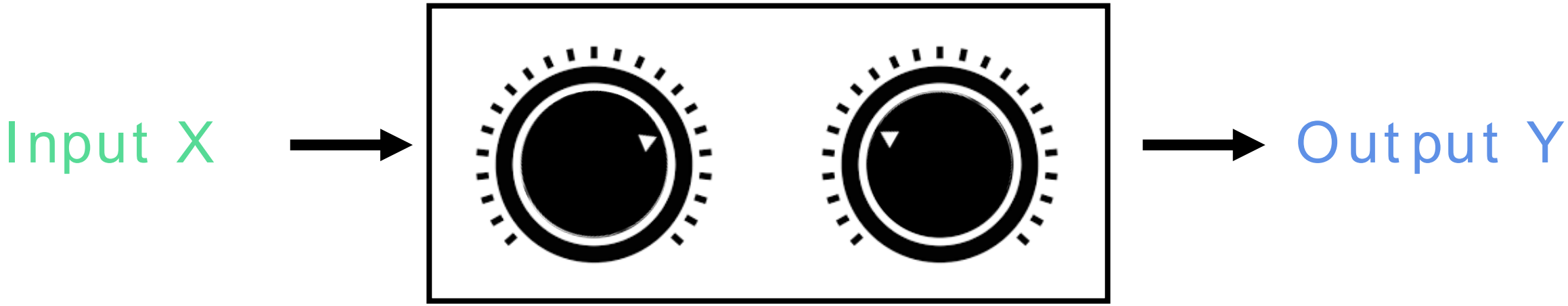


b



How does machine learning work? Generalization, predicting data...

X	Y
...	...
...	...
...	...
...	...
...	...

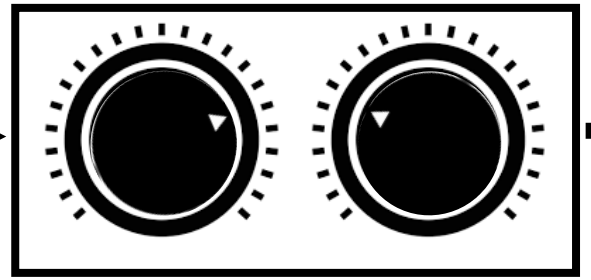


Learning phase

Exemple of the Facebook feed algorithm



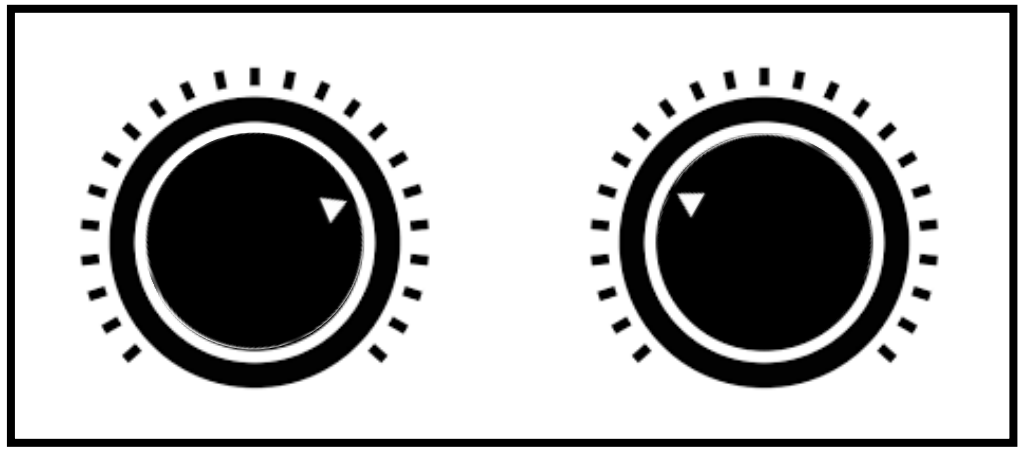
Who posted?
When?
Nature
Subject
Reactions from others
...



Will you like, click, comment, interact...?

Image recognition

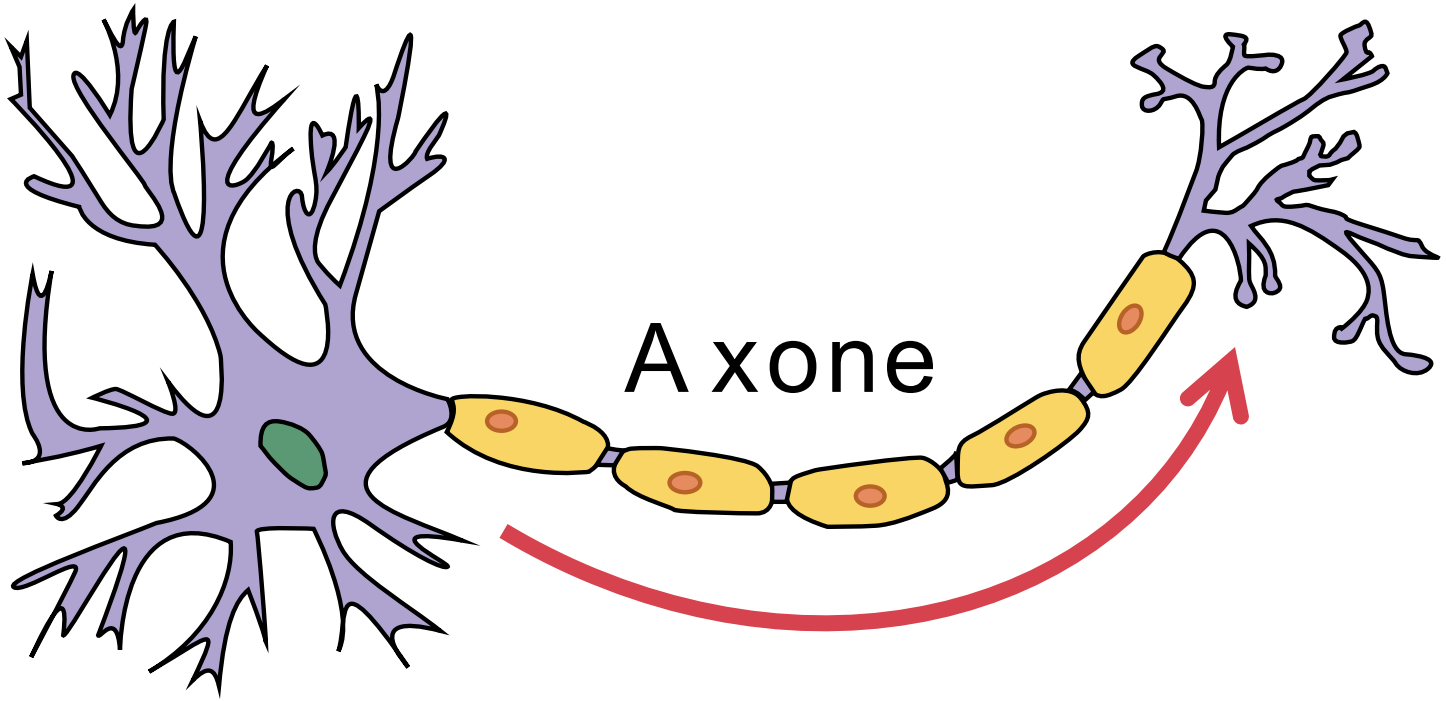




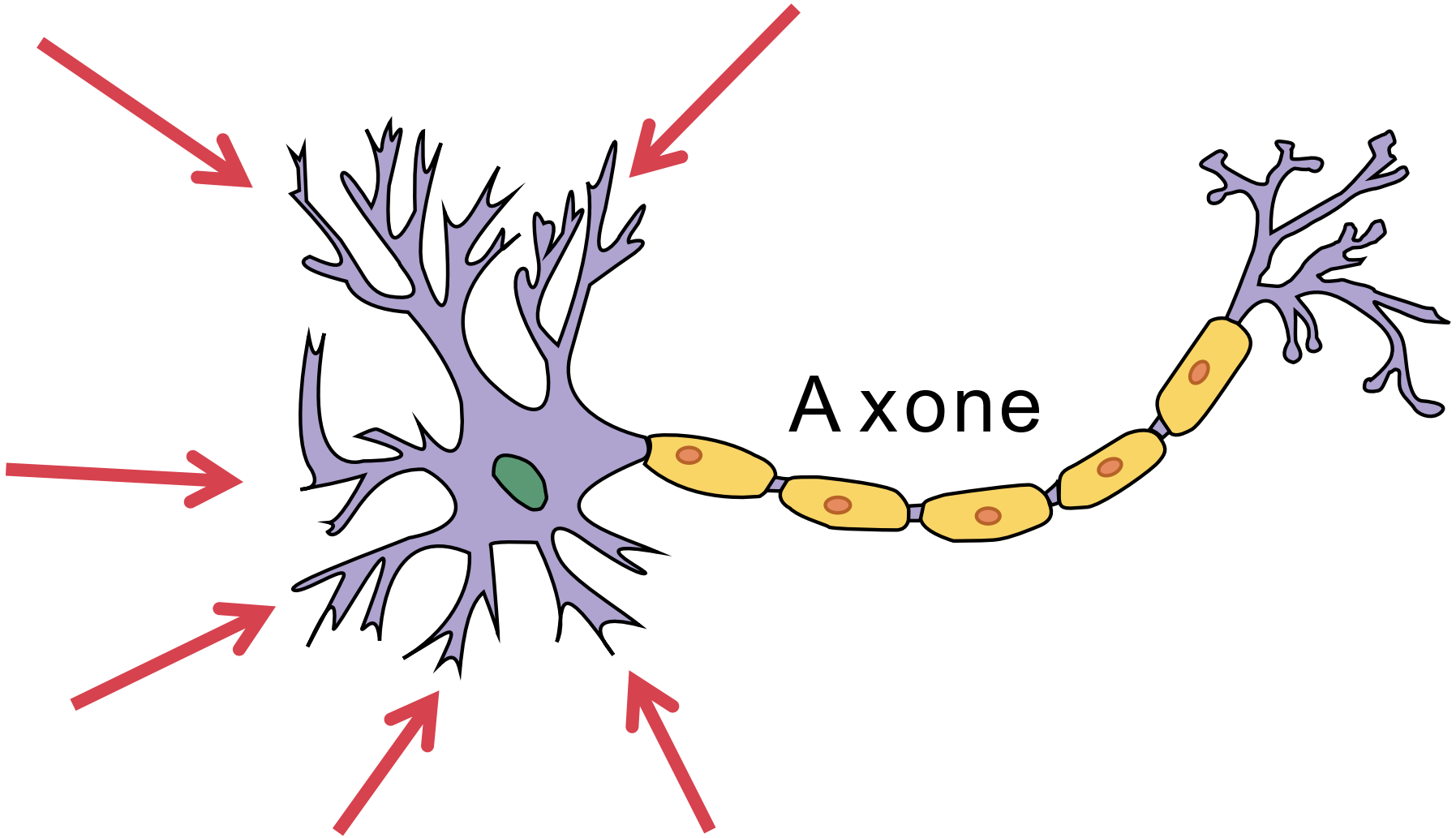
Learning phase

Flower?
Heron?
Lamp?
Robot?
...

