

Real-world Multi-agent Systems

Machine learning and AI

Akin Delibasi
Lecturer in Distributed Systems
Computer Science
University College London, UK



Introduction

- Definition and concepts
- History and current state of research
 - Turing Test
- Machine learning paradigms/techniques
- Fundamentals of neural networks:
 - Activation functions
 - Backpropagation
 - Reward design
- Examples & applications

Artificial Intelligence (AI)

The *Dartmouth workshop*

In 1956, a summer workshop for artificial intelligence, named the *Dartmouth Summer Research Project on Artificial Intelligence* open a new field of AI.

In John McCarthy's proposal, he stated that the conference was

"to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it."

The *Dartmouth workshop*

The Proposal states:

“We propose that a 2 month, 10 man study of artificial intelligence be carried out during the summer of 1956 at Dartmouth College in Hanover, New Hampshire. The study is to proceed We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer.”

1956, birth of the artificial intelligence (AI) research.

Artificial Intelligence (AI)

- **Definition:** AI is a broad field within computer science that aims to build machines or software capable of performing tasks that typically require human intelligence; tasks include reasoning, learning, problem-solving, perception, and language understanding.
- **A Broader Scope:** Encompasses a wide range of intelligent behaviors.
- **Goal:** To develop systems capable of functioning intelligently and independently by itself.
- **Applications:** autonomous vehicles, chess-playing (eg, Alpha Go), prediction of protein folding (Alpha-fold).

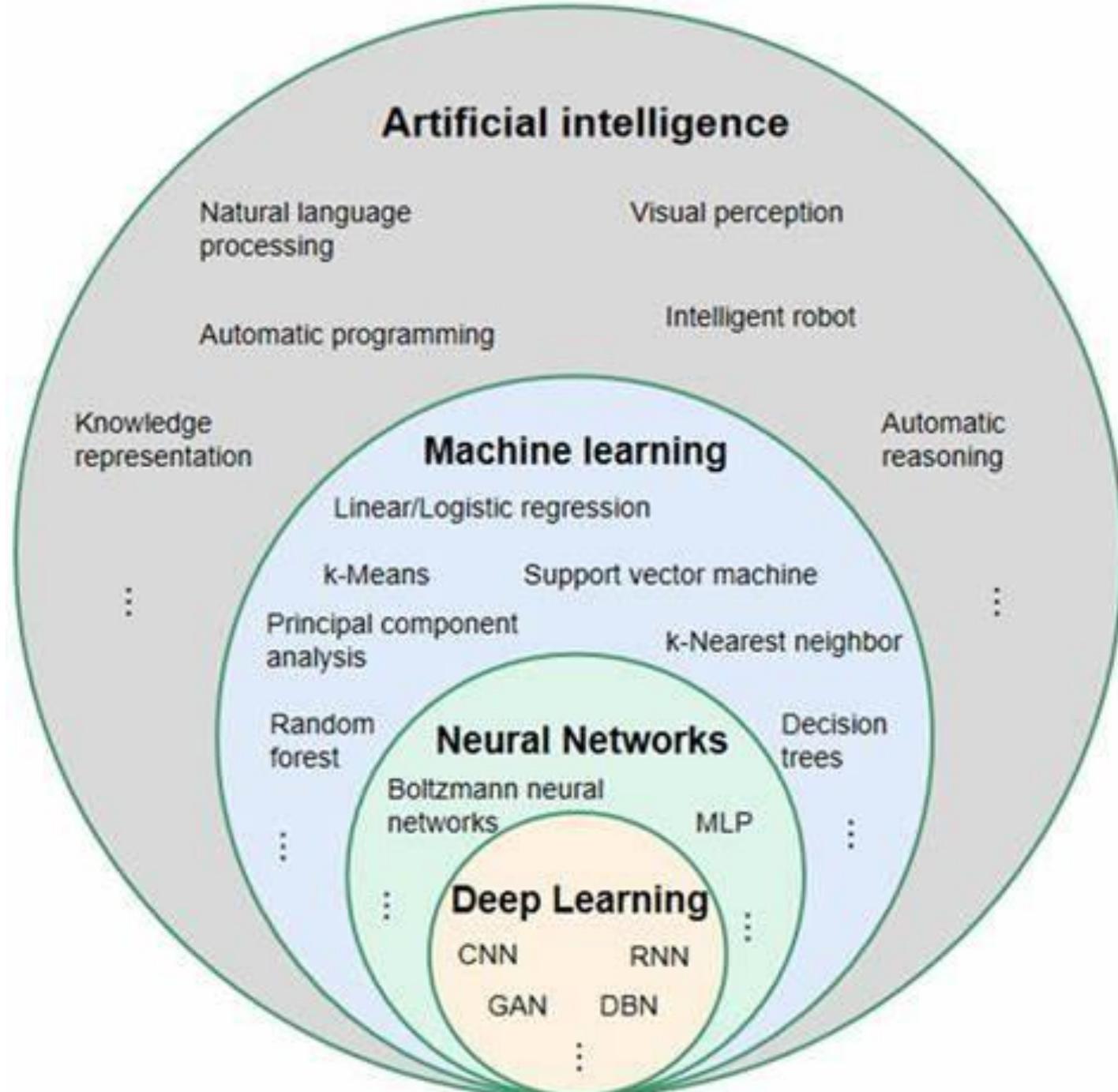
Machine Learning (ML)