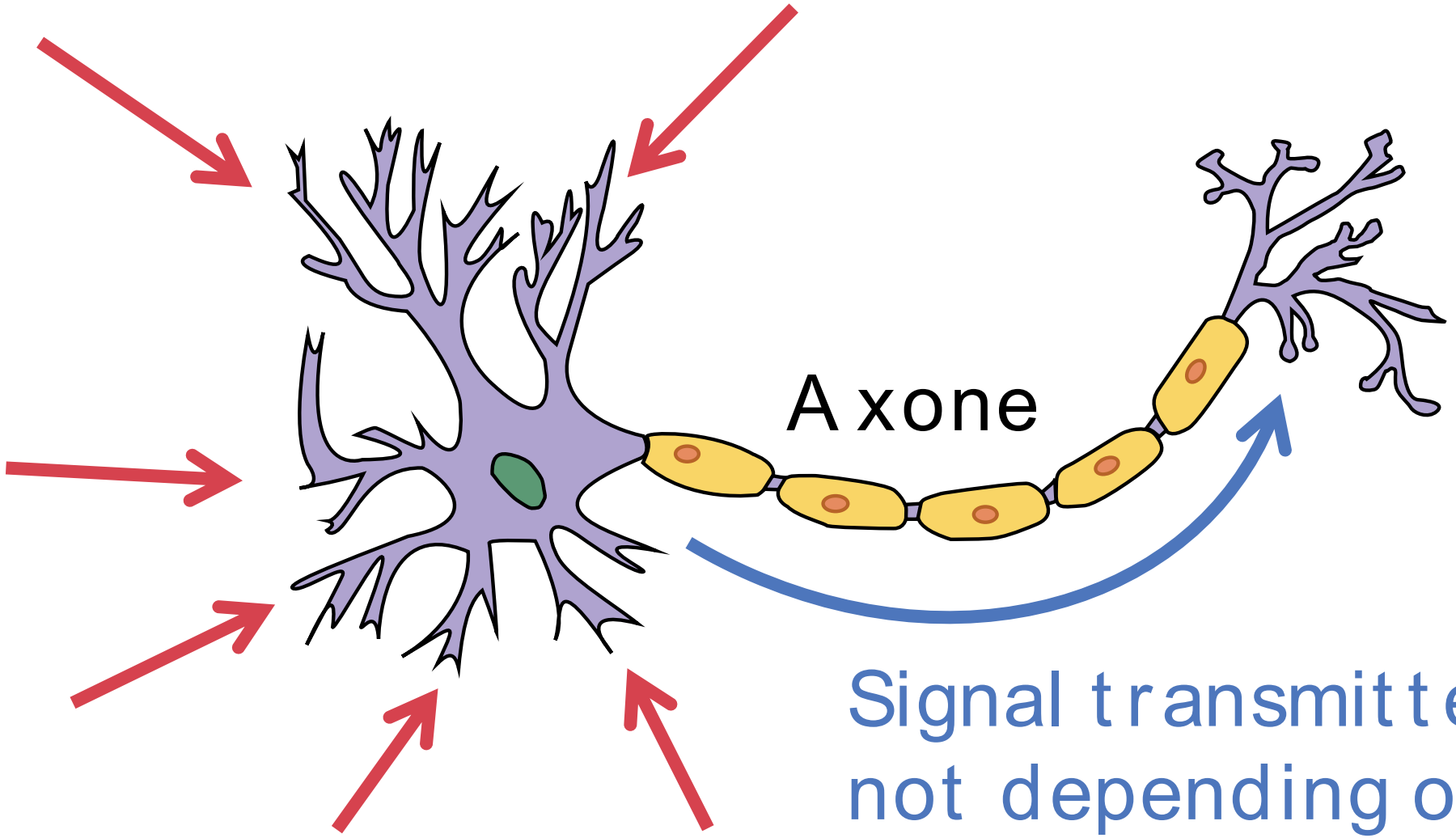
Biological neuron



Biological neuron

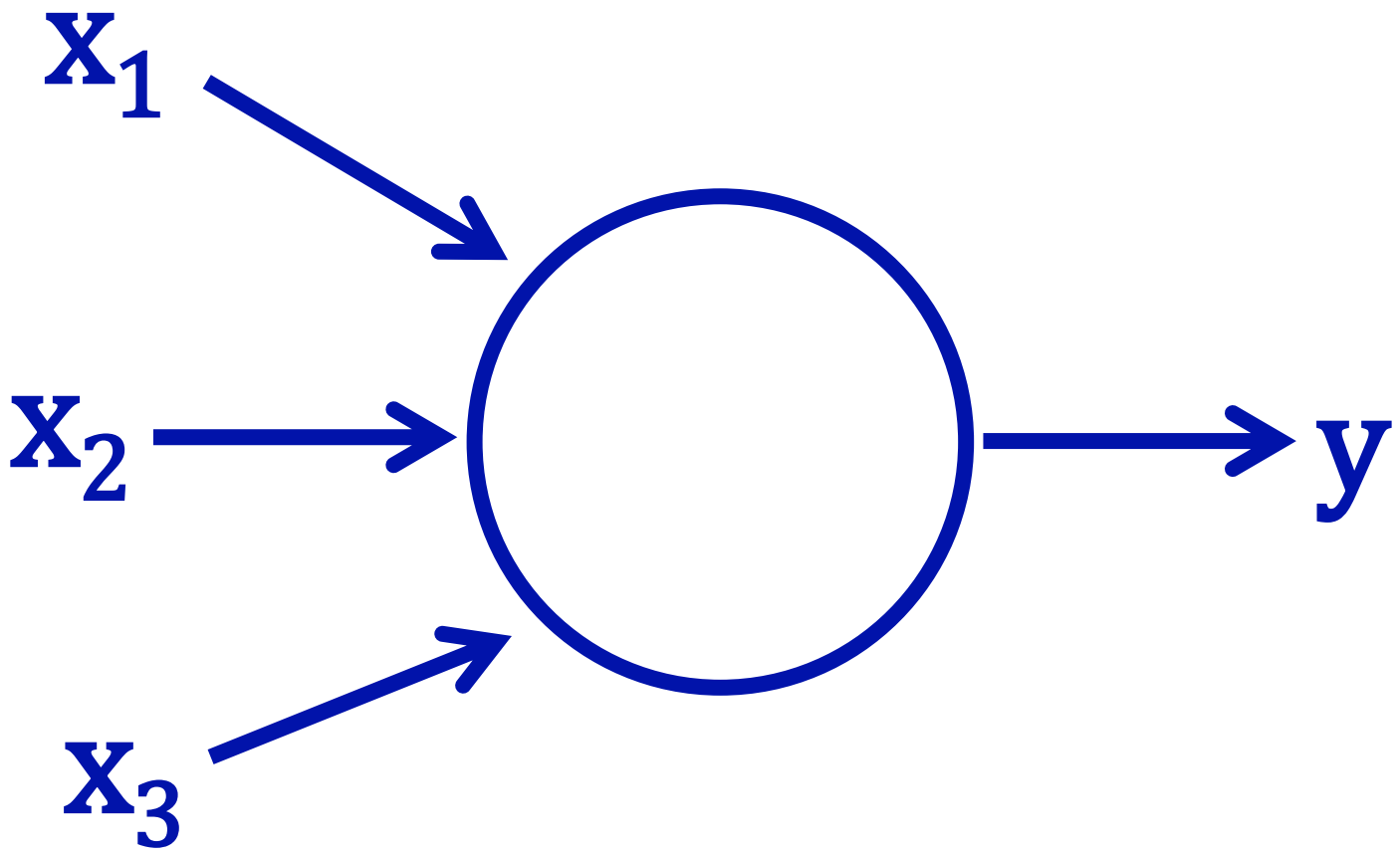


Biological neuron

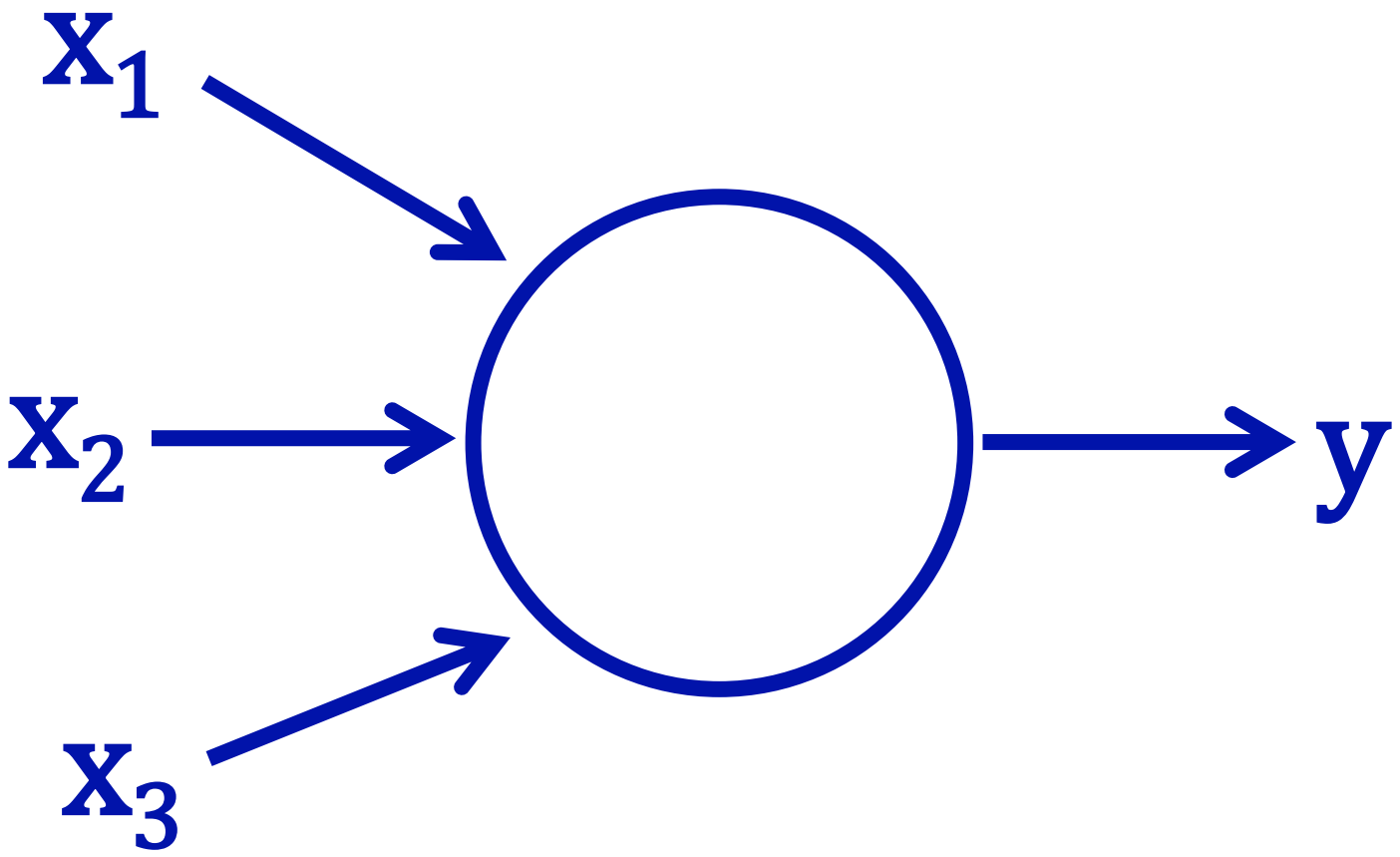


Signal transmitted or not depending on the information received

Artificial neuron

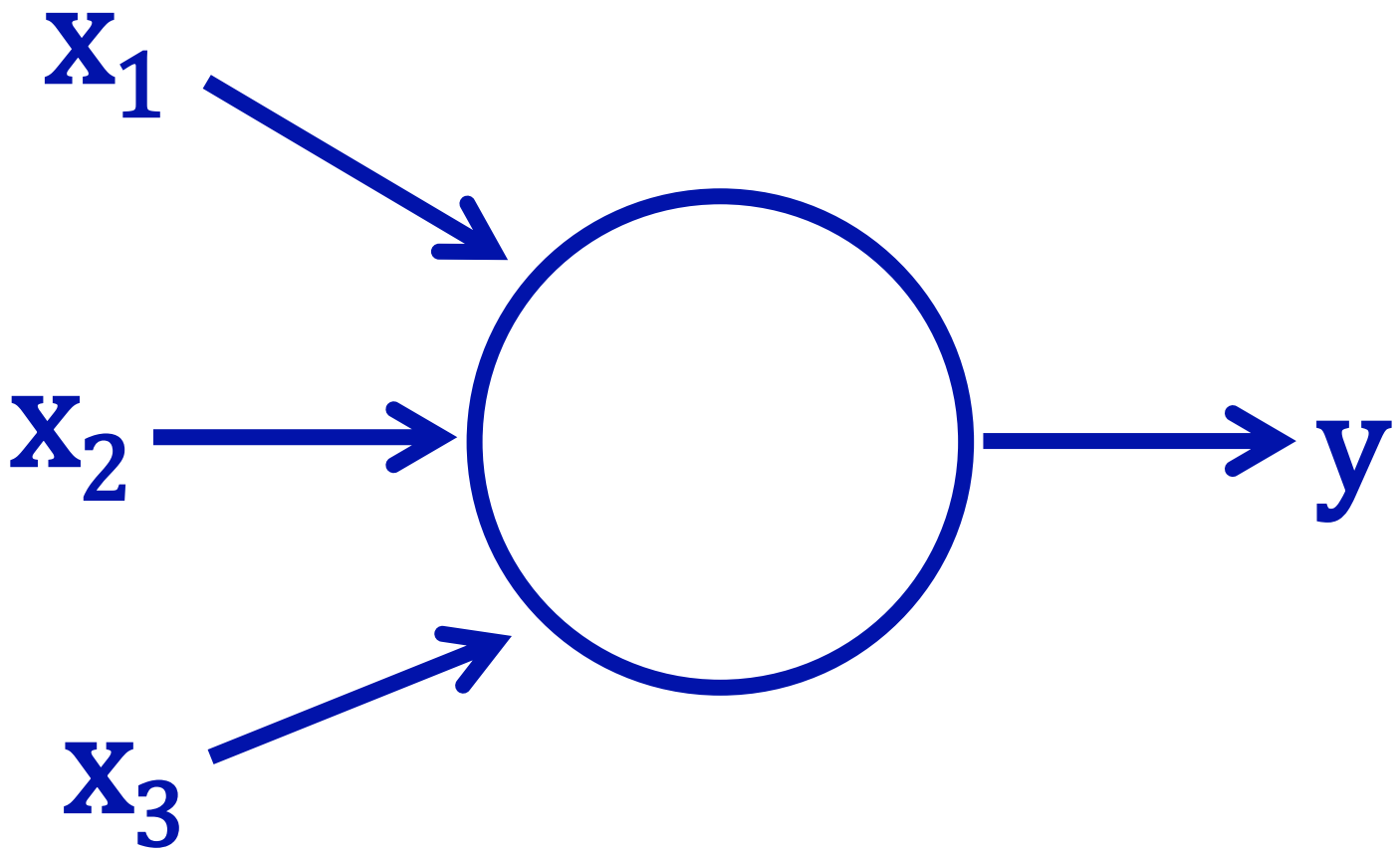


Artificial neuron



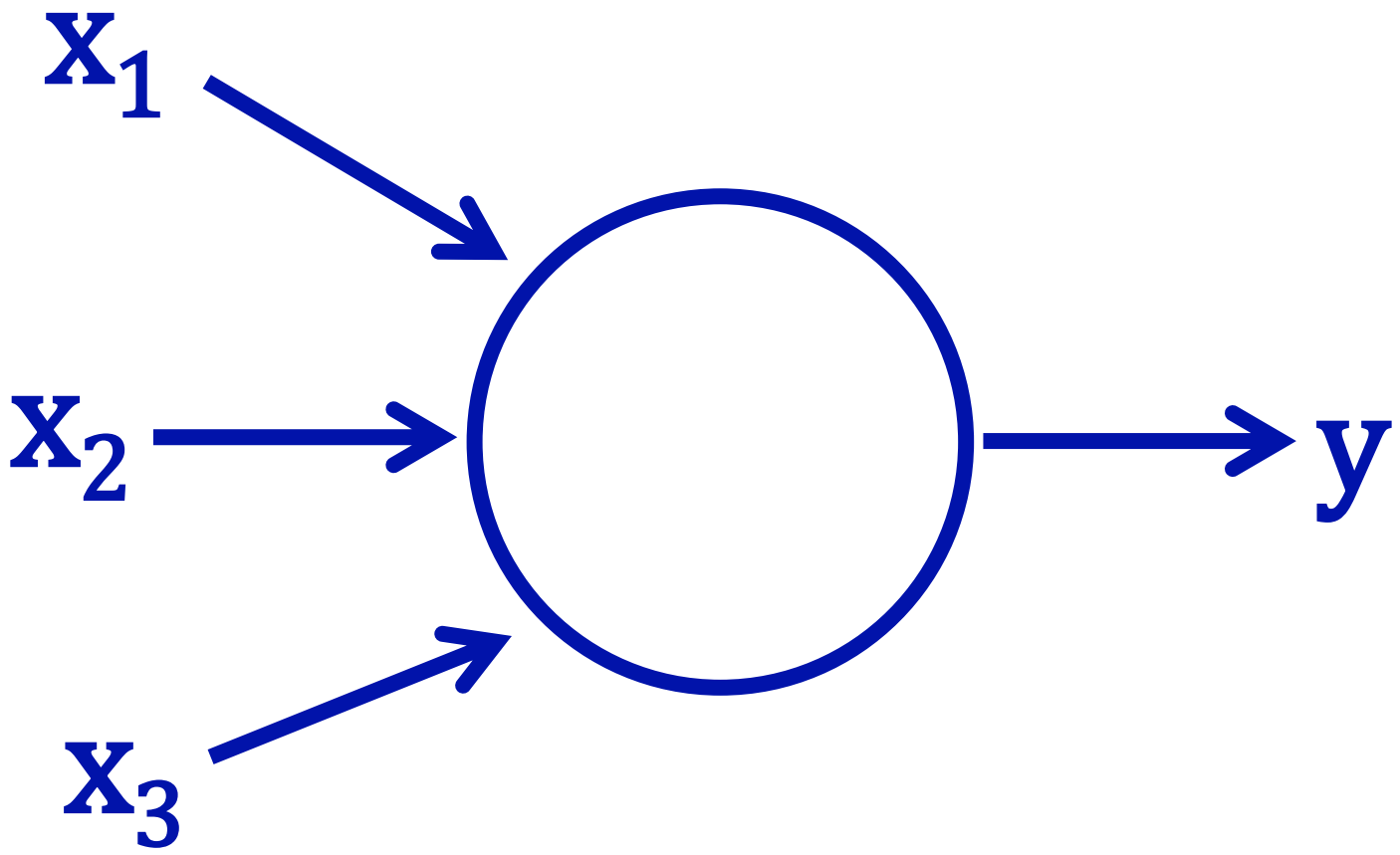
$$x_1 + \quad x_2 + \quad x_3$$

Artificial neuron



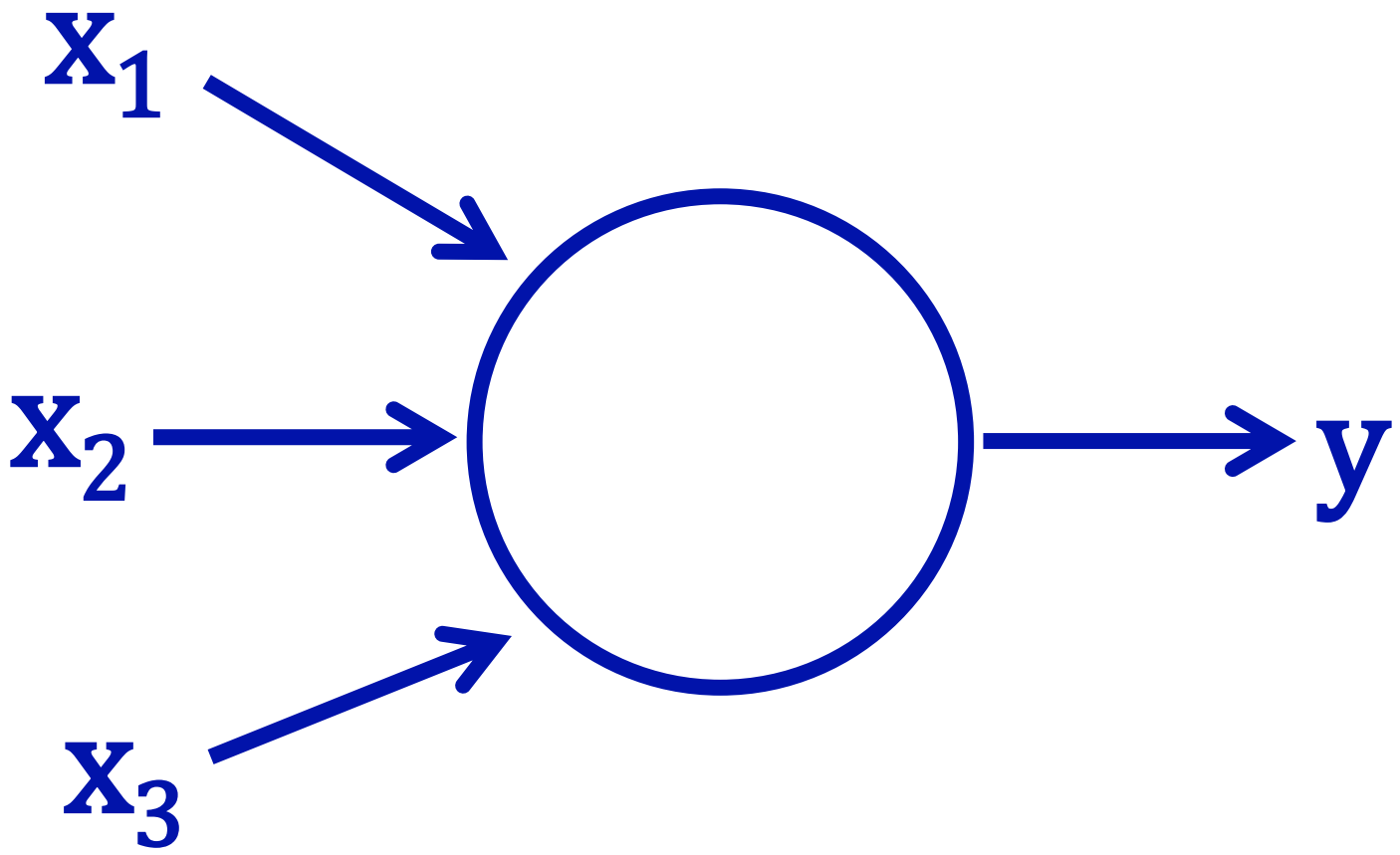
$$p_1x_1+p_2x_2+p_3x_3$$

Artificial neuron



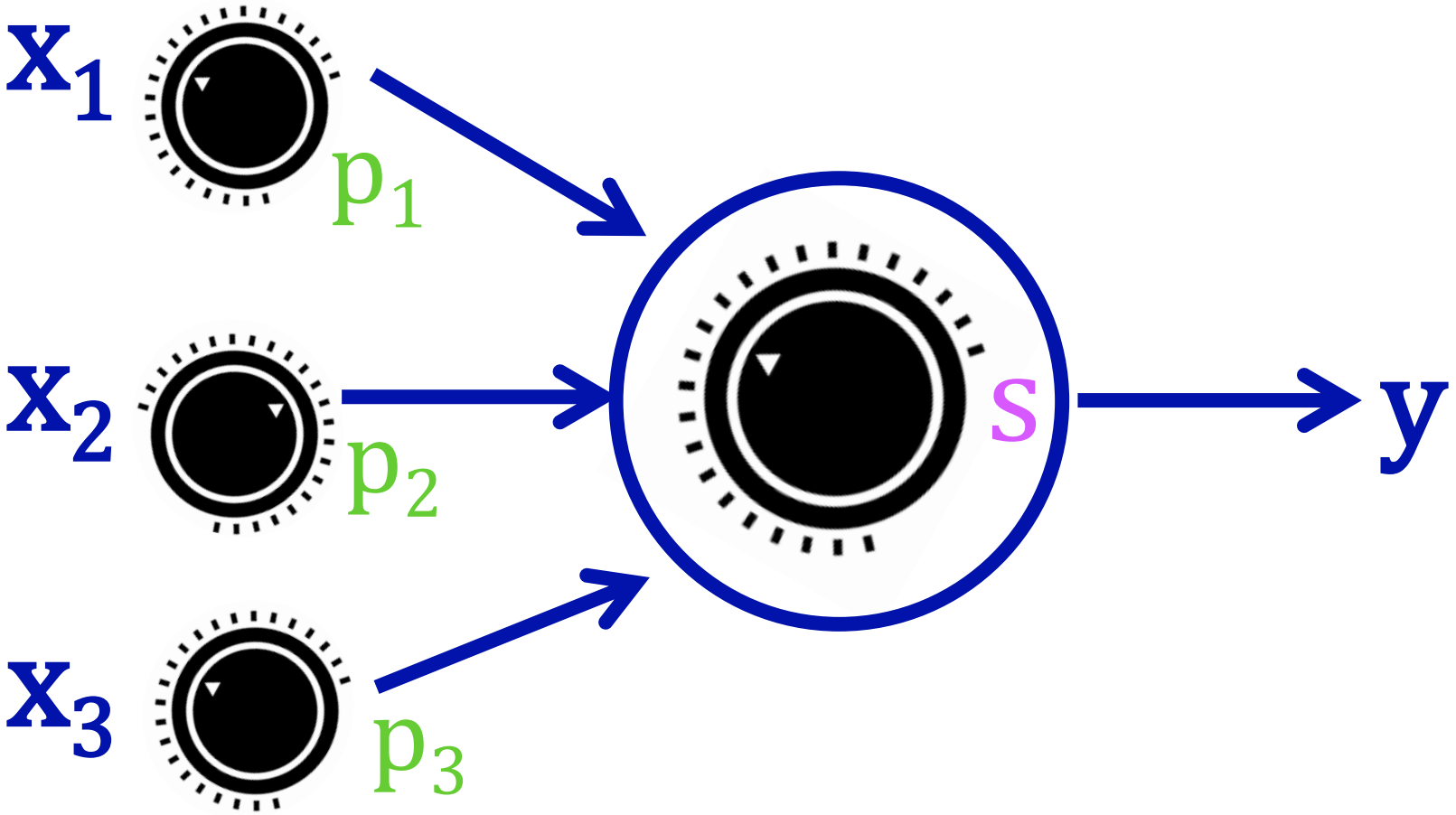
$$p_1x_1 + p_2x_2 + p_3x_3 > s \; ? \rightarrow y = 1$$

Artificial neuron



$$p_1x_1+p_2x_2+p_3x_3 > s \; ? \rightarrow y = 1$$
$$< s \; ? \rightarrow y = 0$$

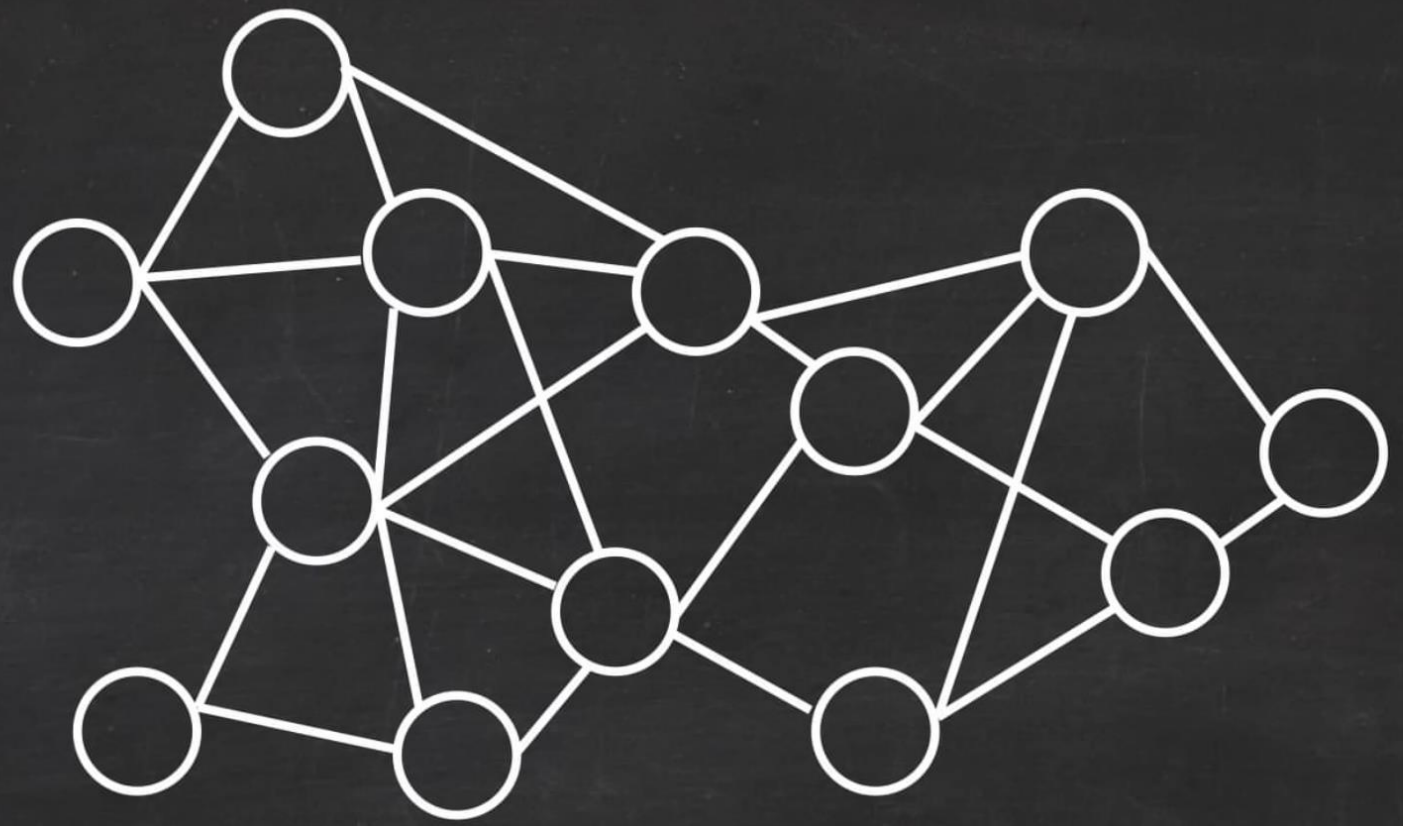
Artificial neuron



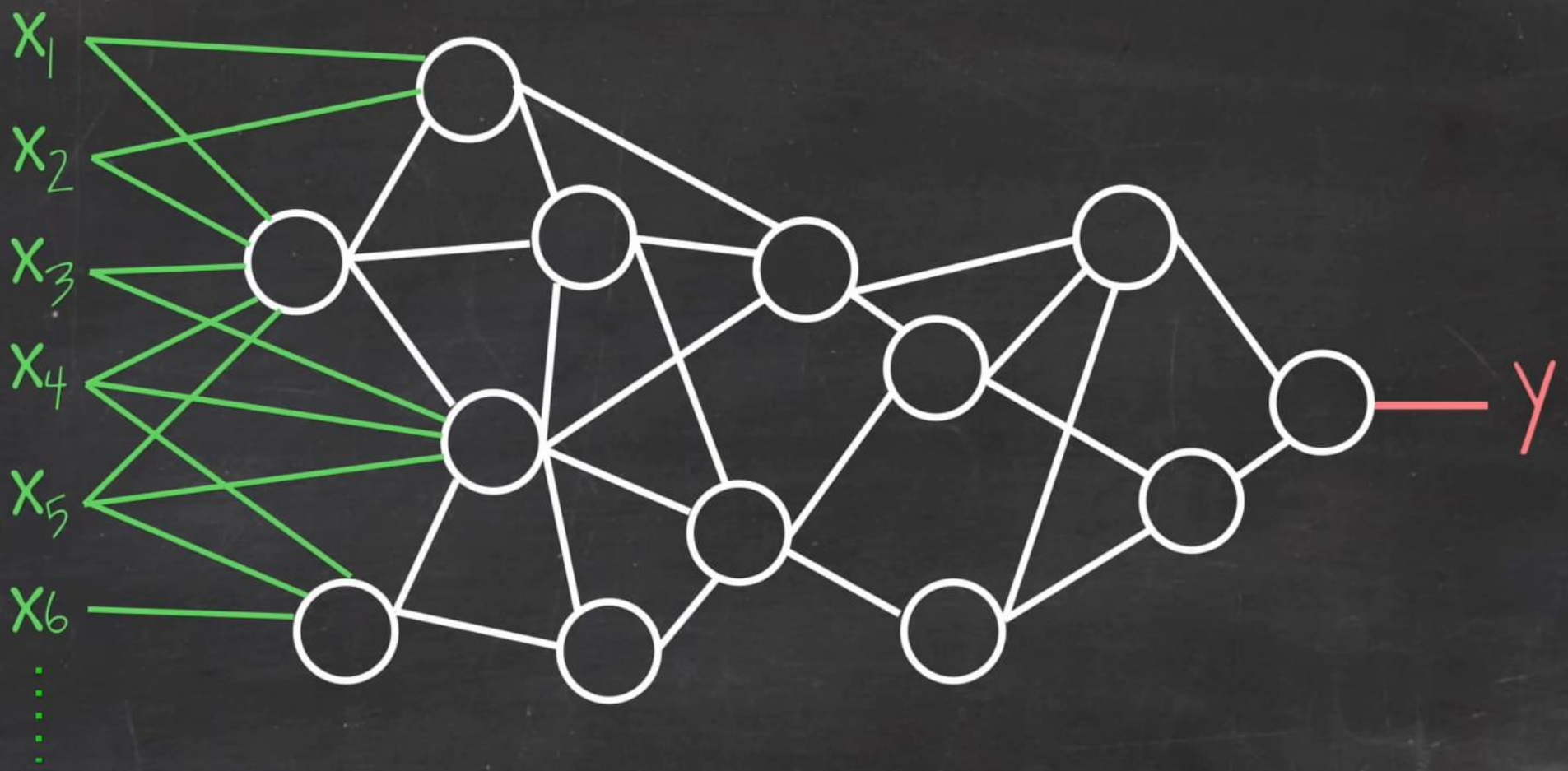
$$p_1x_1+p_2x_2+p_3x_3 > s \; ? \rightarrow y = 1$$

$$< s \; ? \rightarrow y = 0$$

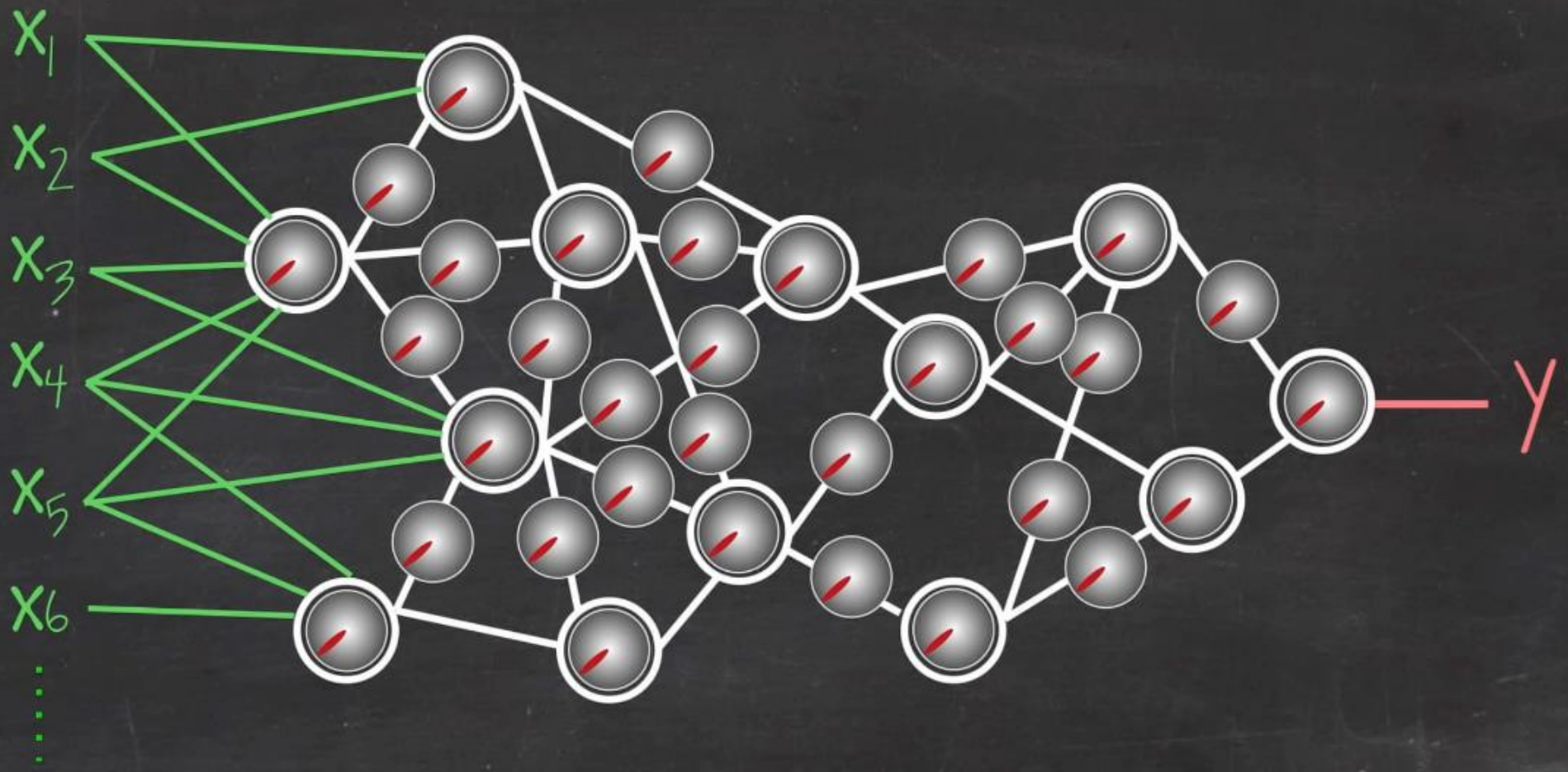
Neural network (artificial)



Neural network (artificial)

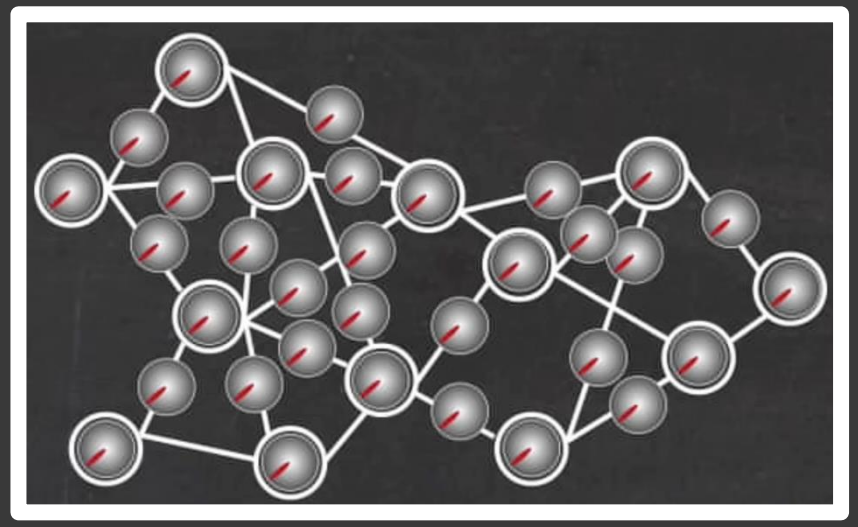


Neural network (artificial)



Learning phase

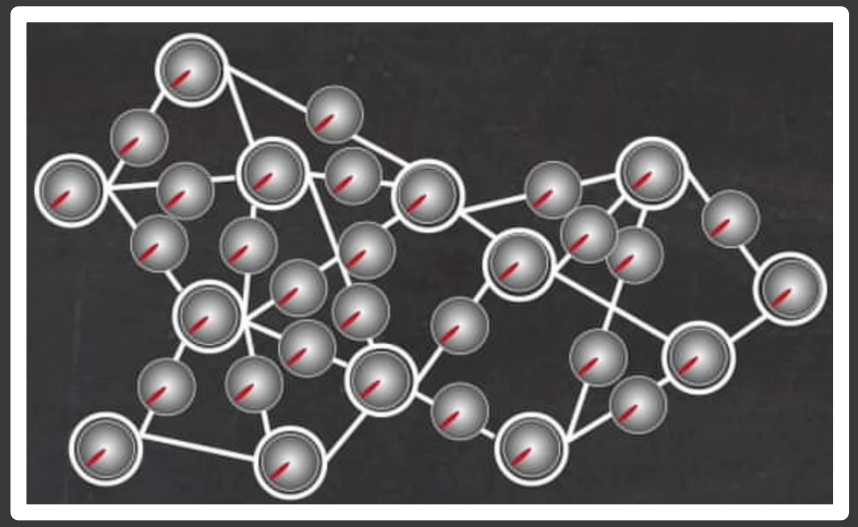
Input
X



Output
Y

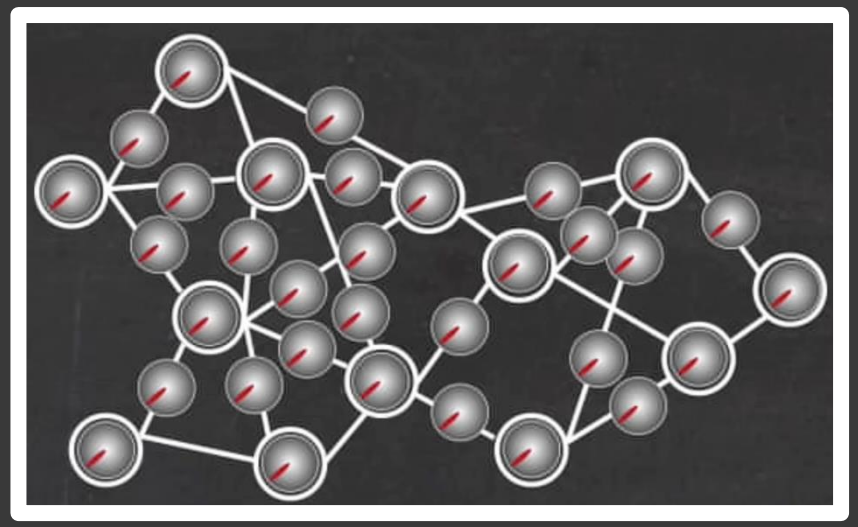
Learning phase

Input
X



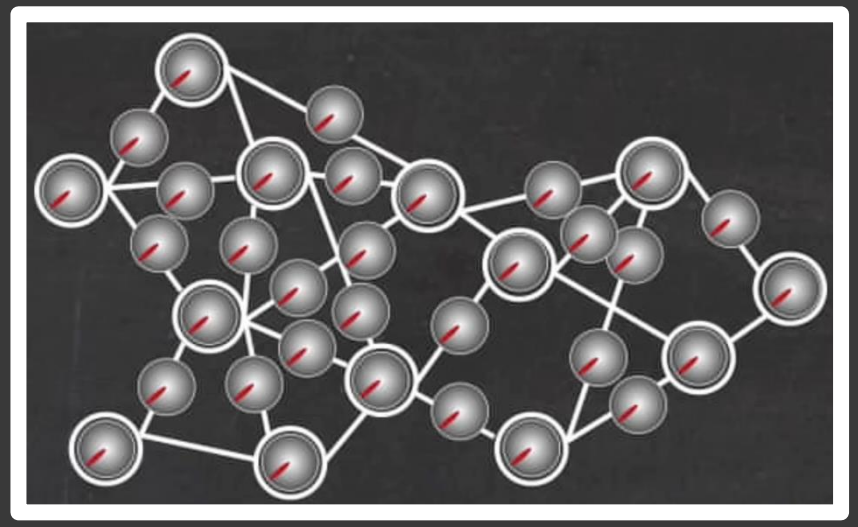
Output
Y

New
entry
X



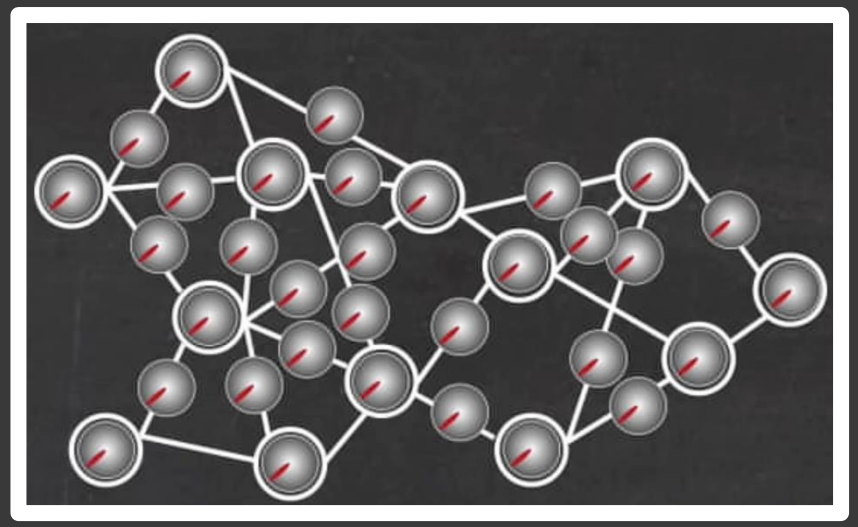
Learning phase

Input
X



Output
Y

New
entry
X



Y
prediction

But how many neurons are needed?

What is their best arrangement?

There are 100 billion neurons in our brain.

But how many neurons are needed?

What is their best arrangement?

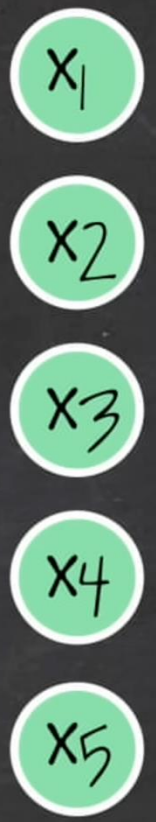
There are 100 billion neurons in our brain.

But how many neurons are needed?

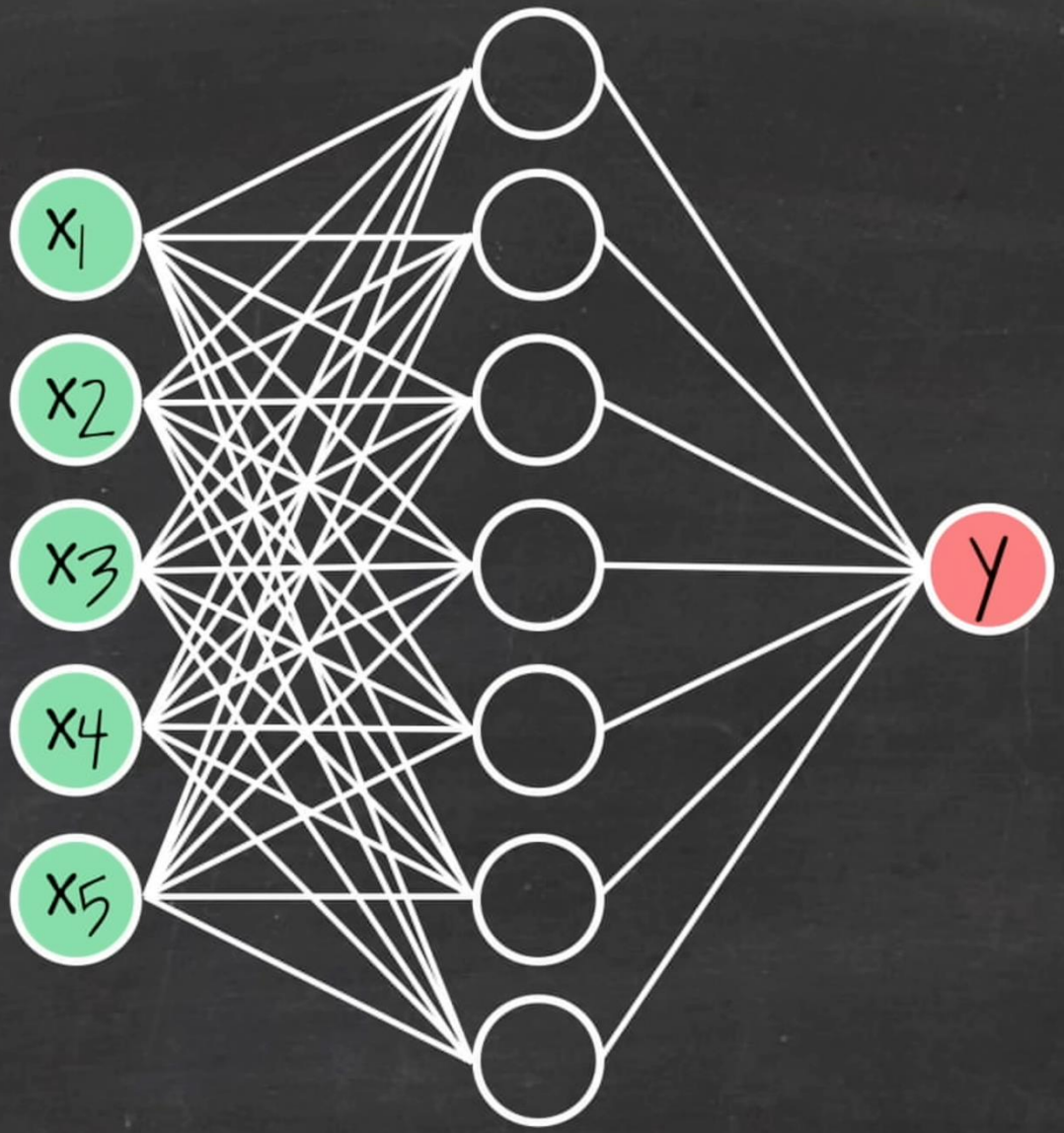
What is their best arrangement?

There are 100 billion neurons in our brain.

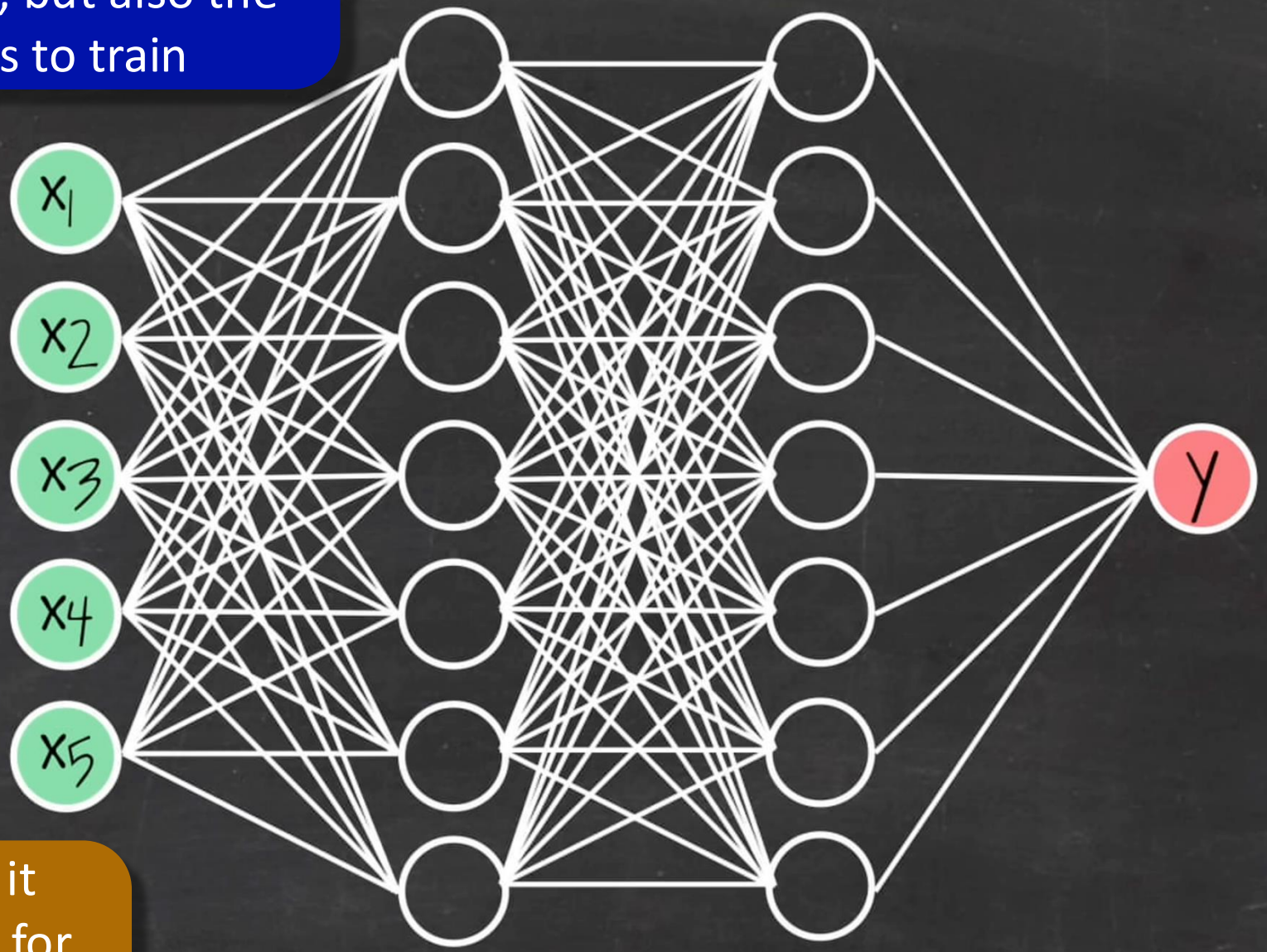
Several thousand synapses per neuron, meaning several million billion connections. That's too much for a simple computer


x1
x2
x3
x4
x5


x1
x2
x3
x4
x5
y


x1
x2
x3
x4
x5
y

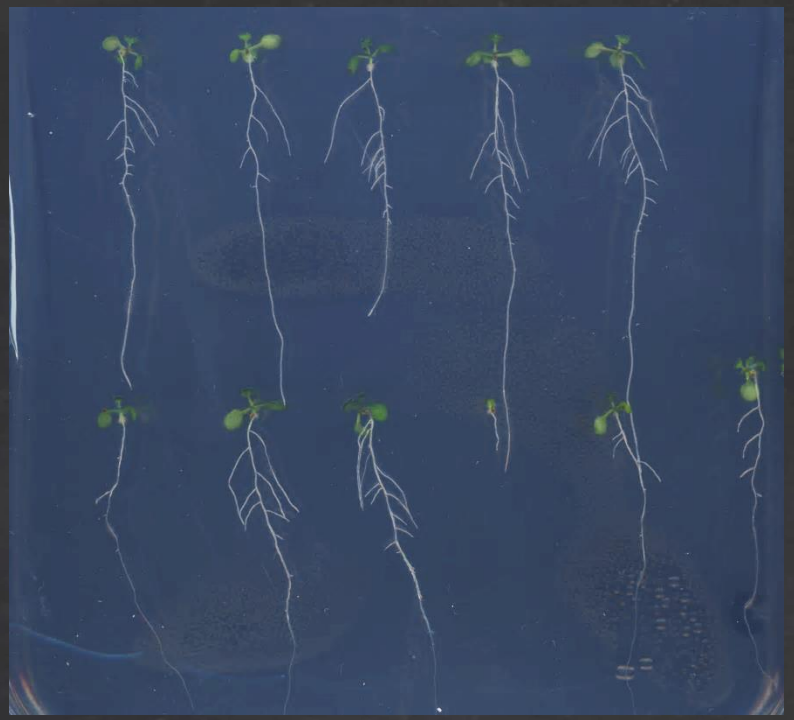
The more neurons there are, the more powerful the algorithm is, but also the more complicated it is to train



For simple algorithms, it works very well. But not for image analysis

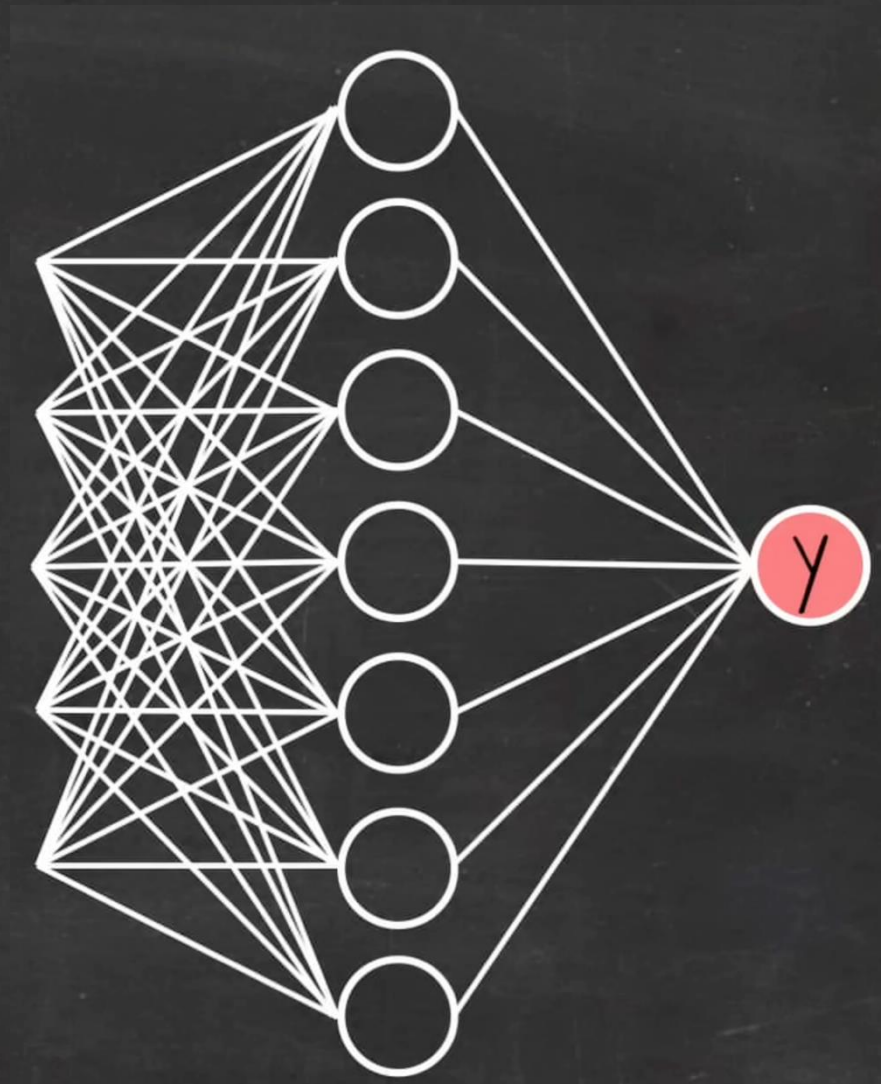
Image recognition is quickly limited (**machine learning**, not deep learning)

Image 5 000 x 5 000

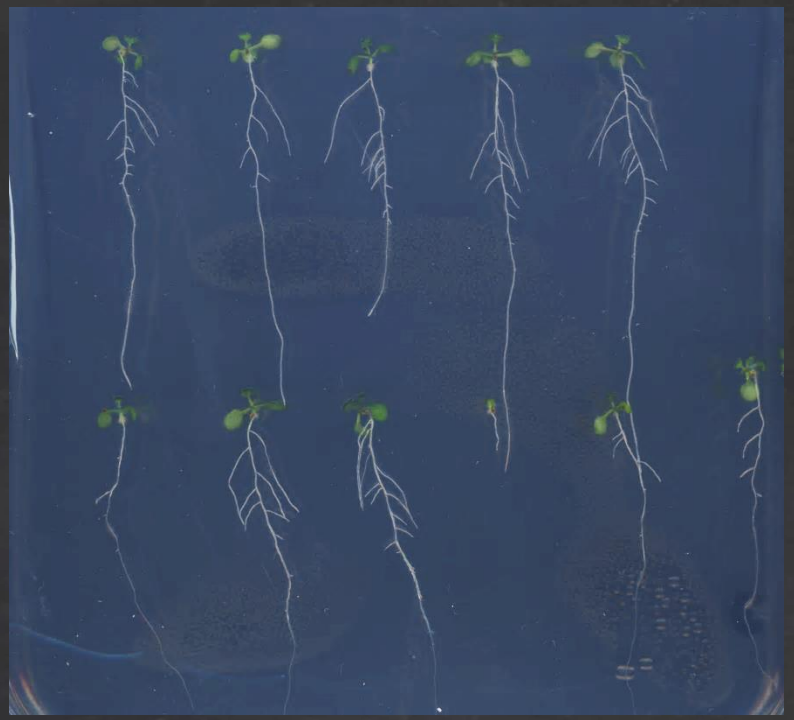


25000000 in number of input

?

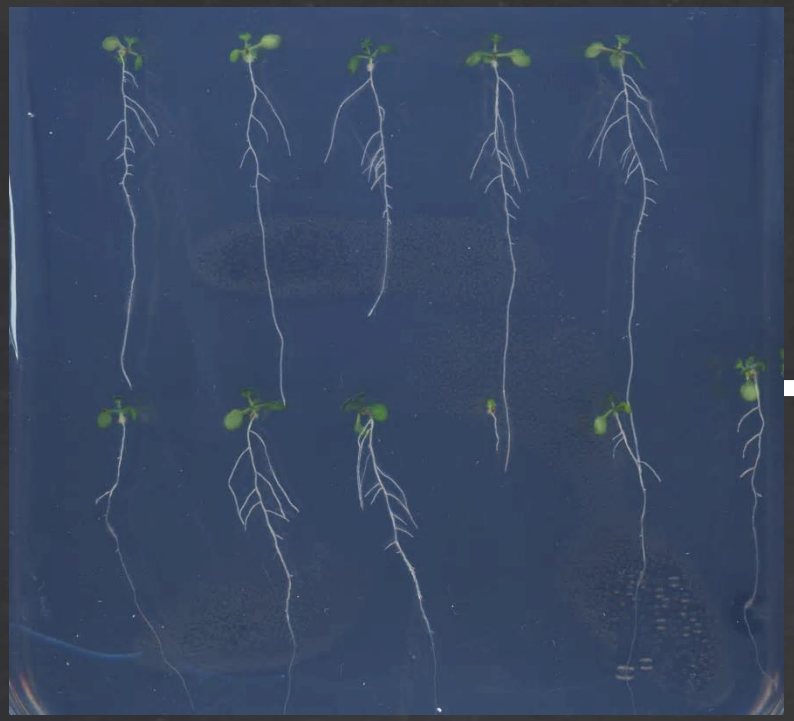


Recognition of plant organs (**machine learning**, not deep learning)





Recognition of plant organs. (**machine learning**, not deep learning)



Raw images or slightly modified images

Feature extraction algorithm

List of features

Recognition by a neural network

Recognition of plant organs. First extracts **hand-crafted image features** (Color/Intensity, Edge, Texture, etc)

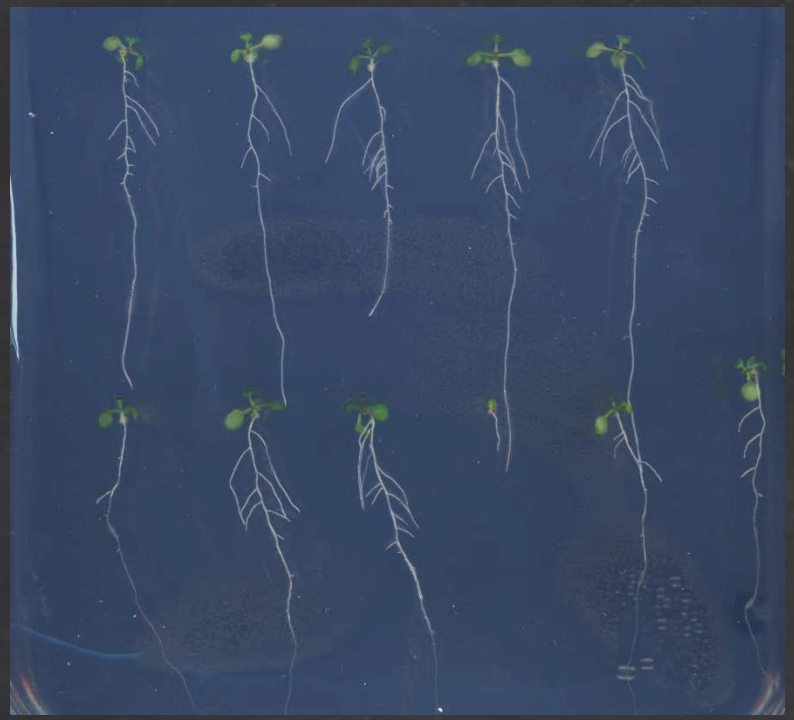
	Color / intensity	Edge	Texture
Leaves	chlorophyll green, brighter veins, darker lamina, patches, color gradients, smooth inner texture	leaf contour, serrated or smooth margin, vein borders, midrib edges, reticulated network edges	reticulated vein network pattern, mesophyll granularity, directional anisotropy from venation
Roots	strong contrast vs background, pale white/yellow roots, thin uniform structures, inner root homogeneity, high signal contrast	fine thin edges, forks, junctions, crossings, broken edges, root branching points	tubular tissue texture, smooth cortex interior, fine micro-texture (root hairs), fiber-like directionality, multiscale branching texture
Plate	uniform intensity, low variance, smooth gradient, glare, shadow zones, light/dark background, homogeneous region	circular border, plate ring edge, sharp transition background/air or background/casing	smooth, isotropic, low structural complexity, background noise sometimes granular but weak directionality
Seeds	brown/cream tone, darker seed coat, uniform inner color, oval colored blob, seed clusters, embryo vs background contrast	elliptical contour, seed coat border, hilum edge, cluster boundaries, clean air/background transition	coarse striated or rough seed coat pattern, repetitive grooves, mottled coat texture, cluster texture when seeds overlap