- **Definition:** ML is a subset of AI that focuses on enabling machines to learn from data and make predictions or decisions without being explicitly programmed to perform the task (generalization).
- Narrower focus: specifically on creating systems that learn from data.
- Goal: To allow machines to learn and make predictions or decisions based on the data they process.
- Applications: handwriting recognition, speech recognition, computer vision.

Similarities and differences

- ML is a part of AI: ML is a specific approach within AI that emphasizes learning from data.
- AI is the broader goal: AI includes a wider range of technologies that simulate human intelligence, with ML being one of them.
- ML is about learning and adapting: ie, algorithms that adjust and improve based on the data.
- AI includes more than learning: AI encompasses reasoning, planning, and problem-solving, not just learning from data (as ML does).

Similarities and differences

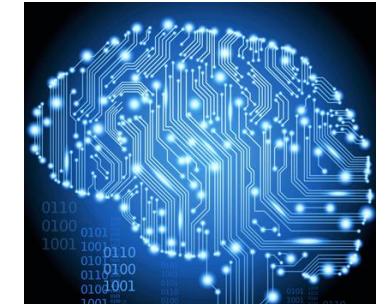


Similarities and differences

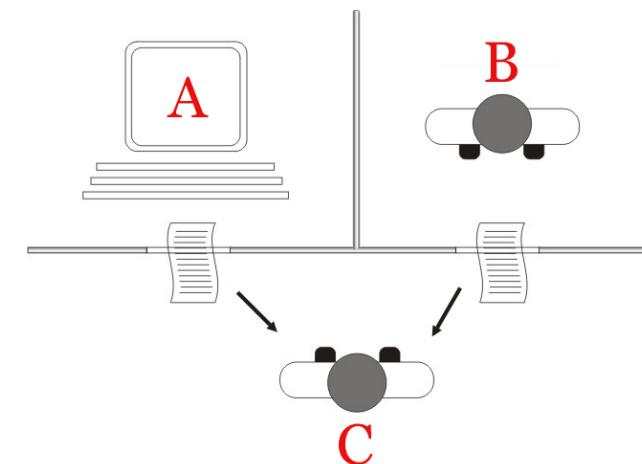