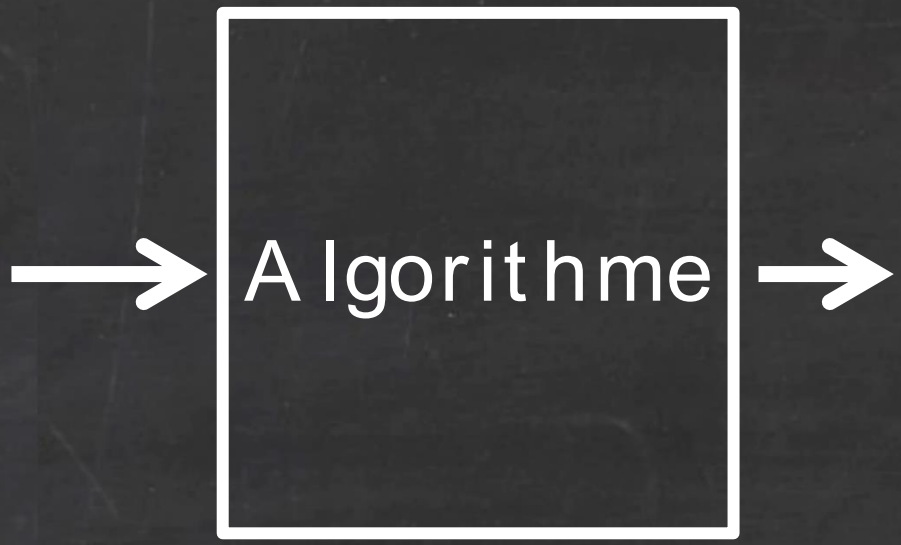


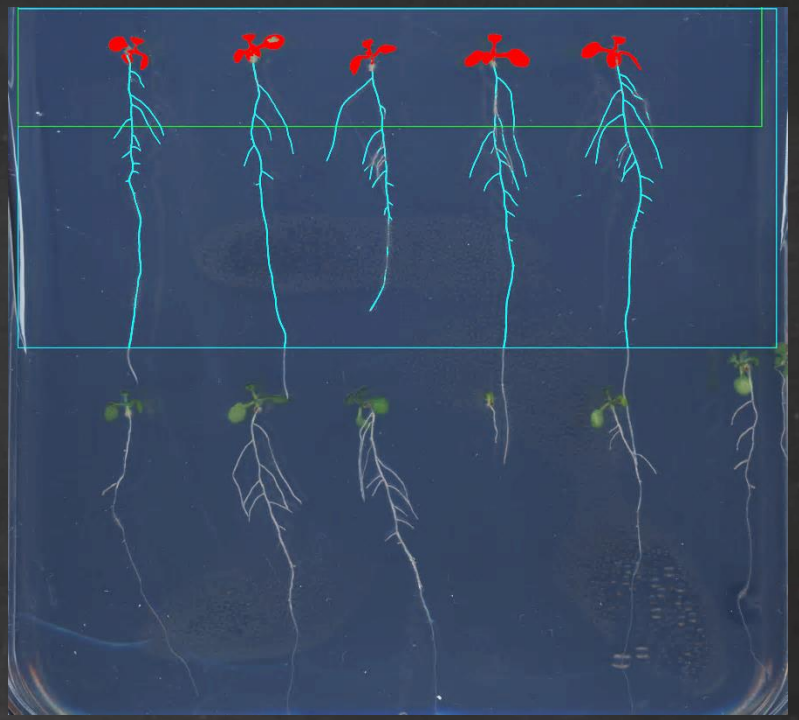
Test with ilastik

- Random Forest classifier based on the pixel labels
- This is classical machine learning, similar to methods used before deep neural networks became mainstream
- Runs fast even without a strong GPU
- Great for interactive segmentation and small datasets

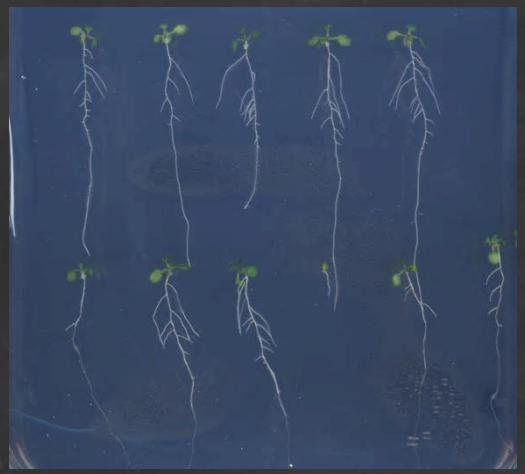
Root branching recognition





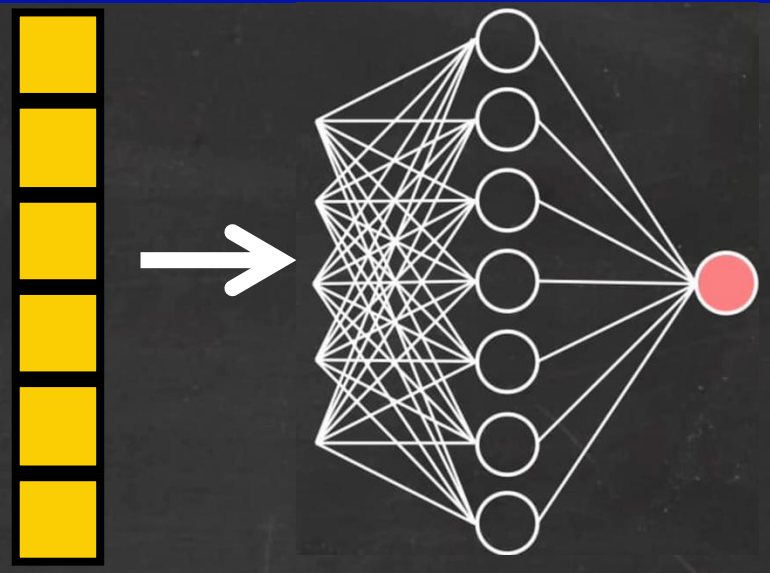


Recognition of plant ramification (**deep learning**, not machine learning)

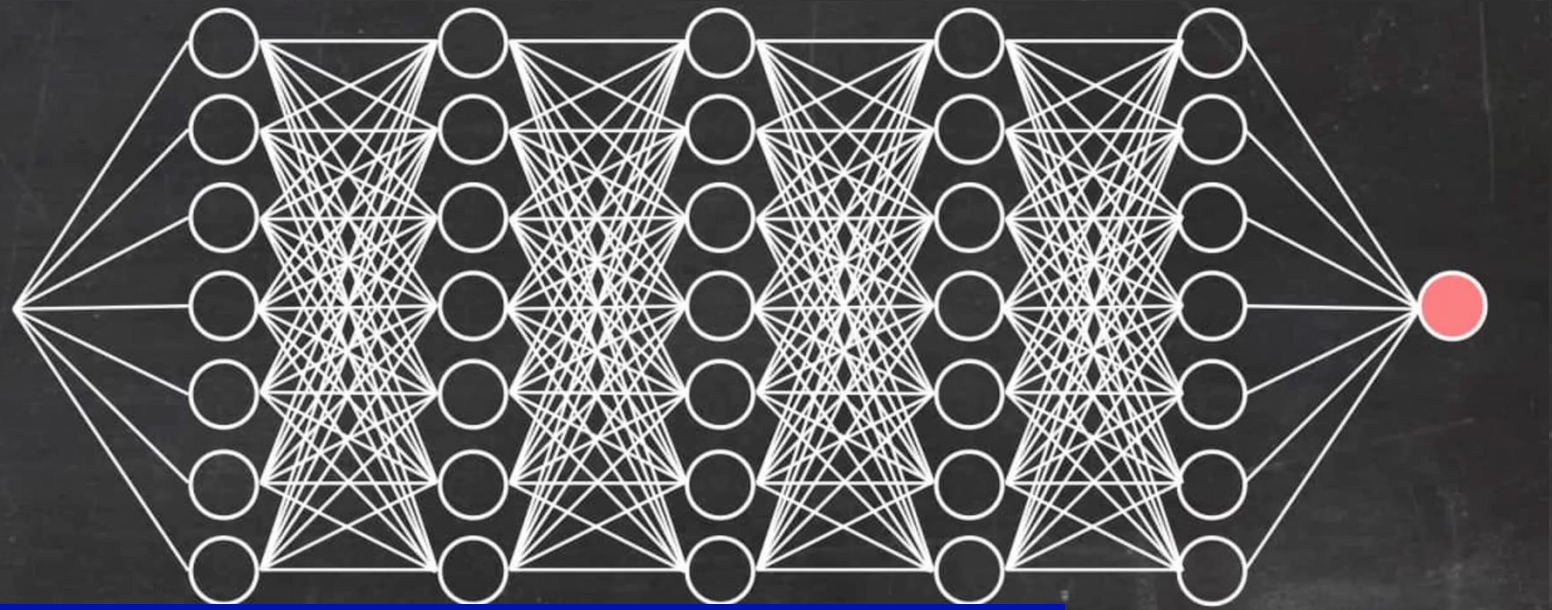




Feature extraction algorithm

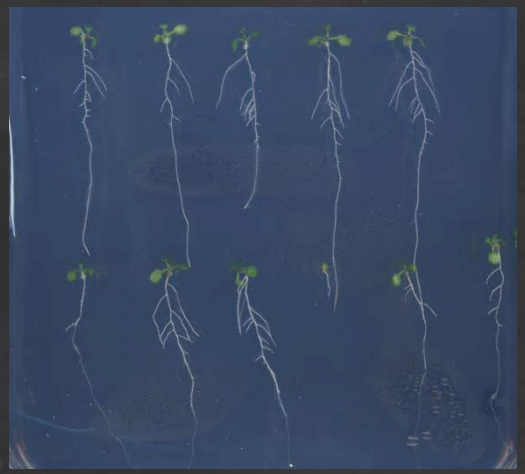


Machine learning



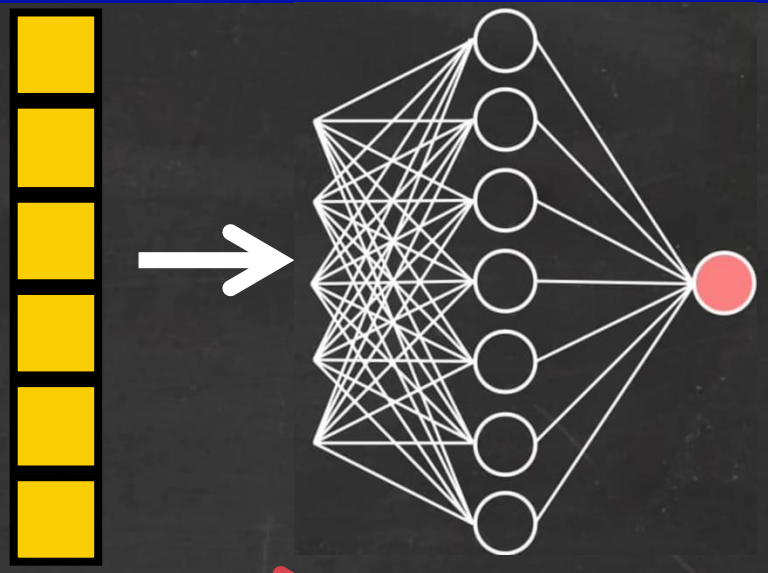
Deep learning

Recognition of plant ramification (**deep learning**, not machine learning)



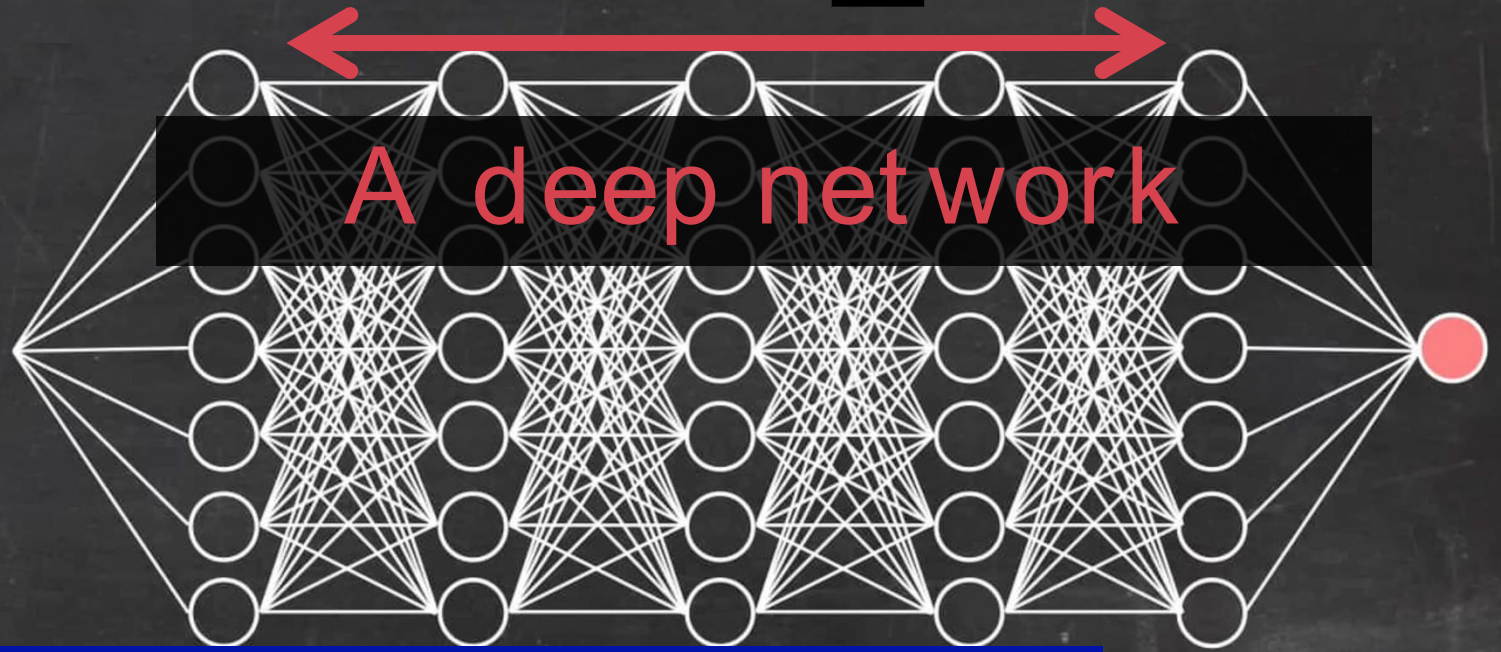


Feature extraction algorithm



Machine learning

A deep network



Deep learning



Yann Le Cun
(1989)

Machine leargning algorithm for image recognition

Large Scale Visual Recognition Challenge

2010	
1. NEC	28%
2. XRCE	34%
3. ISIL	45%
4. UCI	47%
5. Hminmax	54%

↑ Error rate



Yann Le Cun
(1989)

Machine leargning algorithm for image recognition

Large Scale Visual Recognition Challenge

2010	
1. NEC	28%
2. XRCE	34%
3. ISIL	45%
4. UCI	47%
5. Hminmax	54%

2011	
1. XRCE	26%
2. Uv A	31%
3. ISI	36%
4. NII	50%



Yann Le Cun
(1989)

Machine leargning algorithm for image recognition

Deep learning algorithm

Large Scale Visual Recognition Challenge

2010

1.	NEC	28%
2.	XRCE	34%
3.	ISIL	45%
4.	UCI	47%
5.	Hminmax	54%

2011

1.	XRCE	26%
2.	Uv A	31%
3.	ISI	36%
4.	NII	50%

2012

1.	SuperVision	16%
2.	ISI	26%
3.	VGG	27%
4.	XRCE	27%
5.	Uv A	30%



Yann Le Cun
(1989)

Machine leargning algorithm for image recognition

Deep learning algorithm

Large Scale Visual Recognition Challenge

2010

1.	NEC	28%
2.	XRCE	34%
3.	ISIL	45%
4.	UCI	47%
5.	Hminmax	54%

2011

1.	XRCE	26%
2.	Uv A	31%
3.	ISI	36%
4.	NII	50%

2012

1.	SuperVision	16%
2.	ISI	26%
3.	VGG	27%
4.	XRCE	27%
5.	Uv A	30%

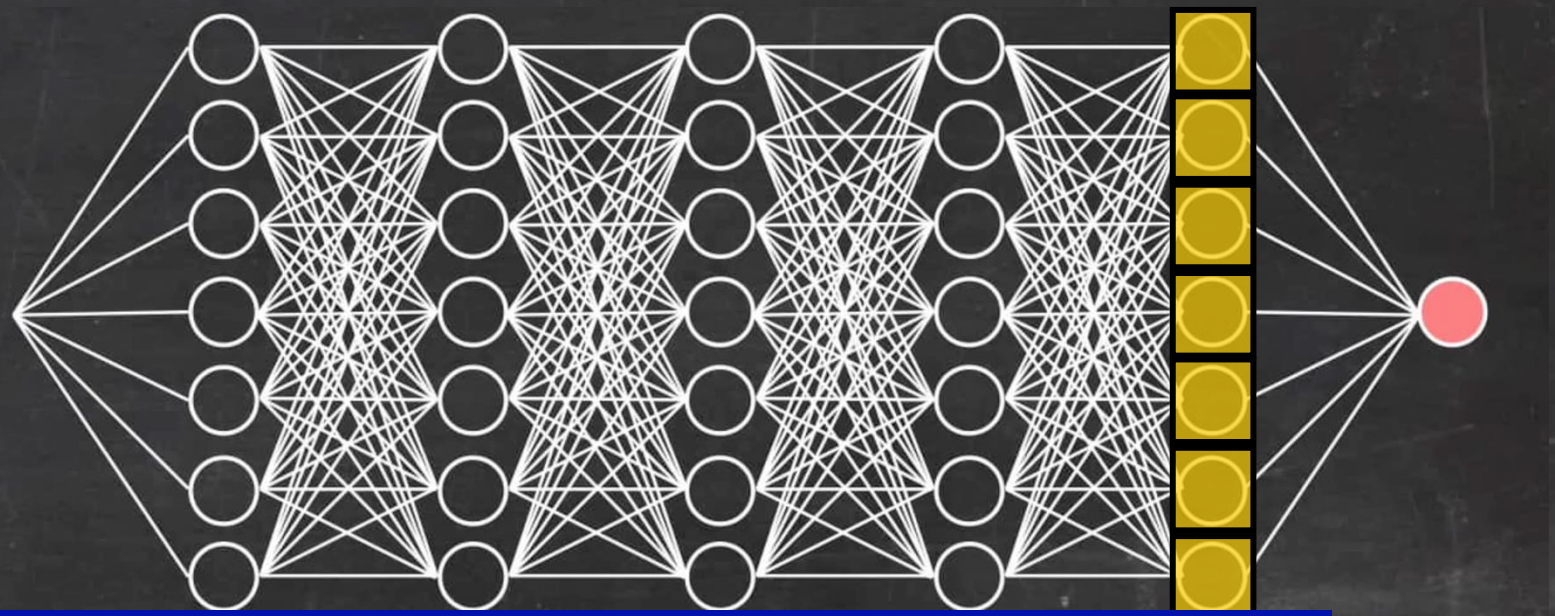
2013

1.	Clarifai	12%
2.	NUS	13%
3.	ZeilerFergus	13%
4.	A.Howard	13%
5.	OverFeat	14%

Why does deep learning finally work?

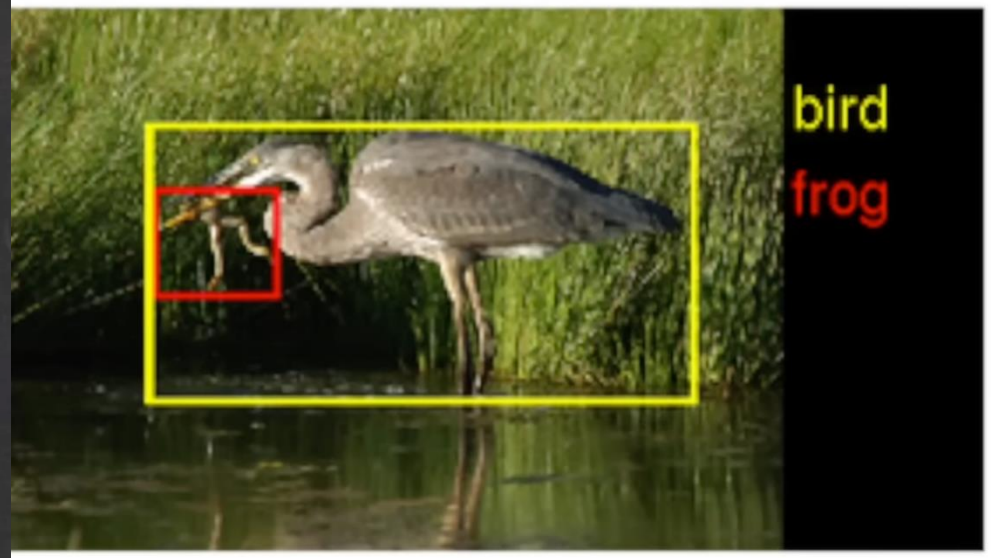
- Better understanding of the importance of network architecture
- Greater computing power than before (GPU)
- More accessible databases (ImageNet)

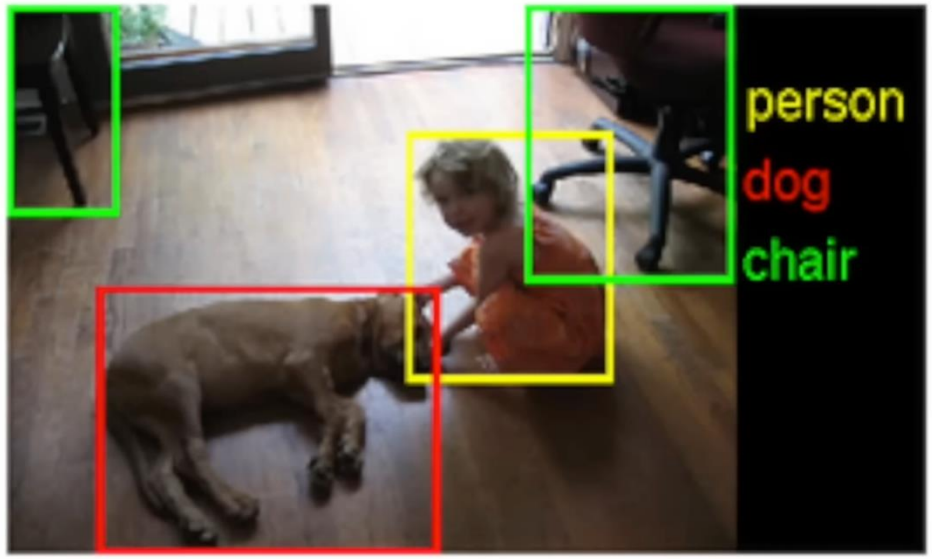


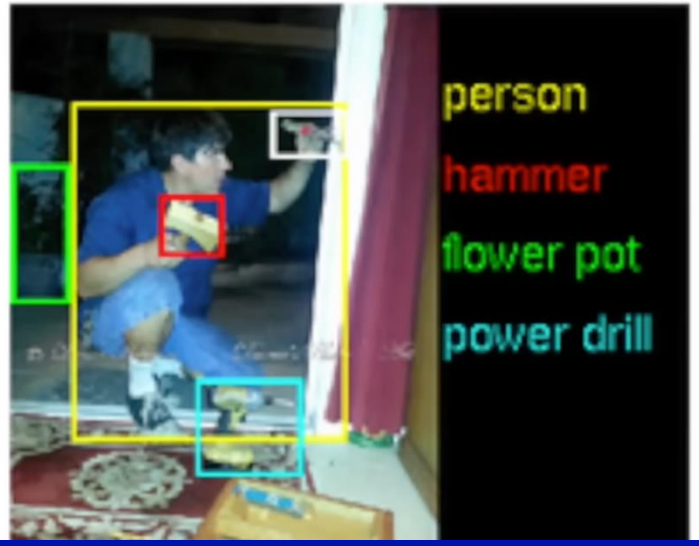


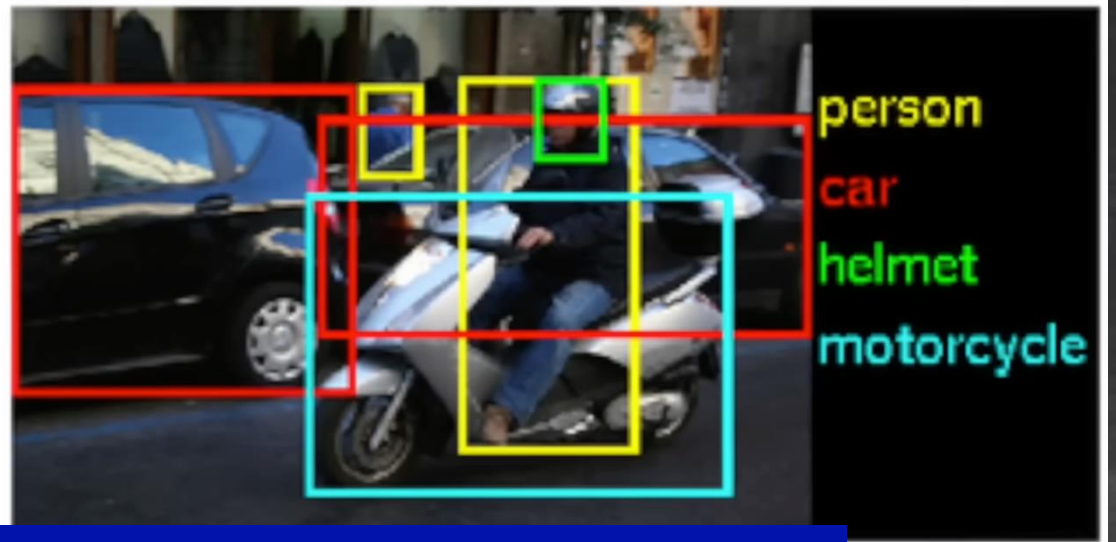
Deep learning

Example of image recognition using Deeplearning. Here YOLO results (high-speed real-time object detection algorithm)









Sometimes a few errors appear…



Hairdryer ?



Camel ?



AnnoTrain (Annotate. Train. Learn. Share)

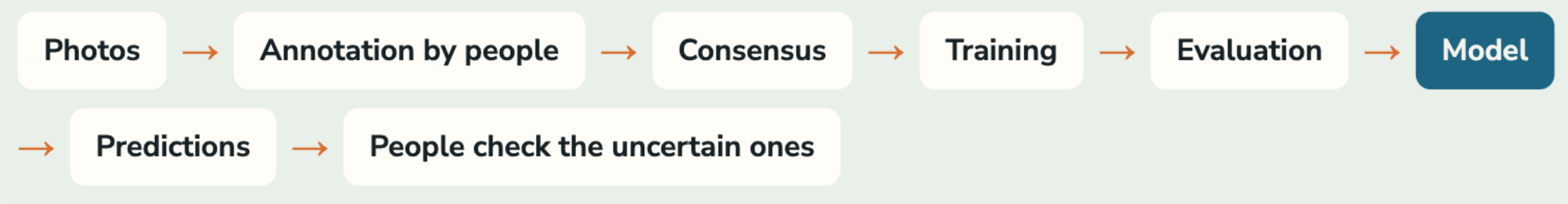
https://annotrain.cmaslard.xyz/

Initial goals¶

- Simple and accessible participatory image annotation
- Scientific dataset creation
- Annotation quality control and consensus
- Deep-learning model training and evaluation
- Human-in-the-loop model improvement
- Interactive visualization of predictions and model errors
- Educational exploration of the complete deep-learning workflow
- Open dataset and model sharing



AnnoTrain: annotate fast, together, and get a model



Classification, detection, segmentation, tracking. **Use it for your Master's project: annotrain.cmaslard.xyz**

What a model can be asked to do

Task	Question	Example in biology
Classification	Which class?	Cat or dog; which amphibian species
Regression	How much?	Leaf area; age of a fish
Detection	Where are the objects?	Each animal in a camera-trap photo
Segmentation	Which pixels are what?	Roots and seeds (DeepPlate); cells
Tracking	Where does each one go?	Fish swimming in a video
Generation	Produce something new	A chatbot's answer; a protein

Chihuahua or muffin?



“Three dark round spots”

Blueberries are dark round spots too.

“Brown and round”

So is a muffin, from above.

“Two eyes above a nose”

Not when the dog turns its head.

Chihuahua or muffin?

When we cannot write the rules, we give examples with the answer, and the computer finds the rules



“Three dark round spots”

Blueberries are dark round spots too.

“Brown and round”

So is a muffin, from above.

“Two eyes above a nose”

Not when the dog turns its head.

Chihuahua or muffin? Annotate photos: as many as you can, carefully

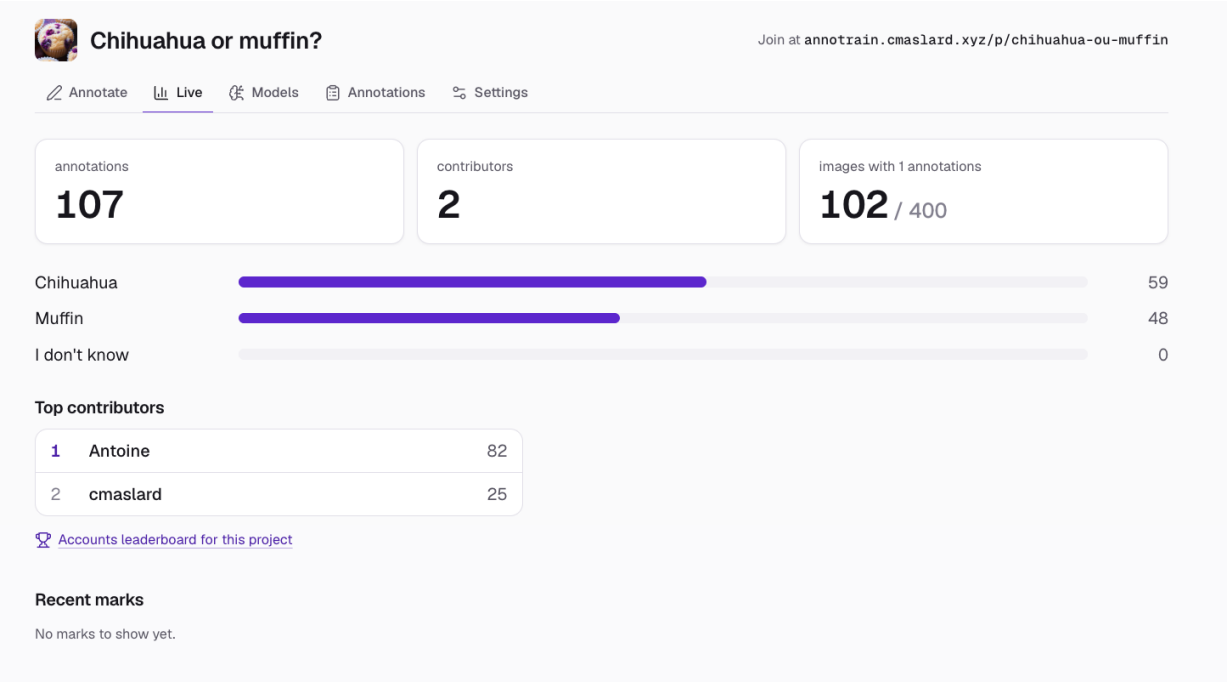


- On your phone or your laptop
- No account needed
- Answers given too fast do not count

Congratulations: you just built a training dataset

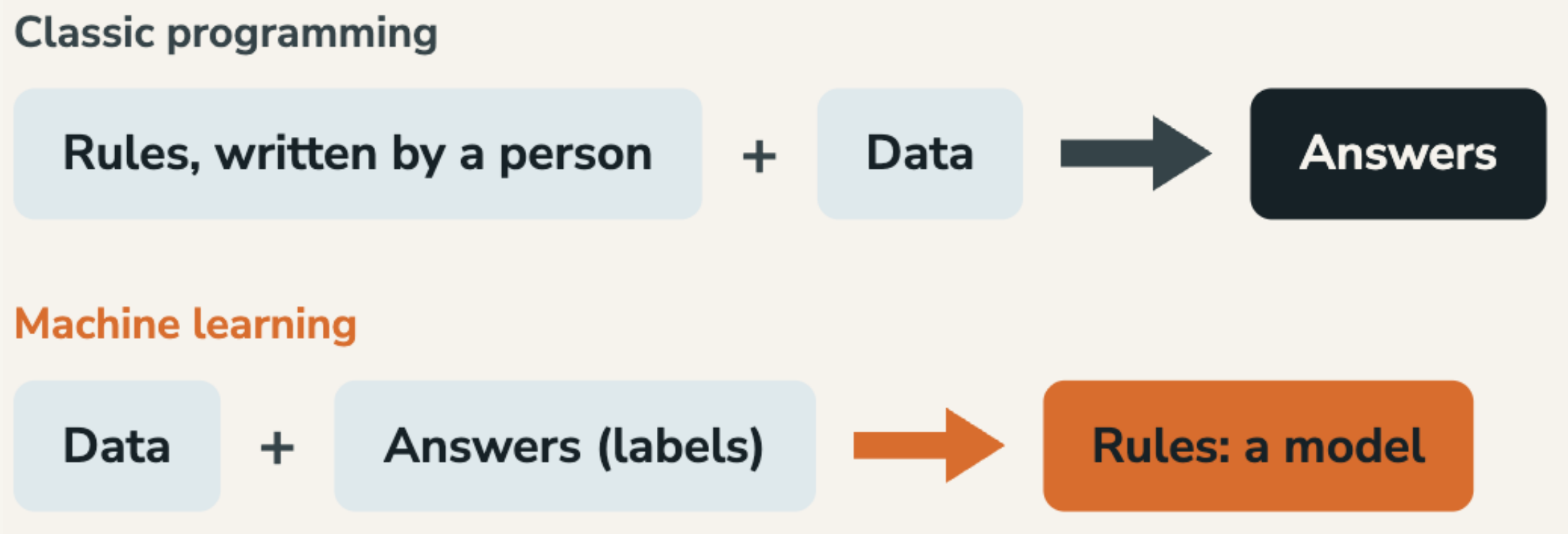
Each click is a **label**. Several of you saw the same photo: the majority is the **consensus**.

A neural network is learning from your clicks right now. You will see it in Theory 3.



https://annotrain.cmaslard.xyz/p/chihuahua-ou-muffin/live

Rules written by people, or rules learned from examples



Three ways to learn

Supervised

Examples come with the answer, the **label**: photo → “toad”.

Today.

Unsupervised

No label: the model finds structure by itself.

The PCA of Session 5.

Self-supervised

Hide part of the data and guess it: a word, an amino acid.

Chatbots, protein language models.

The words of supervised learning

https://penguin-place.co.nz/

Colonne (anglais)	Traduction (français)	Unité
bill_len	Longueur du bec	mm
bill_dep	Profondeur du bec (épaisseur)	mm
flipper_len	Longueur de la nageoire	mm
body_mass	Masse corporelle	g
species	Espèce de manchot	—



Features: what the model looks at				Label: to predict
bill_len	bill_dep	flipper_len	body_mass	species
39.1	18.7	181	3750	Adelie
46.5	17.9	192	3500	Chinstrap
46.1	13.2	211	4500	Gentoo

Model: the recipe features → label. Its **parameters** (weights) are learned.

Training set to learn, **validation set** to tune, **test set** to judge.

Inference: using the trained model on new data.

THE GOLDEN RULE

A model is only judged on data it has never seen !!!

The test set is put aside before training, and used once, at the end. The same idea as the cross-validation of Session 5.

The confusion matrix: 100 camera-trap photos, 30 with an animal

Truth	Predicted: animal	Predicted: empty
animal	24 found	6 missed
empty	4 invented	66 right

Rows: the truth. Columns: what the model said.

Accuracy = (24 + 66) / 100 = 0.90

Share of correct answers.

Recall = 24 / 30 = 0.80

Of the true animals: how many found? (row)

Precision = 24 / 28 = 0.86

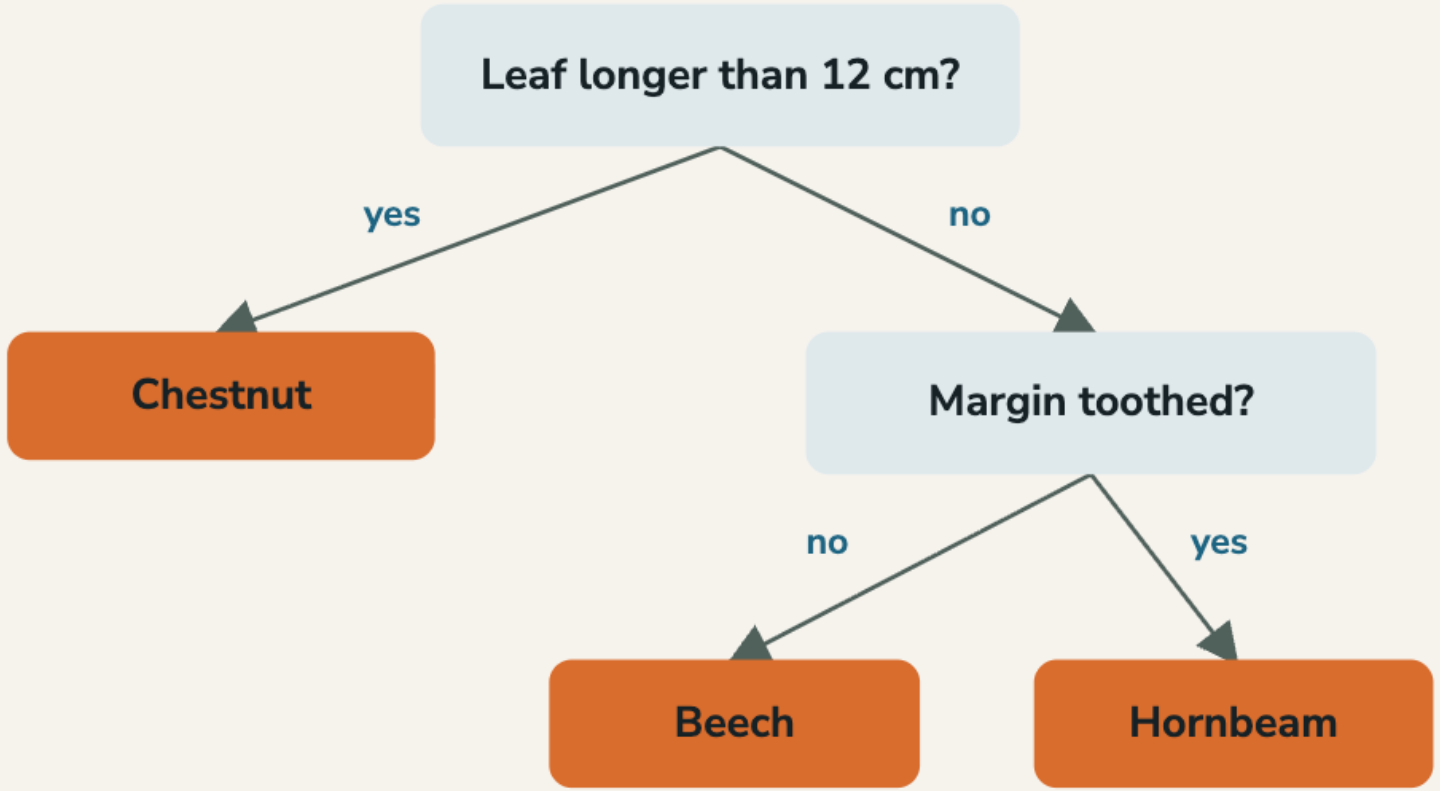
When it says animal, is it right? (column)

The accuracy trap



A model that never looks at the photos reaches 70% accuracy, and **finds no animal at all**. Always look at the recall of each class, especially the rare ones.

A decision tree is an identification key, learned from examples

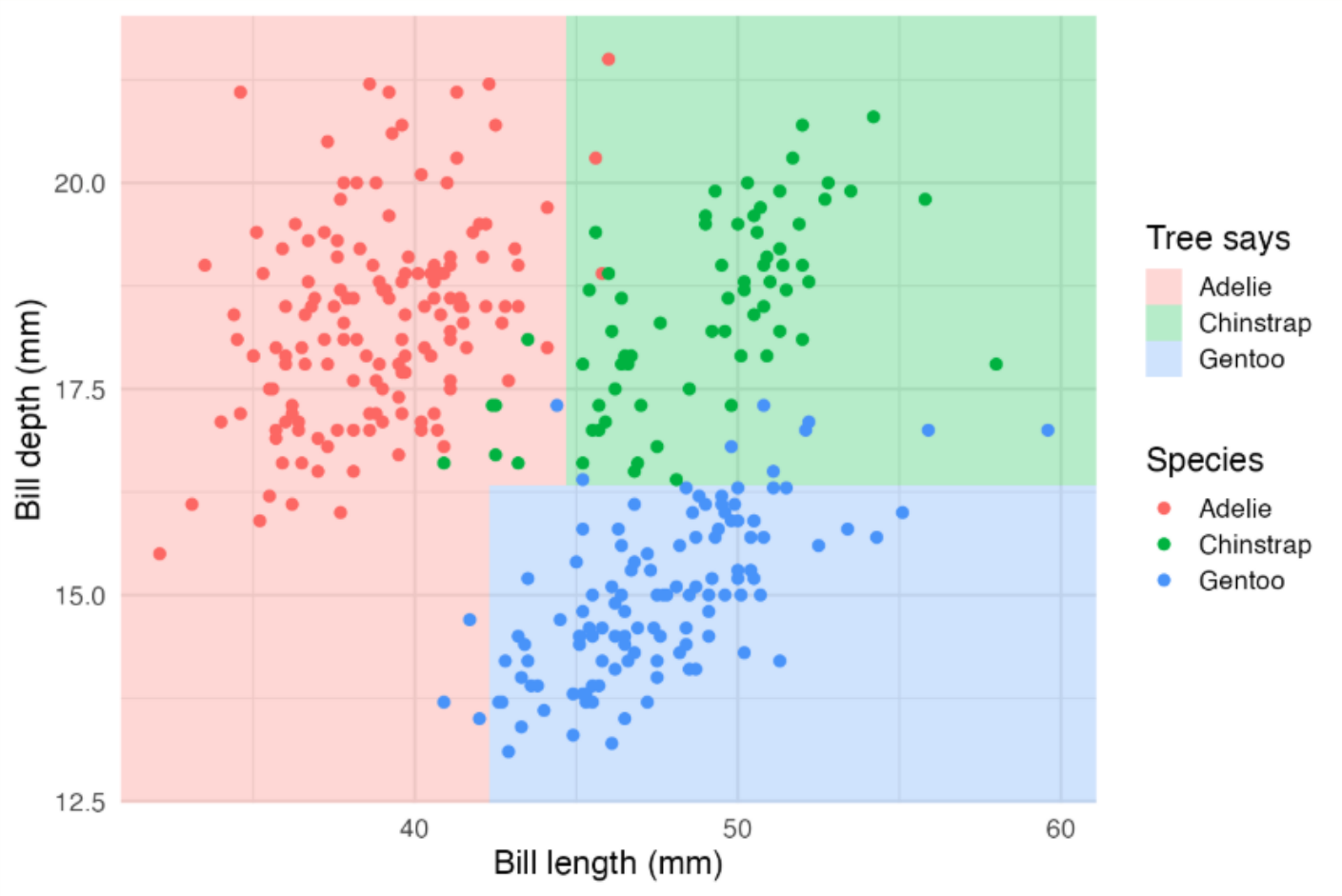


The computer tries every feature and every threshold, and keeps the question that best separates the labels.

Then it does the same in each half, again and again.

Easy to read: its great strength.

Each question cuts the plane: a tree draws boxes

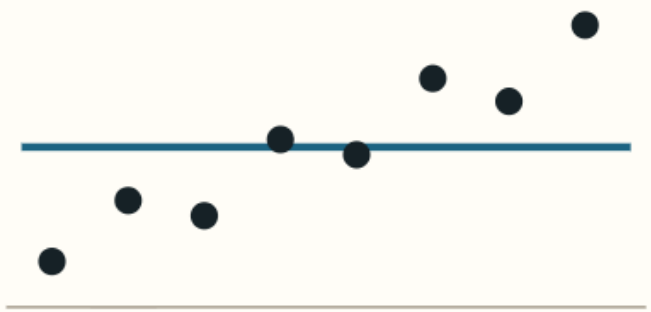


The penguins of the practical, by bill length and bill depth. The shaded boxes are what a tree answers.

Points in the wrong box are its mistakes.

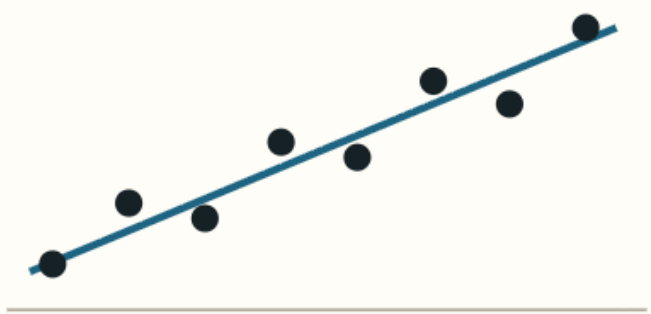
In the workbook, the tree uses all four measures.

Overfitting: learning the examples by heart



Underfitting

Too simple: misses the rule.



Good fit

Learns the rule: as good on new data.



Overfitting

Passes through every point: 100% on training data, poor on new data.

The signature: **very good on the training set, much worse on the test set**. Remember the shuffled labels of Session 5

A random forest: many different trees that vote

1

Draw

Each tree gets a bootstrap sample (drawn again, with replacement) and a few random features.

2

Grow

Hundreds of trees, each a little different, each wrong sometimes.

3

Vote

The majority wins: mistakes that are not the same cancel out.

The same idea as **several people annotating the same photo** in AnnoTrain.

Breiman (2001), Random forests, Machine Learning. The pixel classifier of ilastik is a random forest.

A neuron is the linear model of Session 4, bent

$$\text{output} = f(\ w_1 \cdot x_1 + w_2 \cdot x_2 + w_3 \cdot x_3 + b\ )$$

Inside f

Exactly lm(): the weights w are slopes, b is the intercept.

The activation f

Bends it. ReLU keeps positive values; sigmoid squeezes between 0 and 1.

A network

Neurons side by side make a layer; layers stack. It can draw any boundary.

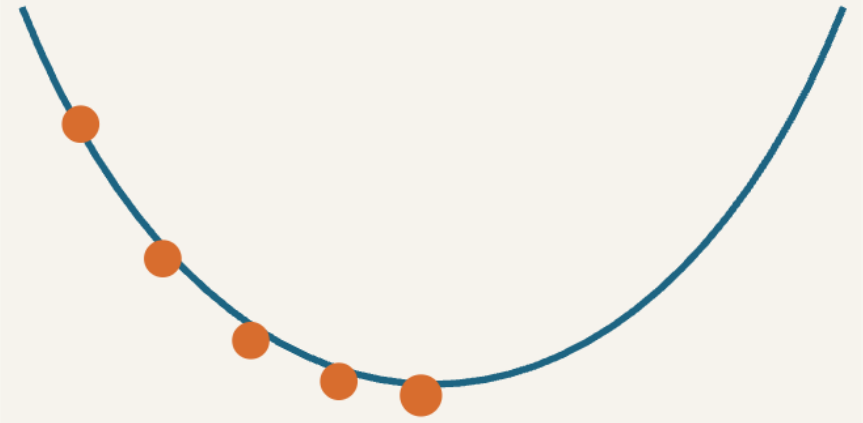
Training: walking downhill in the fog

The **loss** is the error on the training examples: the height of the hill.

Gradient descent: measure the slope for each weight, move every weight a small step downhill. Repeat.

One pass over all the examples is an **epoch**.

Try it: https://playground.tensorflow.org/





Deep learning for images

A photo is numbers; networks learn their own features

A photo is a table of numbers



2,048 × 1,494 pixels

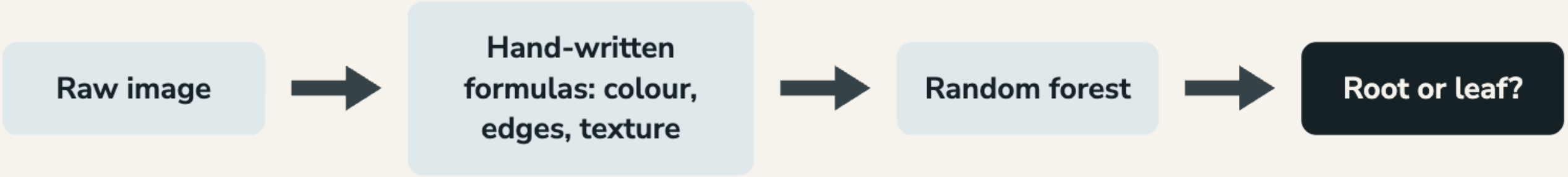
× 3 colours

= 9.2 million numbers

"Is pixel 15,200 brighter than 120?" is a useless question: the deer can be anywhere, at any size, in any light. What matters are **shapes**.

Photo: Caltech Camera Traps (Beery et al. 2018), CDLA-Permissive 1.0, via AnnoTrain.

Before deep learning: features designed by people



Works

On simple, controlled images: seedlings on a plate, cells on a slide (ilastik).

Fails

On the variety of the real world: light, angle, background, a deer half hidden.

A convolution: a 3 × 3 filter slides over the photo

-1	0	1
-1	0	1
-1	0	1

Multiply the 9 pixels under the filter by its 9 numbers, add them up, move one pixel, repeat. The result is **large where the photo turns from dark to light**: this filter finds vertical edges.

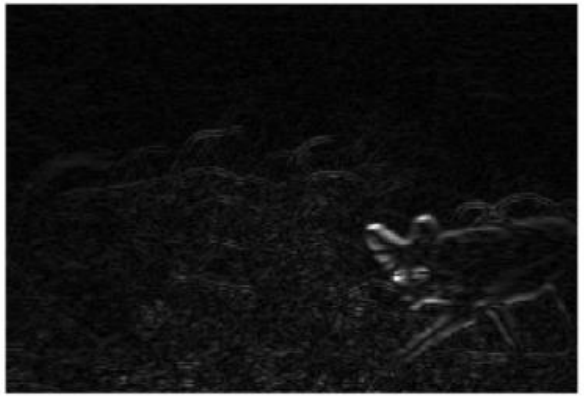
1. The photo: numbers



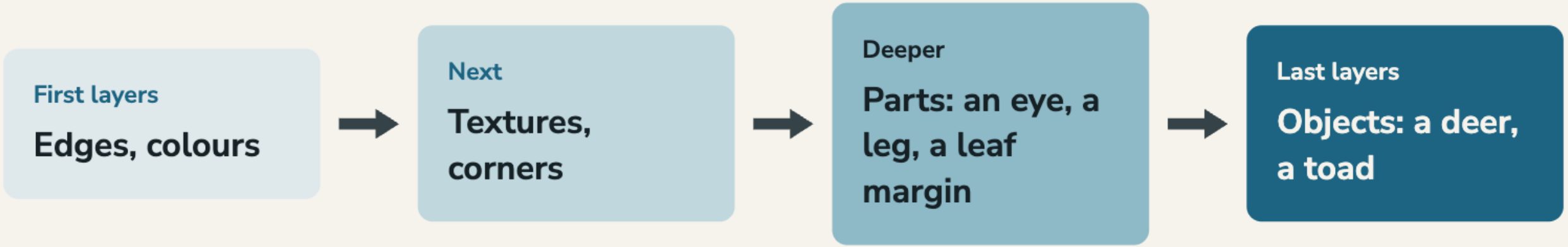
2. Filter: vertical edges



3. Filter: horizontal edges



The filters are weights: the network learns them



Nobody writes the features any more: they are learned from annotated examples. That is a **convolutional neural network** (CNN).

See them: Olah, Mordvintsev and Schubert (2017), Feature Visualization, distill.pub (CC BY).

Interactive: poloclub.github.io/cnn-explainer

Why it took off around 2012

14 M

Data

ImageNet: 14 million photos, annotated by people online.

GPU

Computing

Graphics cards: millions of multiplications at once.

152

Architectures

AlexNet wins ImageNet in 2012; ResNet reaches 152 layers in 2015.

AnnoTrain uses a **ResNet-18**: 18 layers, about 11 million weights.