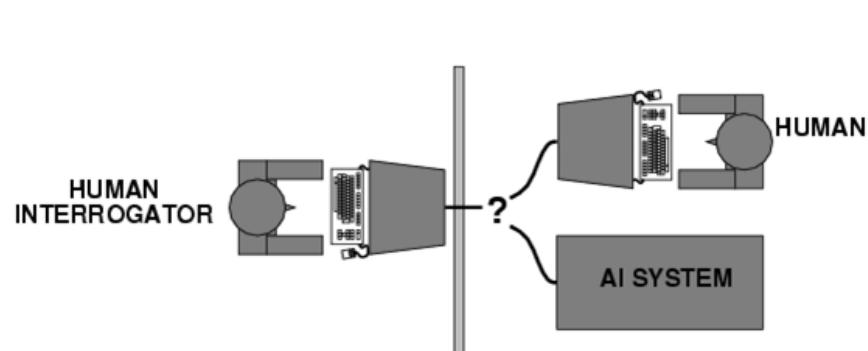
- Other AI algorithms that are not ML-based:
 - **Expert Systems:** AI programs that simulate the decision-making of a human expert; reasoning through preprogrammed knowledge database, mainly if–then rules.
 - **Genetic Algorithms:** (global) search and optimization, based on the principles of genetics and natural selection, i.e., survival of the fittest.
 - **Swarm Intelligence Algorithms:** emergent intelligence “naturally” generated by the interplay of many simple agents (inspired by the behavior of social insects like ants and bees).
 - **Fuzzy Logic Systems:** mathematical systems for dealing with uncertain or approximate information; particularly useful when there is a lack of precise models.
 - **Symbolic AI:** This involves the explicit embedding of human knowledge and reasoning capabilities into computer programs.

Turing Test

In the 1940s and 50s, a lot of discussion was held by scientists from different fields on the possibility of creating an artificial brain.

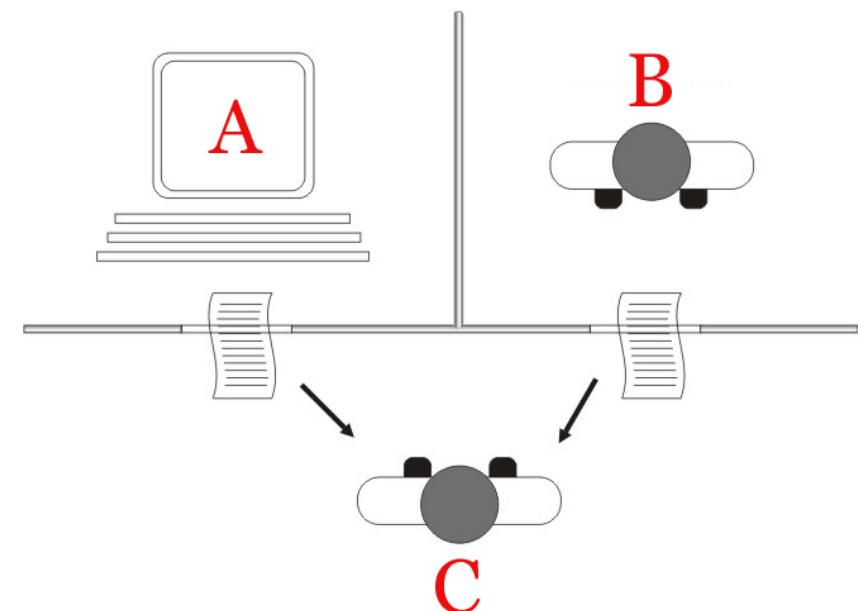
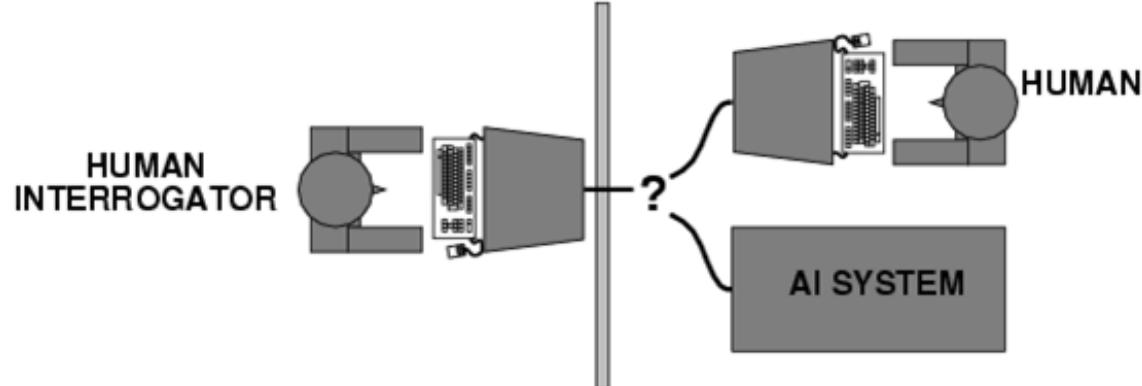