ImageNet Classification with Deep Convolutional Neural Networks



Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

Ilya Sutskever
Un[redacted]to
ilya[redacted].ca



Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto



We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implemen-

Deep learning with U-Net

How U-Net works:

2 key ideas behind U-Net:

1. **Understand the image context (where roots, leaves, seeds are)**
2. **Locate fine details precisely (thin root pixels, tissue boundaries)**

Unlike classical methods, U-Net:
learns its own features automatically from raw pixels
does *not rely on the user defining edge or texture manually*

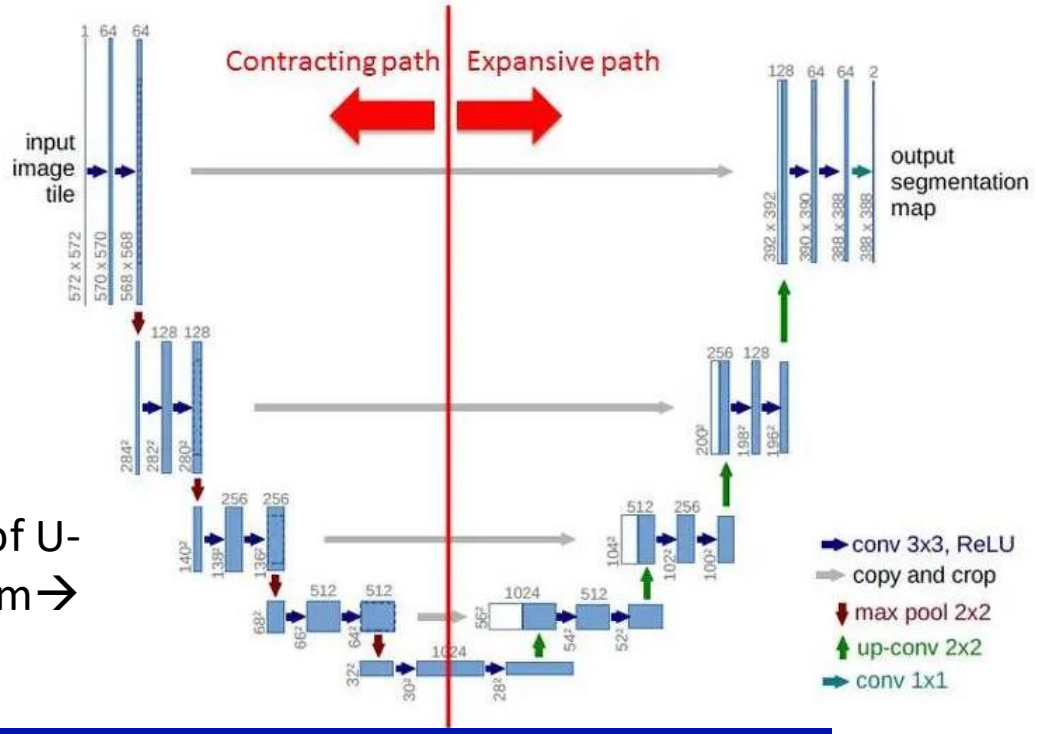
Unlike classical ML, U-Net learns its own features directly from pixel data through backpropagation

Training workflow

Typical steps when using a U-Net model:

- Input images
- Marked the areas that are considered to be ramifications
- Neural network learns to map image → mask
- Output = segmentation of new images

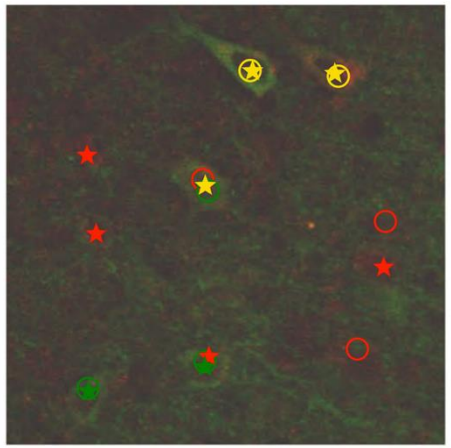
Network architecture of U-Net algorithm→



nature methods
BRIEF COMMUNICATION
https://doi.org/10.1038/s41592-018-0261-2

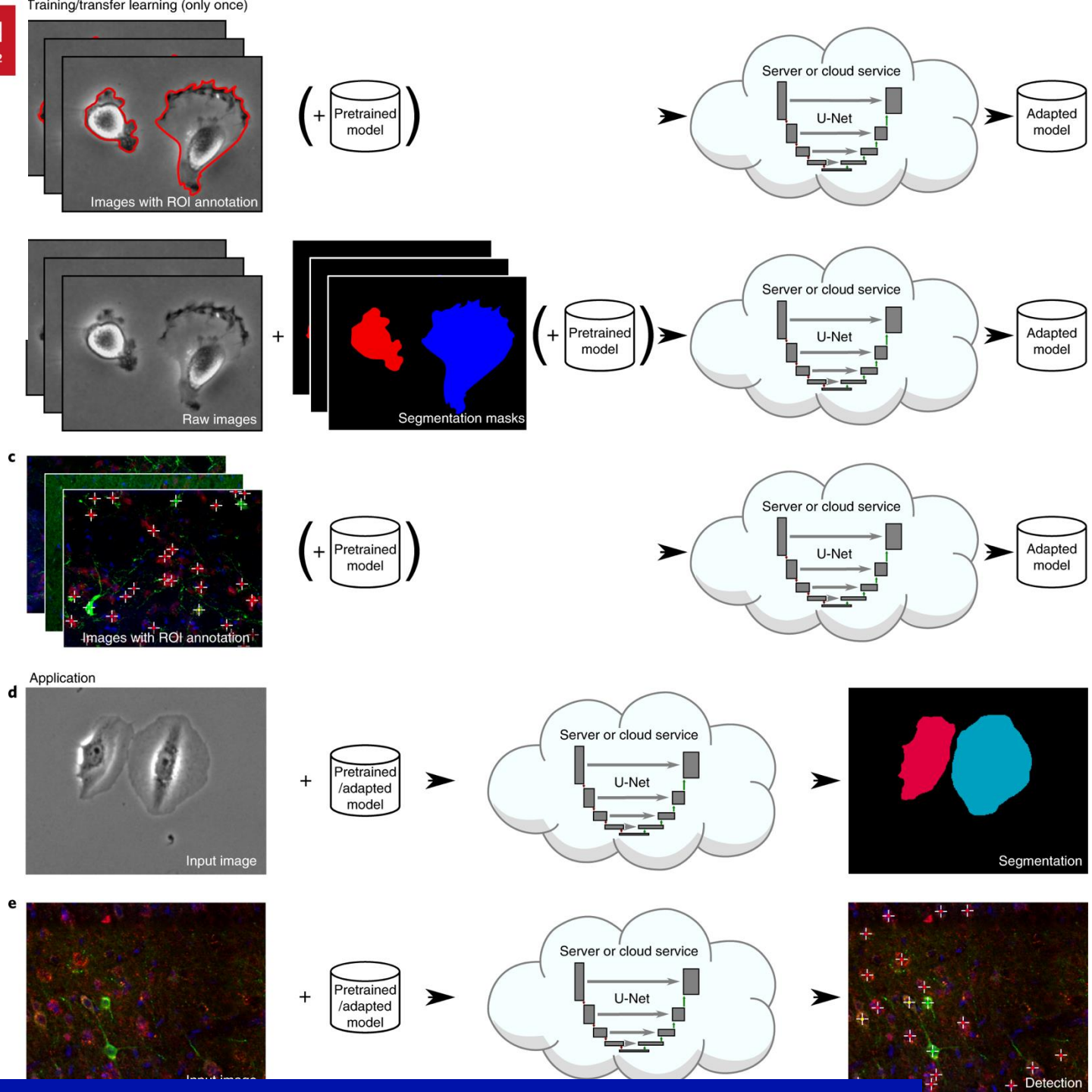
U-Net: deep learning for cell counting, detection, and morphometry

Thorsten Falk[1,2,3,19], Dominic Mai[1,2,4,15,19], Robert Bensch[1,2,16,19], Özgün Çiçek[1], Ahmed Abdulkadir[1,5], Yassine Marrakchi[1,2,3], Anton Böhm[1], Jan Deubner[6,7], Zoe Jäckel[6,7], Katharina Seiwald[6], Alexander Dovzhenko[8,17], Olaf Tietz[8,17], Cristina Dal Bosco[8], Sean Walsh[8,17], Deniz Saltukoglu[2,4,9,10], Tuan Leng Tay[7,11,12], Marco Prinz[2,3,11], Klaus Palme[2,8], Matias Simons[2,4,9,13], Ilka Diester[6,7,14], Thomas Brox[1,2,3,7] and Olaf Ronneberger[1,2,18]*



○ Human expert
★ U-Net

— Antibody only
— eYFP only
— Colocalization



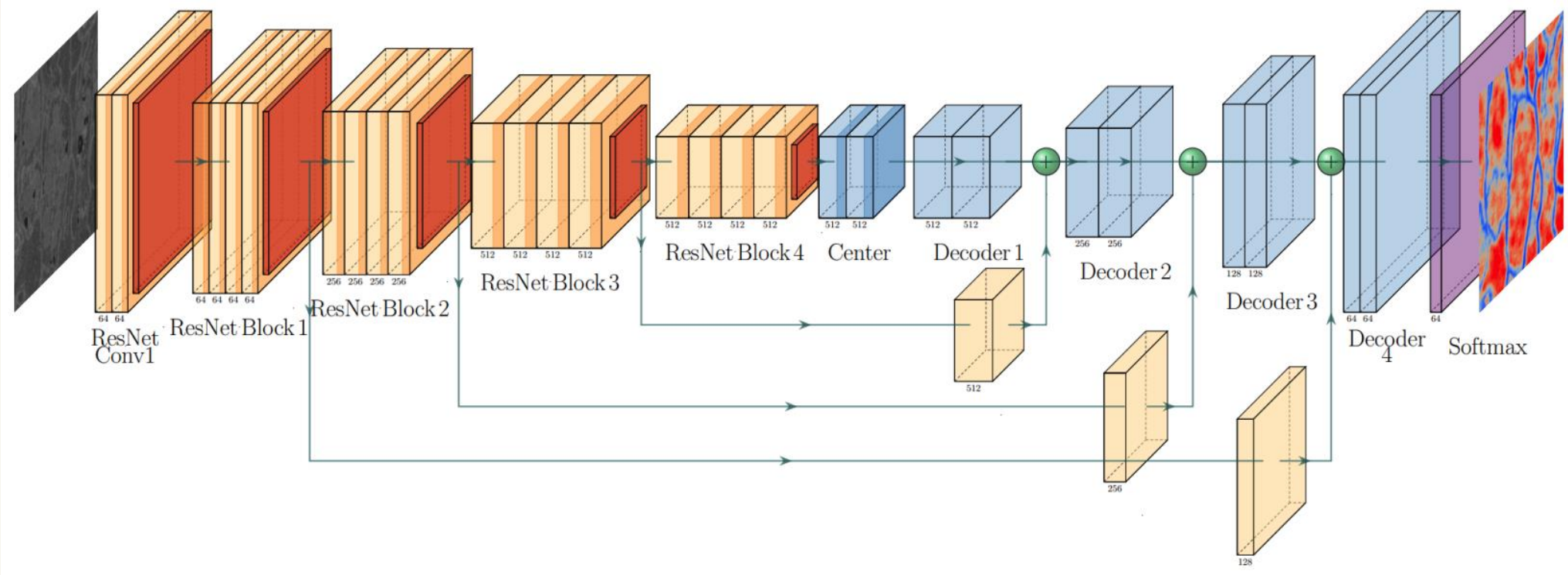
Transfer learning: you do not start from zero (Exemple of ResNet18 – Unet)

Pre-trained on ImageNet

Millions of everyday photos. Its first layers already know edges, textures and shapes.

Fine-tuned on your photos

A few hundred to a few thousand annotations of your own question.



How AnnoTrain, Teachable Machine and most biology tools work.

A model is only as good as its labels

Ground truth

People make mistakes and disagree. Several answers per photo; the majority is the consensus. Never perfectly true.

Class imbalance

A few common species, many rare ones with almost no examples. The rare ones are learned badly.

Domain shift

New cameras, new places, a new microscope: the model is often much worse.

Beery, Van Horn and Perona (2018), Recognition in Terra Incognita, ECCV. Citizen science: Zooniverse, iNaturalist.

Deep learning to find plant branching patterns with U-Net

Scan



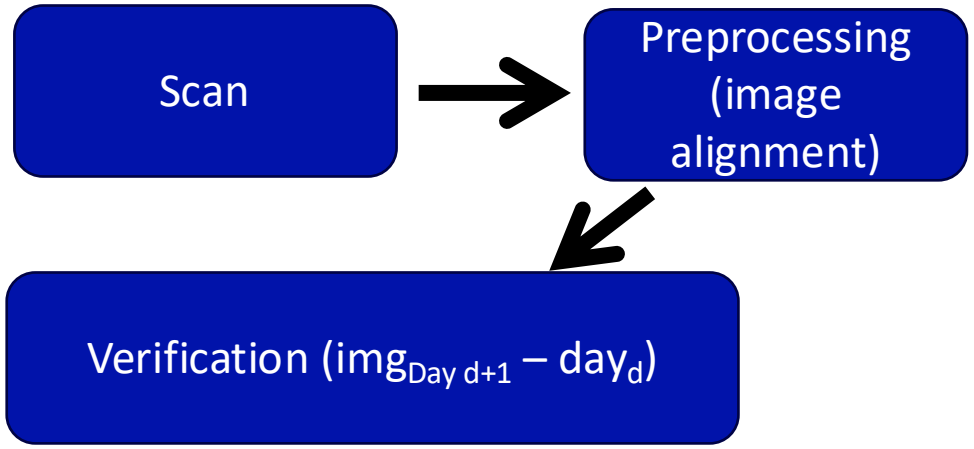
Deep learning to find plant branching patterns with U-Net

Scan → Preprocessing (image alignment)





Deep learning to find plant branching patterns with U-Net





Deep learning to find plant branching patterns with U-Net

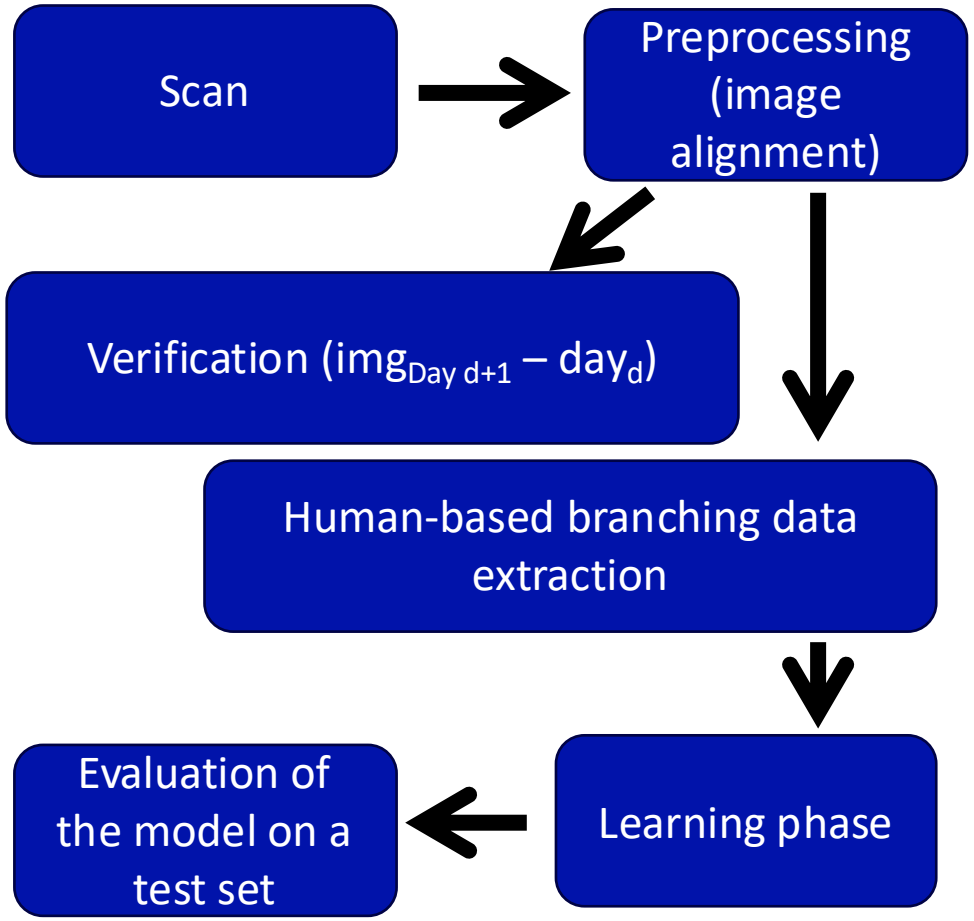
Scan → Preprocessing (image alignment)

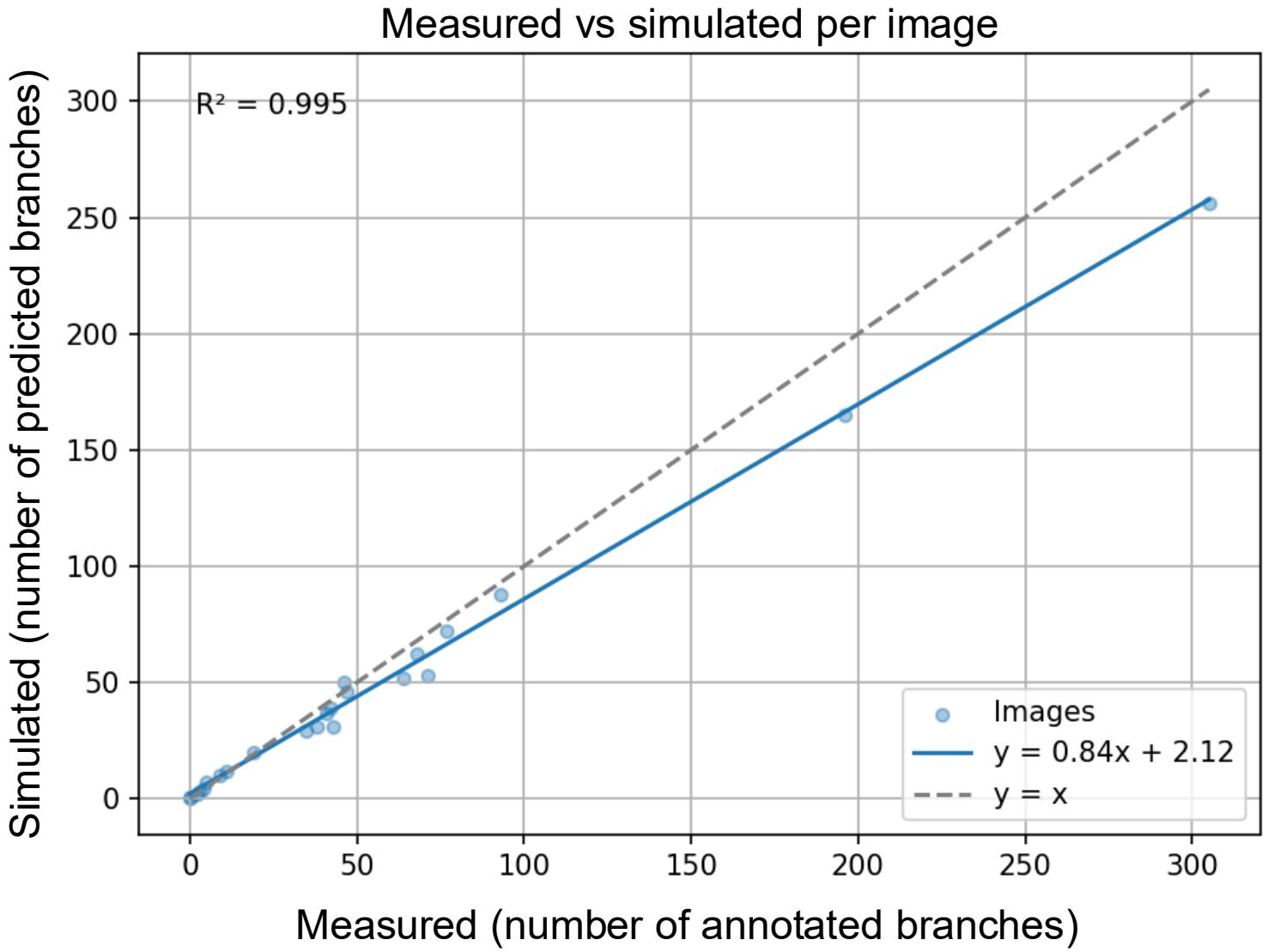
Preprocessing (image alignment) → Verification ($img_{Day\ d+1} - day_d$)

Preprocessing (image alignment) → Human-based branching data extraction

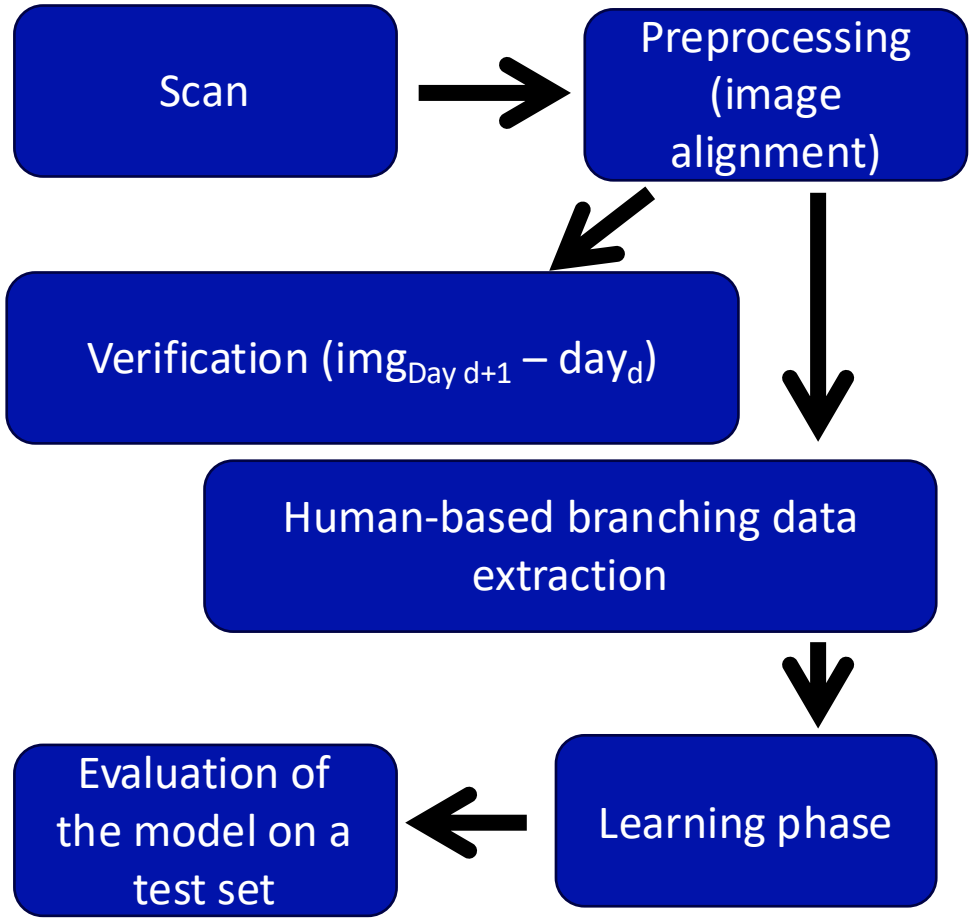


Deep learning to find plant branching patterns with U-Net



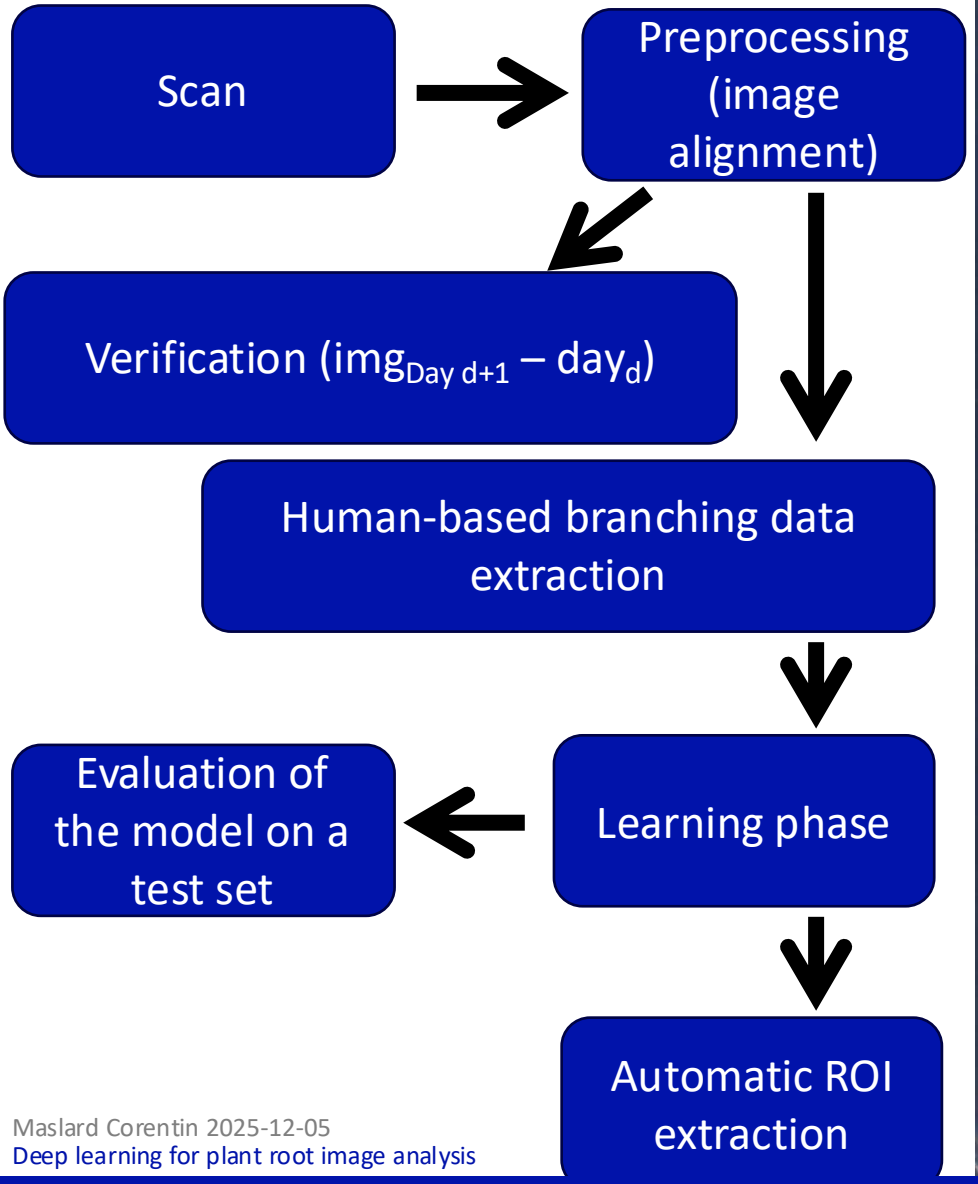


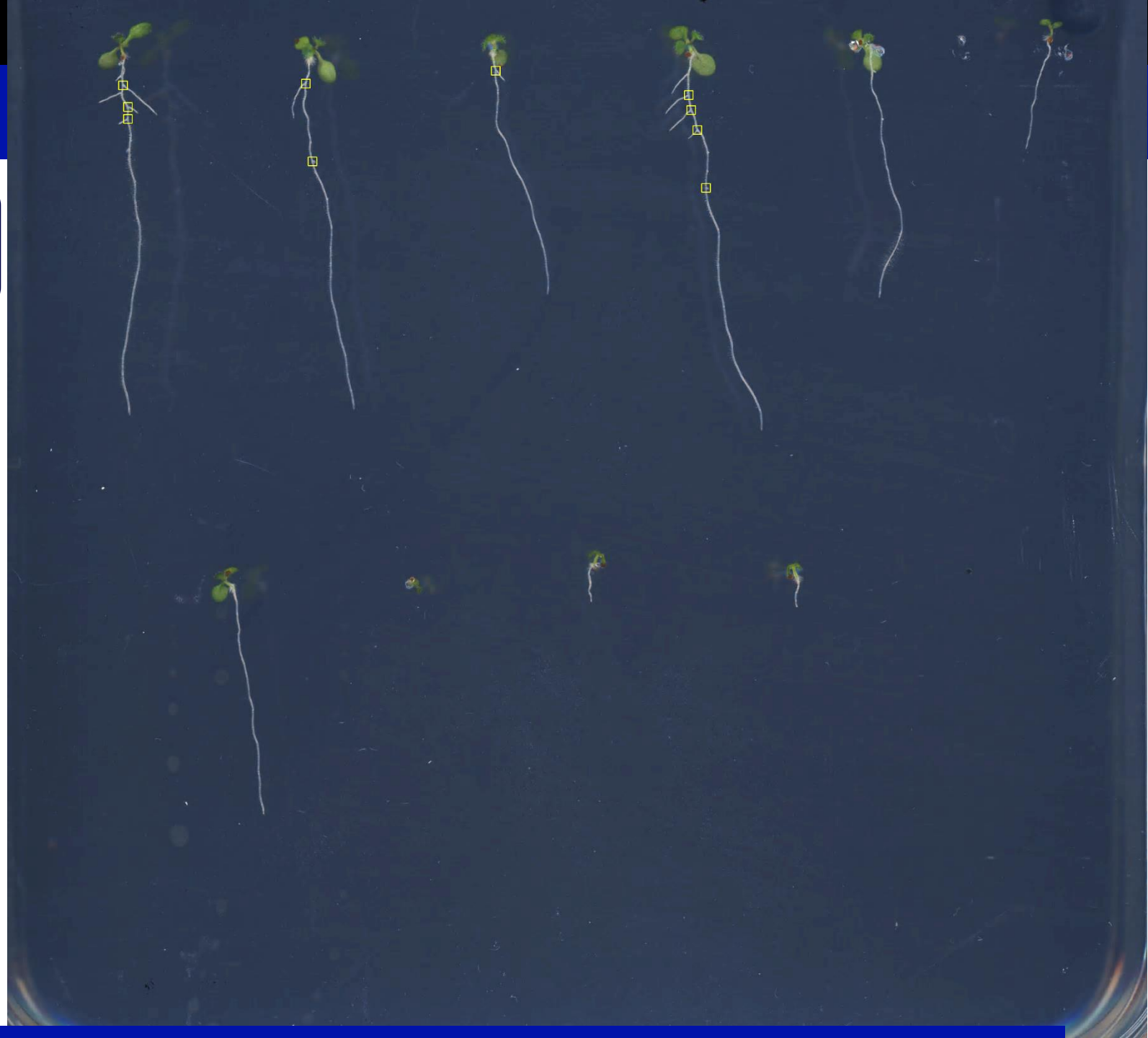
Deep learning to find plant branching patterns with U-Net



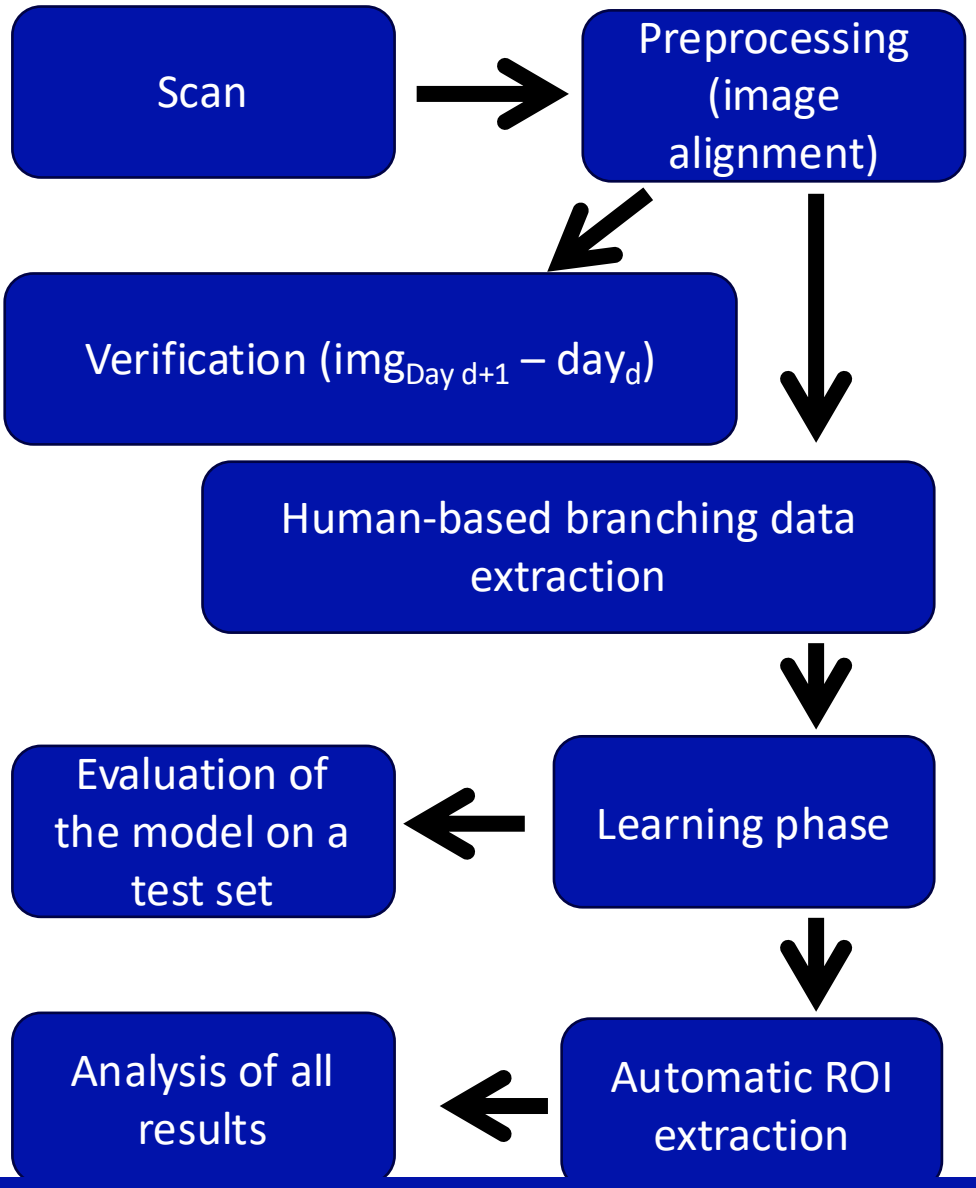


Deep learning to find plant branching patterns with U-Net

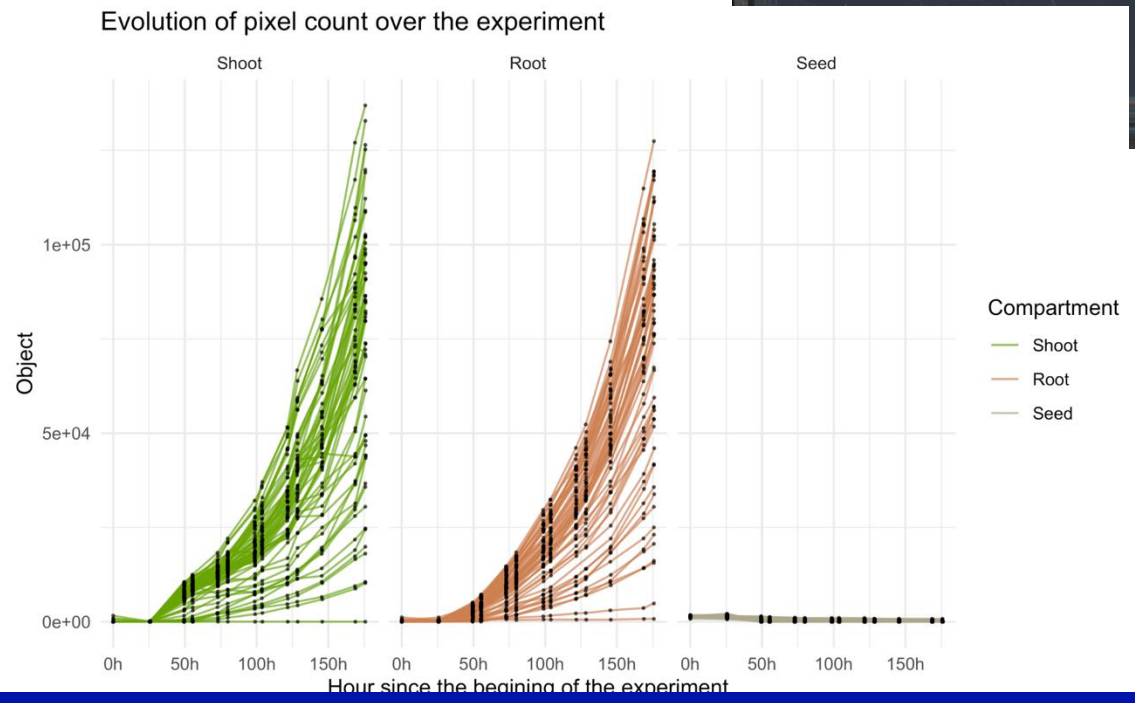




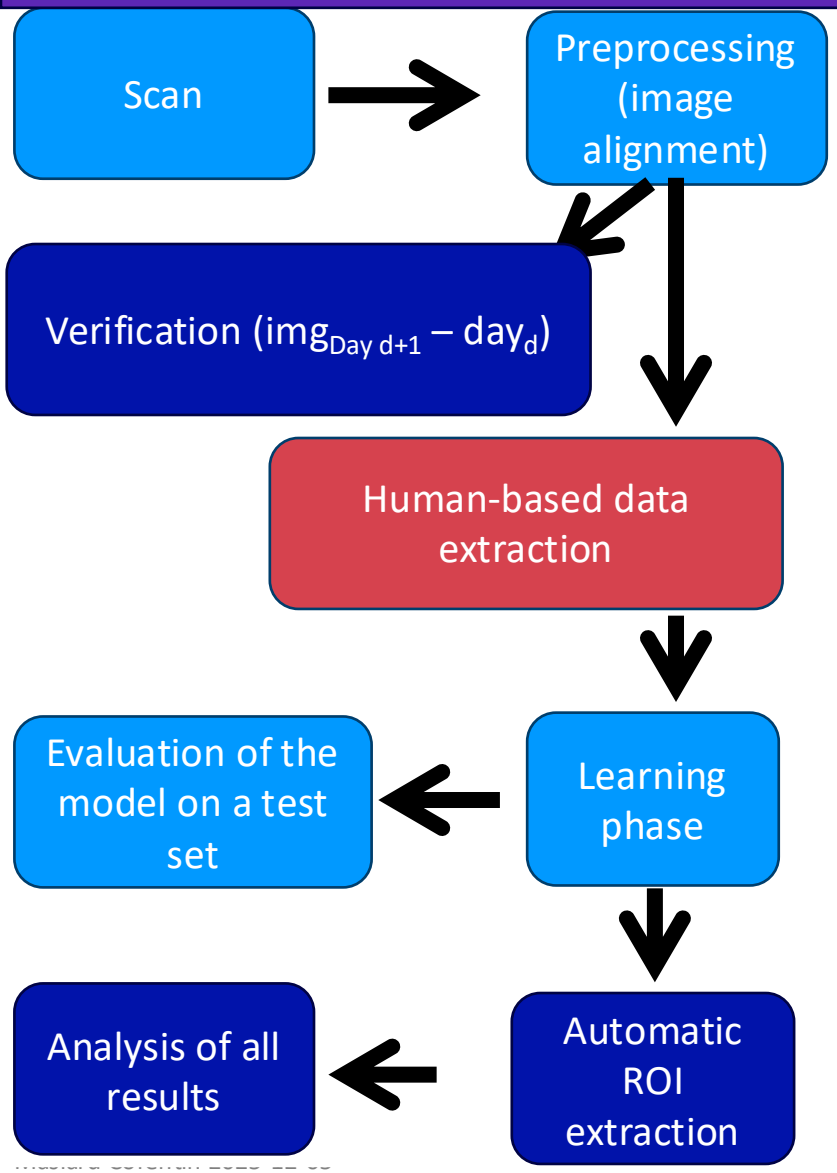
Deep learning to find plant branching patterns with U-Net



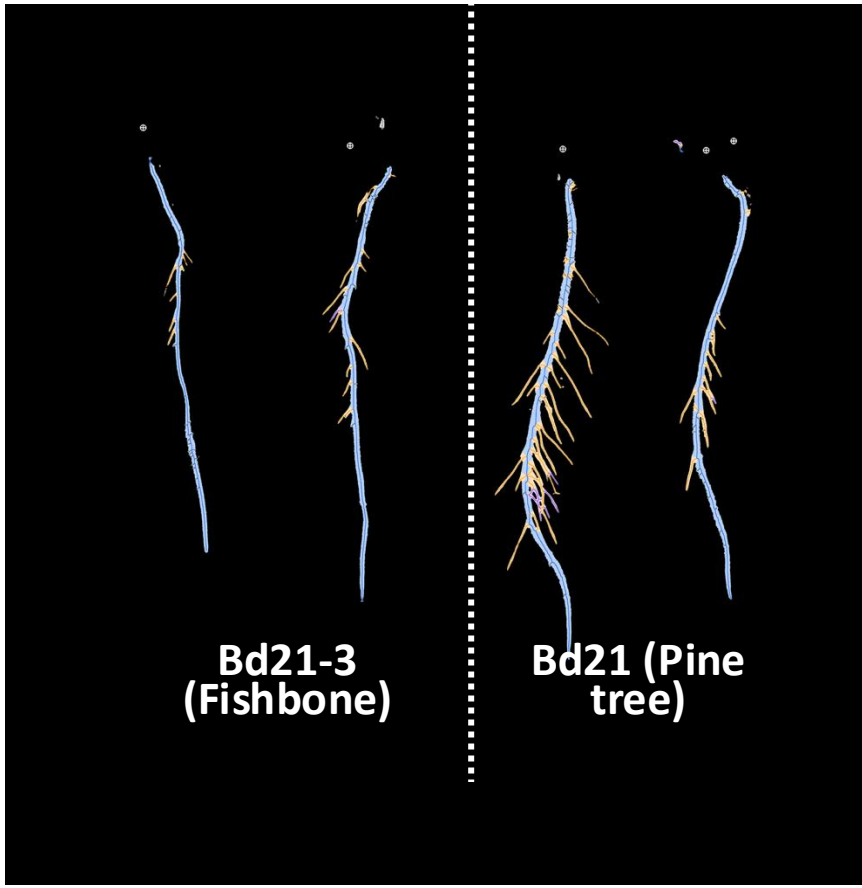




High-throughput phenotyping could capture accession-specific stress responses

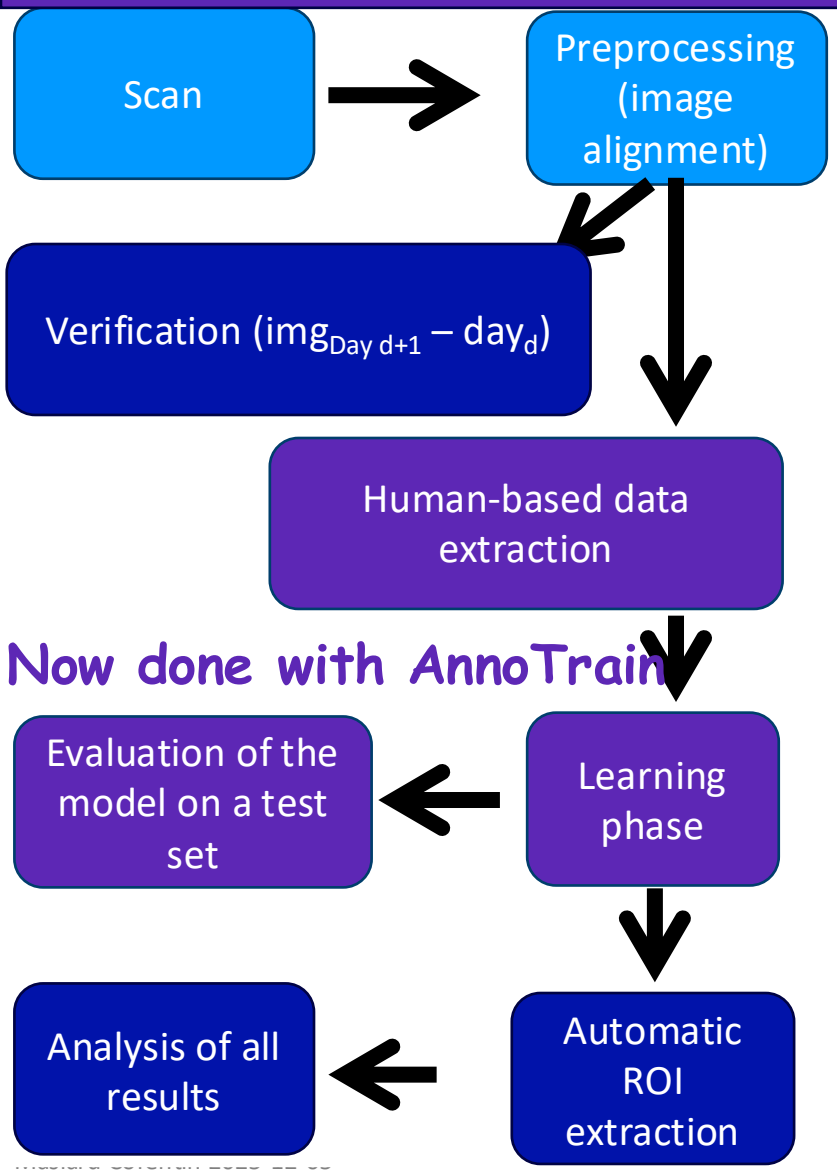




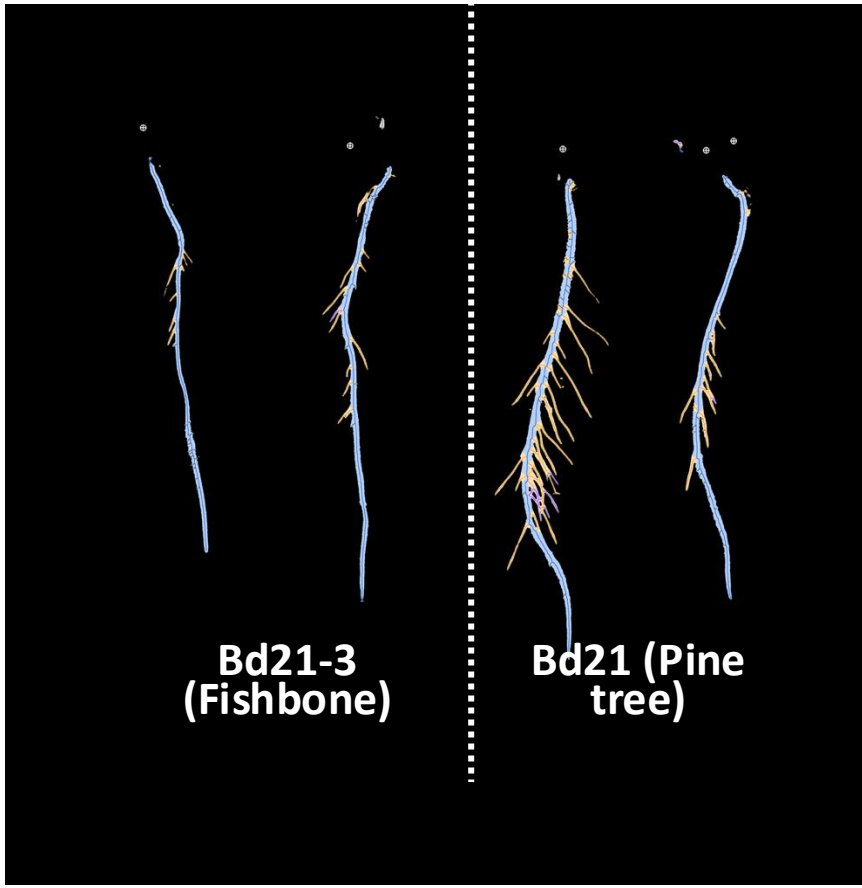


- The two accessions with contrasting lignification under low nitrogen

High-throughput phenotyping could capture accession-specific stress responses



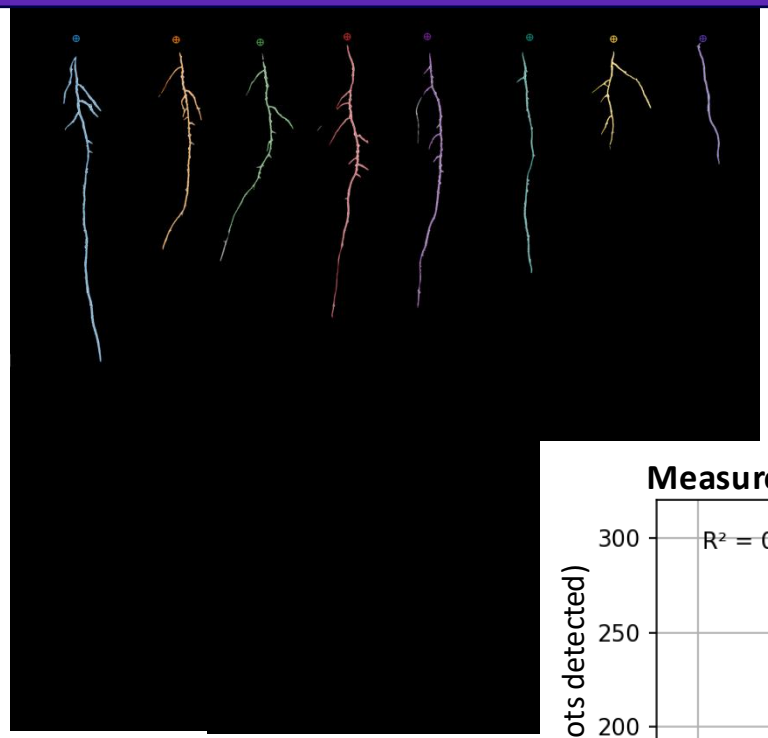


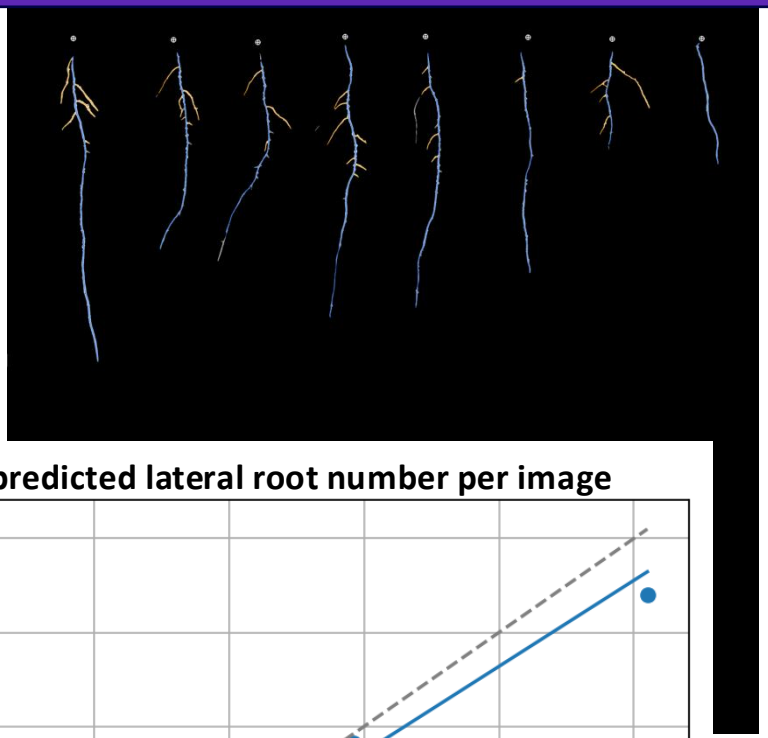


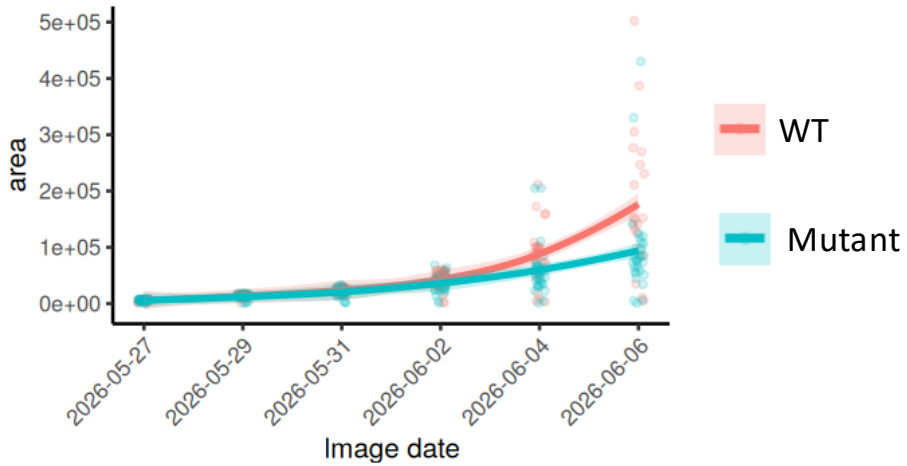
- The two accessions with contrasting lignification under low nitrogen