Can machines think? In 1950, Alan Turing proposed ***Turing Test***



Turing Test

Can machines think? Turing Test (1950): a human judge engages in a natural language conversation with a human and a machine designed to generate performance indistinguishable from that of a human being.



Turing Test (2016)

A natural language conversation, is that a human or a machine?

[red circle] Hi, what is the news about Harvey?

[blue circle] **Alana_live**

[eliza-e557b16-2017-08-12-1858] Sorry I don't have access to that document.

[blue circle] I mean the hurricane.

[red circle] **Alana_live**

[eliza-e557b16-2017-08-12-1858] So, what is your guilty pleasure?

[blue circle] Ok. What is the news about Harvey hurricane?

[red circle] **Alana_live**

[persona-e557b16-2017-08-12-1858] Thank you! I am still learning. What do you want to talk about next? I can tell you an interesting fact or chat about politics, news, or movies..



[red circle] news on Harvey hurricane



[blue circle] **Alana_live**

[eliza-e557b16-2017-08-12-1858] So what is the last concert you went to?



[red circle] news on Harvey



[blue circle] **Alana_live**

[eliza-e557b16-2017-08-12-1858] Anyway, what are your plans for the rest of the day?



[red circle] news on Harvey



[blue circle] **Alana_live**

[news_api-f5f1472-2017-08-11-1640] I heard on the radio that The Specter of Climate Change Hangs Over Hurricane Harvey. Was Hurricane Harvey the result of climate change? Do you want to know more?

Turing Test (2023)

A natural language conversation, is that a human or a machine?

Hi, what is the news about Harvey?

✓ Searching for: latest news about Harvey

✓ Generating answers for you...

I mean the hurricane.



✓ Searching for: latest news about Hurricane Harvey

The

✓ Generating answers for you...

1.

🛡 Your personal and company data are protected in this chat

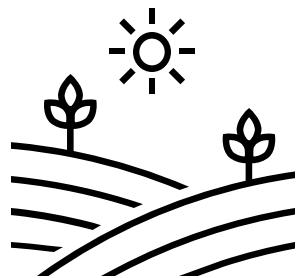
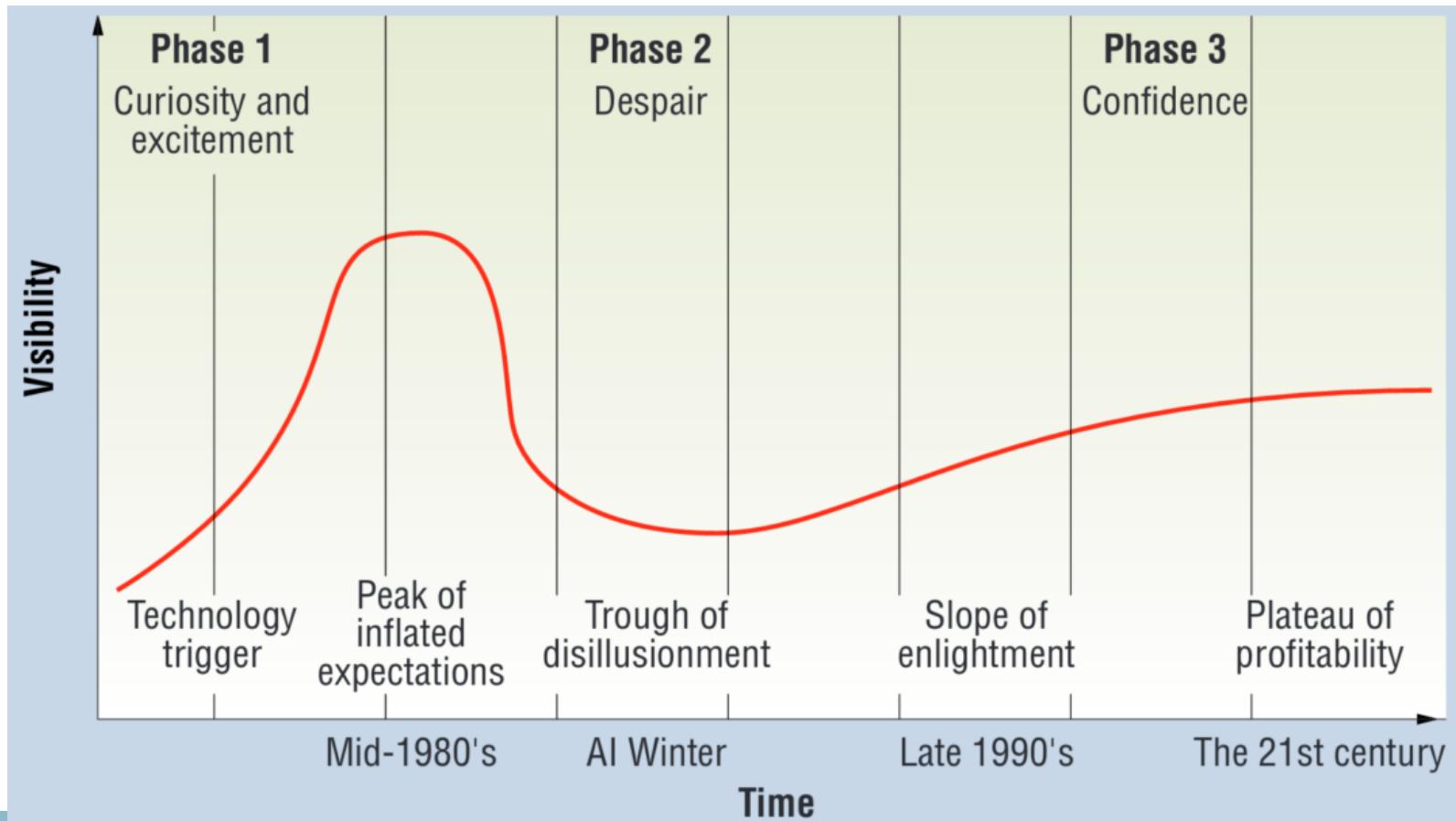
Here are some recent updates related to Hurricane Harvey:

1. A Houston pizzeria, Jackpot Pizza, which fed victims of Hurricane Harvey, is now on the brink of collapse ¹.



AI winter → AI Spring

AI, cannot be exempted from the hype cycle for new technology.