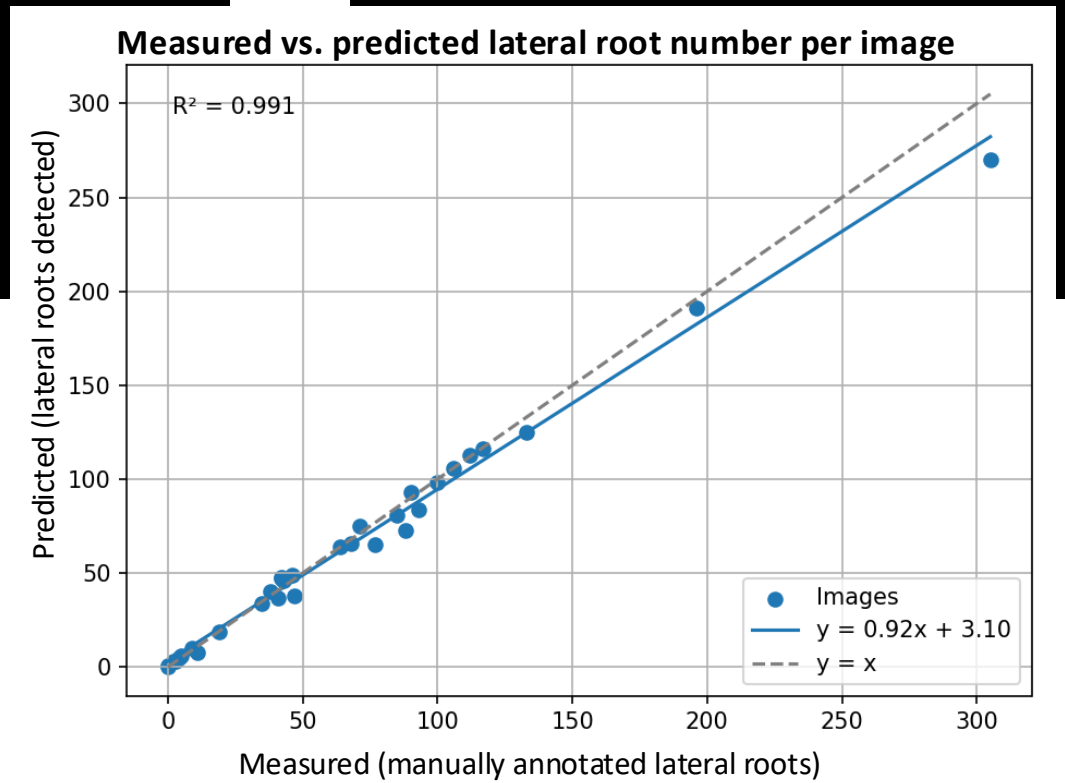
A few annotated images enable adaptation to new species and growth conditions





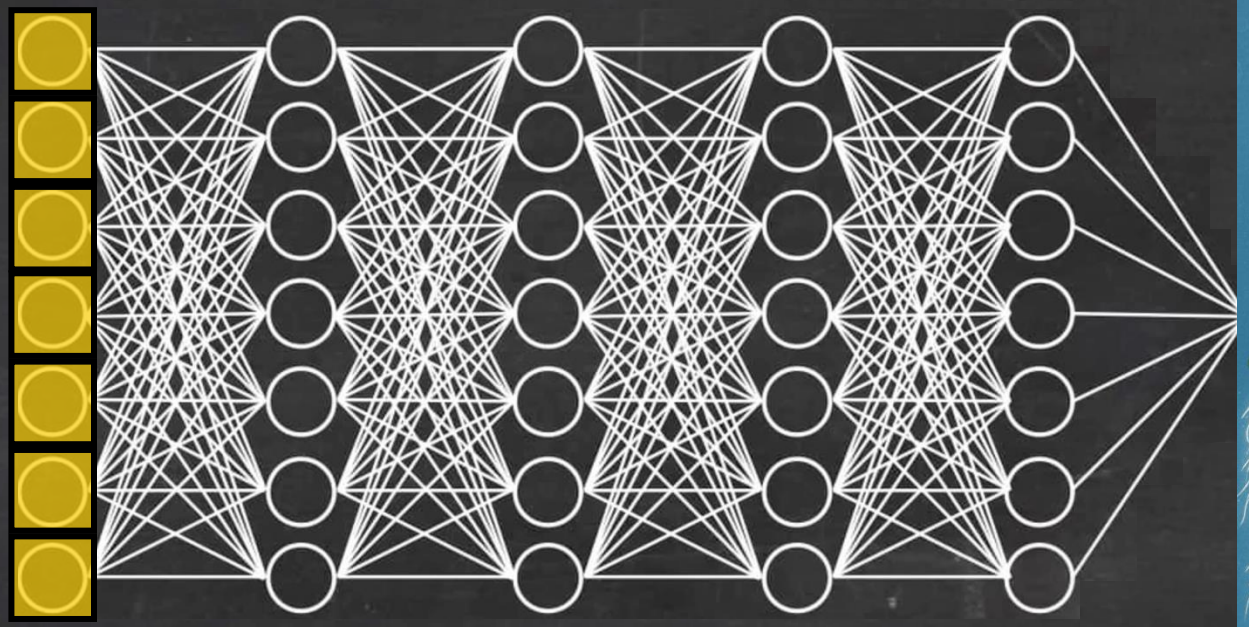






Bonus: How work generative models ? just by returning the network

Values of essential features



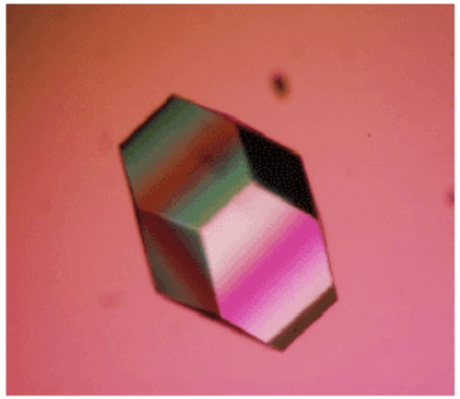
Invented images



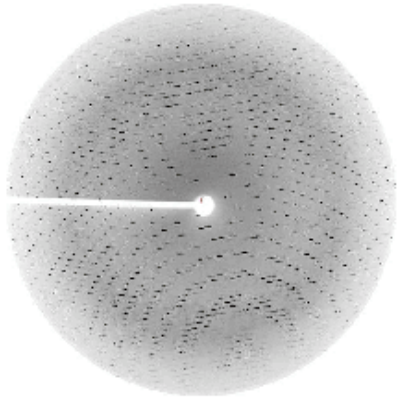
AlphaFold

From a sequence to a structure, and how much to trust it

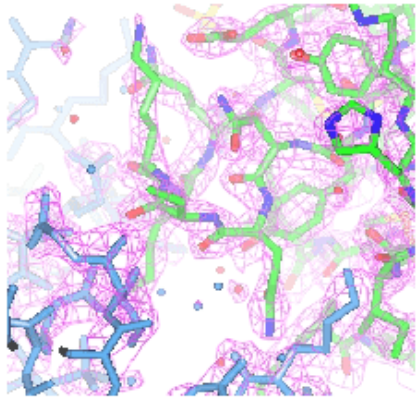
X-ray Crystallography Platform



Crystal



Diffraction pattern



Electron density map

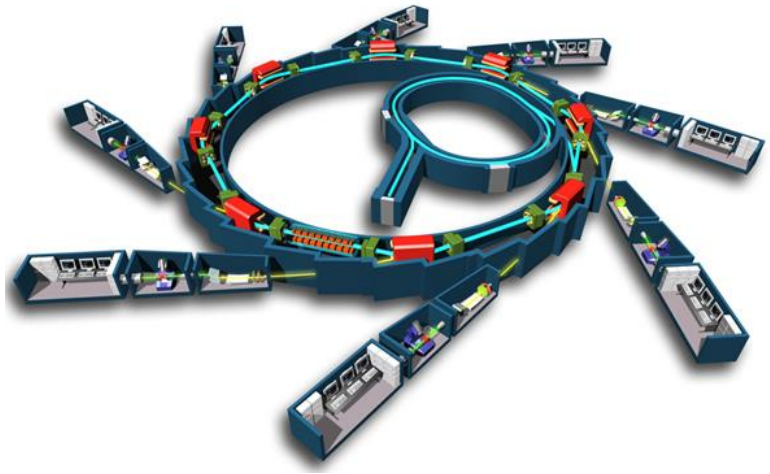


Protein model





Aerial photograph of the synchrotron at the ESRF in Grenoble (France)



Sequence → shape → function: a 50-year-old problem

~200,000

structures solved by experiment (Protein Data Bank), months to years each

100s of millions

of known protein sequences

1972

Anfinsen: the sequence contains the fold

1994

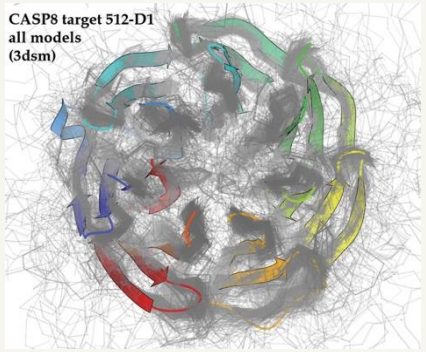
CASP: blind prediction contest

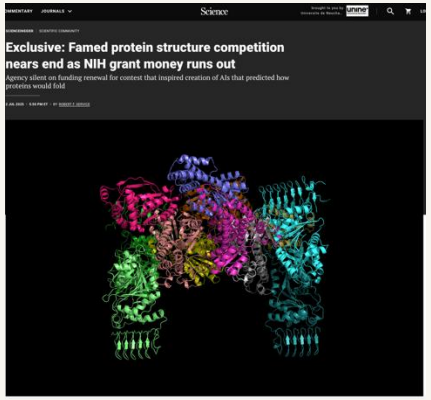
2020

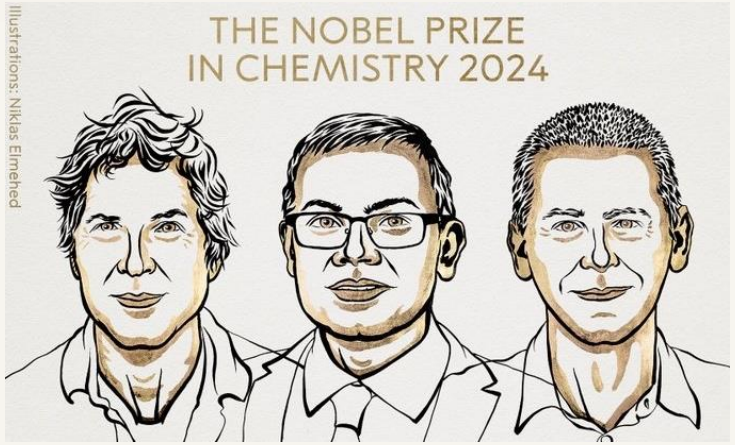
AlphaFold 2: close to experiments

2024

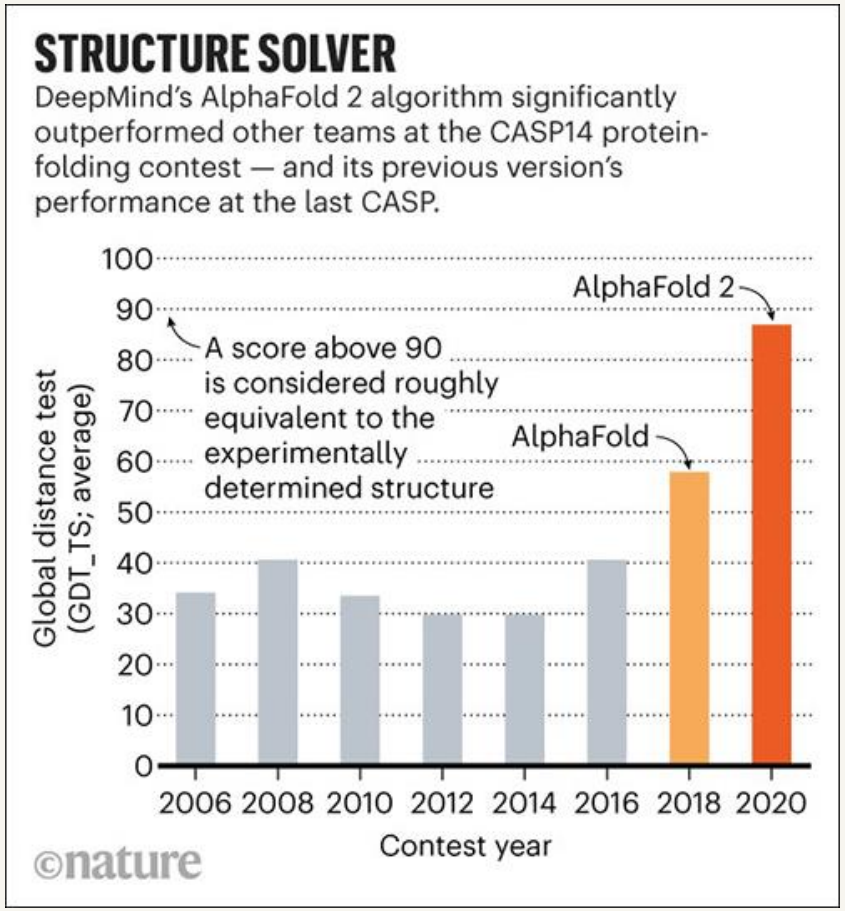
Nobel: Hassabis, Jumper; Baker

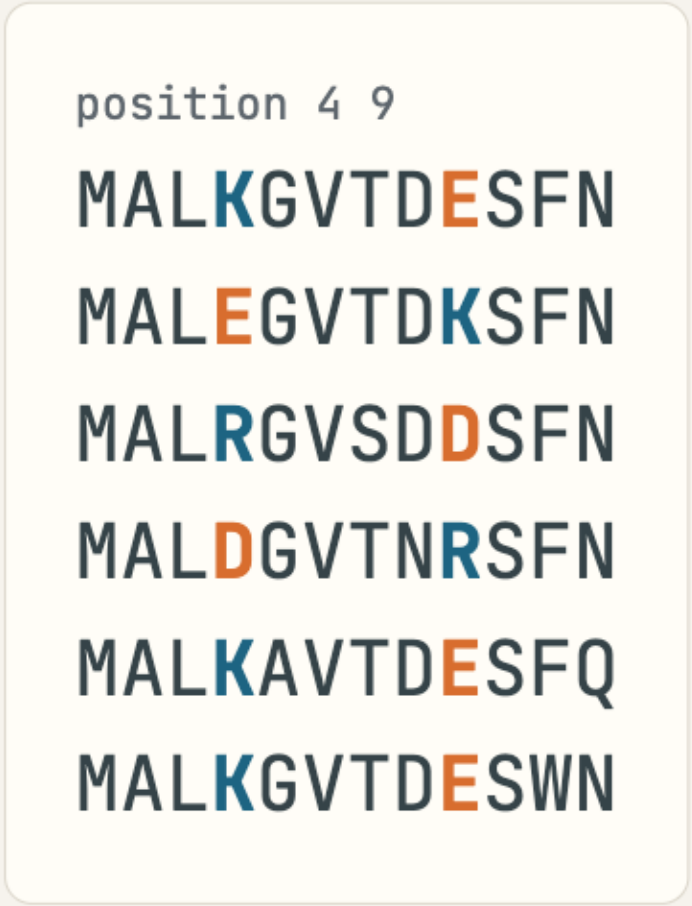






The key idea: amino acids that touch evolve together





- A positive K (blue) faces a negative E (orange): the charges hold the fold.
- If K becomes E, the protein unfolds, **unless its partner also changes**.
- Align thousands of related sequences (as with MAFFT in Session 3): **columns that change together point to amino acids in contact.**

From a sequence to a structure, step by step

1 · Sequence

The letters of a FASTA file: 196 for FLC.

2 · Search

Related sequences, also from soils and oceans, aligned; known structures (templates).

3 · Evoformer × 48

The alignment table and a pair table talk to each other: attention, triangle checks.

4 · Structure module

Moves and turns every amino acid until all agree: x, y, z of every atom.

5 · Recycling × 3

The structure goes through the network again; each pass corrects the last.

6 · Output

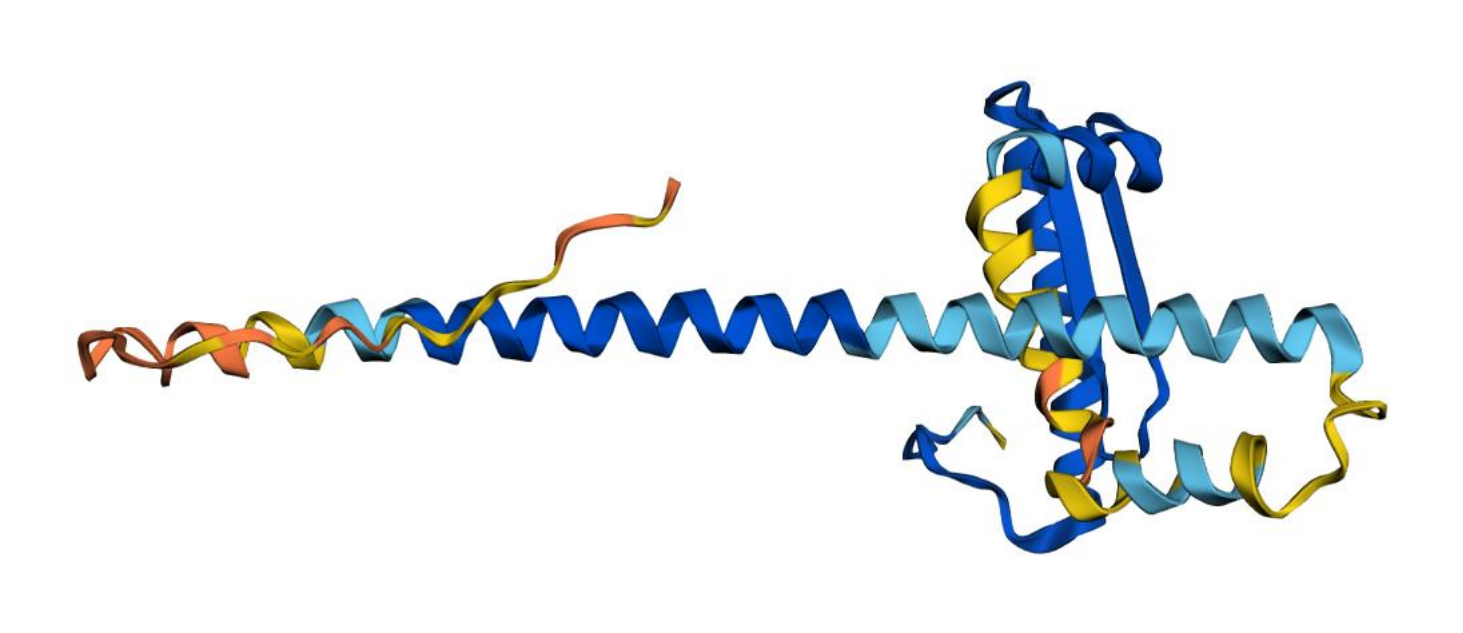
The structure, and its own confidence: pLDDT and PAE.

How it learned: supervised learning, like everything today

Word of today	For AlphaFold
Examples	About 170,000 structures of the Protein Data Bank
Label	The 3D structure solved by experiment
Loss	How far each atom is from where it should be
Weights	About 93 million (ResNet-18: 11 million)
Training	About two weeks on 128 specialised processors

Plus a game of **fill in the blanks**: hidden letters of the alignment to guess, like the ___ of your workbook. To guess them, it learns coevolution by itself.

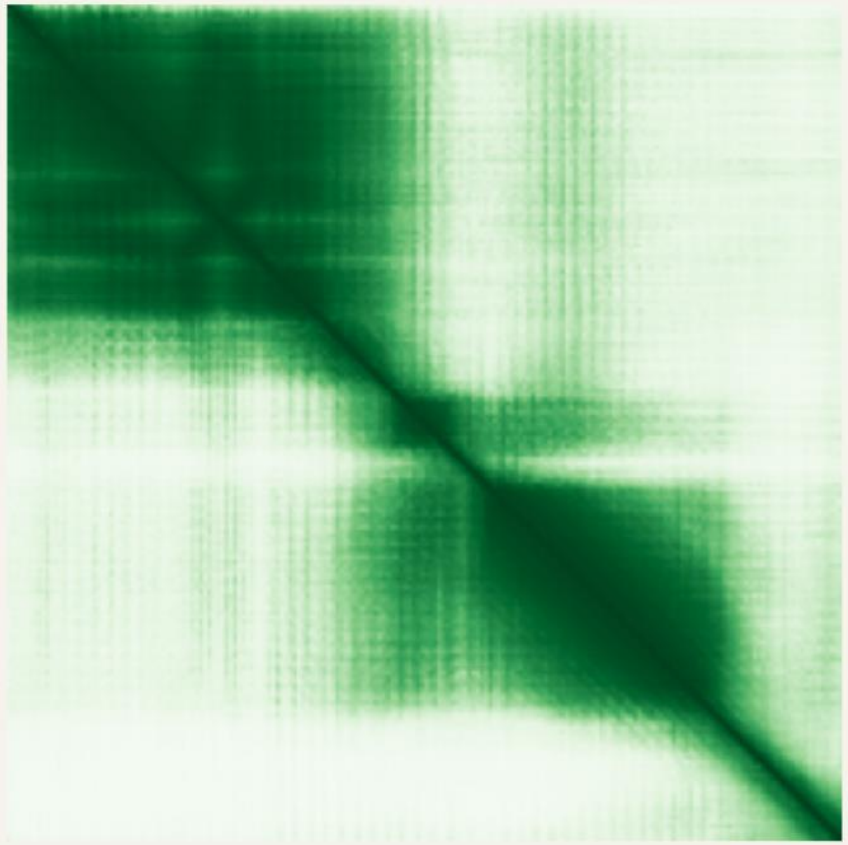
pLDDT: how sure it is, amino acid by amino acid



> 90 very high	70-90 confident	50-70 low	< 50 very low

FLC (Session 6): the MADS-box that grips DNA is dark blue; the tail floats. A very low pLDDT often means **disordered in reality**: information, not a failure.

PAE: are the domains placed right relative to each other?



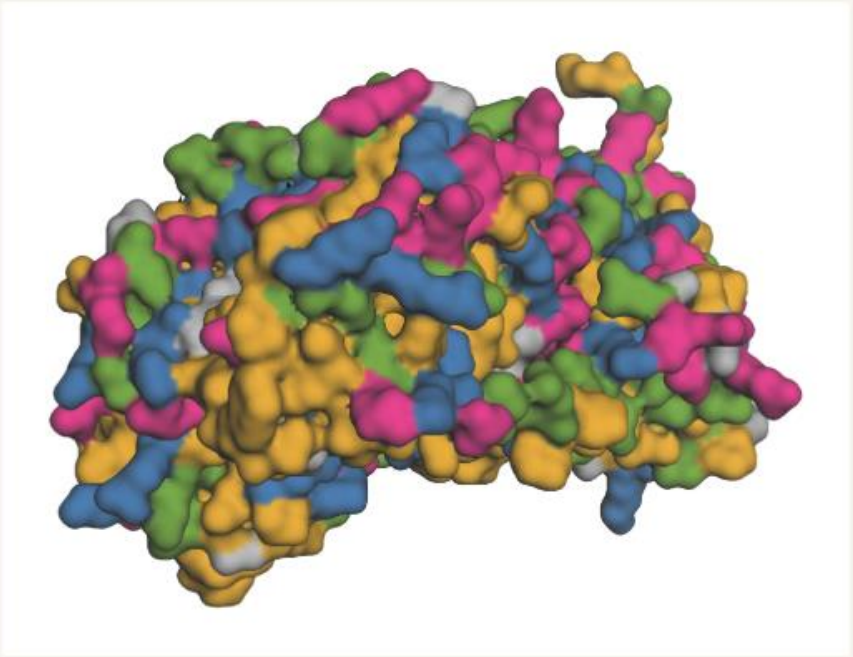
For each pair of amino acids: if I superpose the prediction on the truth using A, how far off is B (in Å)?

Dark squares on the diagonal: domains whose shape AlphaFold is sure of.

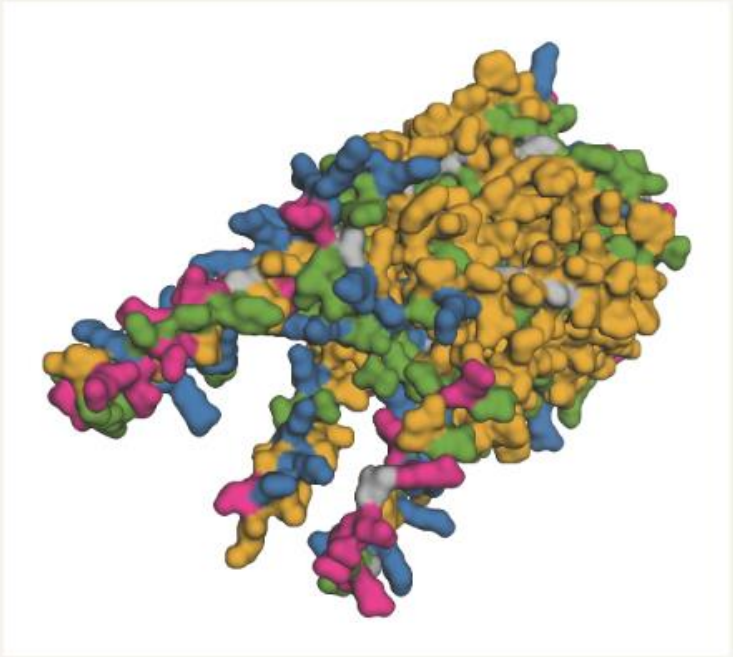
Light between them: where one domain sits relative to the other is unknown.

Image: AlphaFold Protein Structure Database, entry Q9S7Q7 (FLC), CC BY 4.0.

The structure tells where a protein lives



RuBisCO floats in water: most hydrophobic amino acids hidden inside



HKT1 sits in the membrane: a hydrophobic surface facing lipids

hydrophobic **positive** **negative** **polar**

https://odh-like-search.private.cmaslard.xyz/

What AlphaFold 2 does not do

One chain, one state

No DNA, no ligand, no partner, no movement. AlphaFold 3 and the AlphaFold Server add complexes.

It needs relatives

Few related sequences: a poor alignment, a poor prediction.

Not for one mutation

The mutant usually comes out looking like the normal protein.

A hypothesis

To be checked by experiment, like a GWAS peak.

Using AI well as a biologist

1. **Use an existing tool first**: Pl@ntNet, BirdNET, MegaDetector, Cellpose, AlphaFold. Train your own only when none fits

2. **Check it on your own data**: annotate a sample yourself, compute the recall of each class

3. **Look beyond the accuracy**: the confusion matrix, the rare classes, the uncertain cases

4. **Chatbots: excellent assistants, unreliable sources.** Never trust a number or a reference without checking it

5. **Never paste** unpublished, personal or confidential data into an online AI

Now your turn: WORKBOOK.qmd

```
mkdir -p ~/bioinfo_course/session_07/data ~/bioinfo_course/session_07/results
cp /home/cmaslard/bioinfo_tools_course/session_07/data/* ~/bioinfo_course/session_07/data/
cp /home/cmaslard/bioinfo_tools_course/session_07/WORKBOOK.qmd ~/bioinfo_course/session_07/
```

A-B Rules by hand, then a decision tree

C-D Overfitting, a forest, a neural network

E-F A photo is numbers; a real AnnoTrain model

G-H AlphaFold; a tour of AI tools

Each block of the course page links back to its theory part. Nothing to submit: **understand, ask, enjoy**.