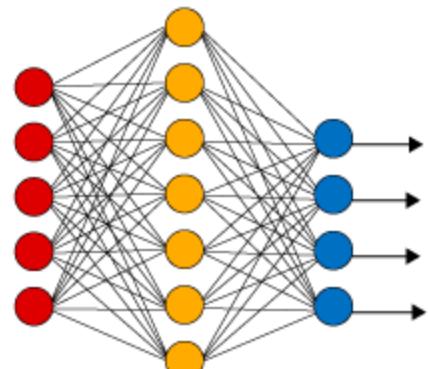


New era of deep learning, 2012

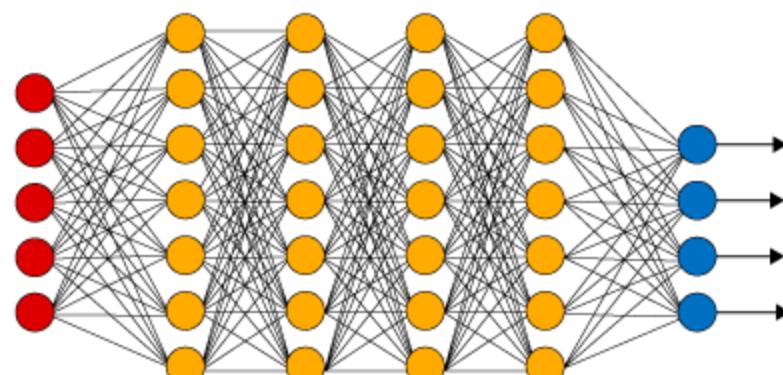
In 2012, Deep Convolutional Neural Networks won the large-scale ImageNet competition by a significant margin over shallow machine learning methods.

Deep learning: more hidden layers, which enable composition of features from lower layers, potentially modeling complex data with fewer units than a similarly performing shallow network.

Simple Neural Network



Deep Learning Neural Network



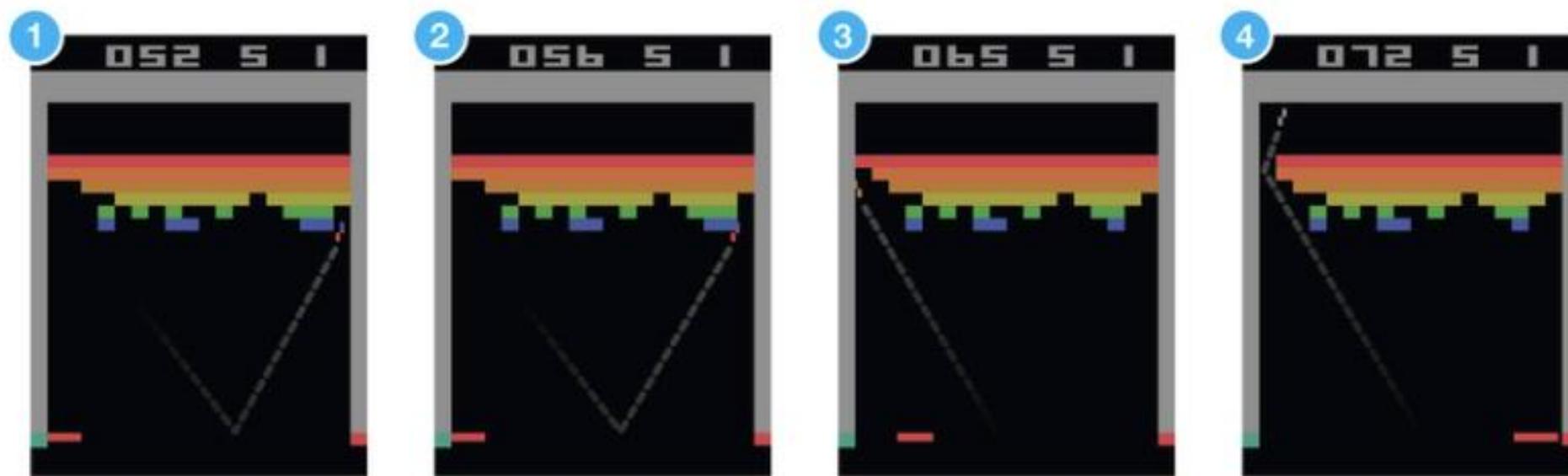
● Input Layer

● Hidden Layer

● Output Layer

New era of deep learning, 2014

2014, DeepMind developed Deep Q-learning capable of learning how to play Atari video games using only pixels as data input. [[video](#)]



Alpha-Go vs Lee Sedol, 2016

AlphaGo is a narrow AI specialized in playing the board game Go.

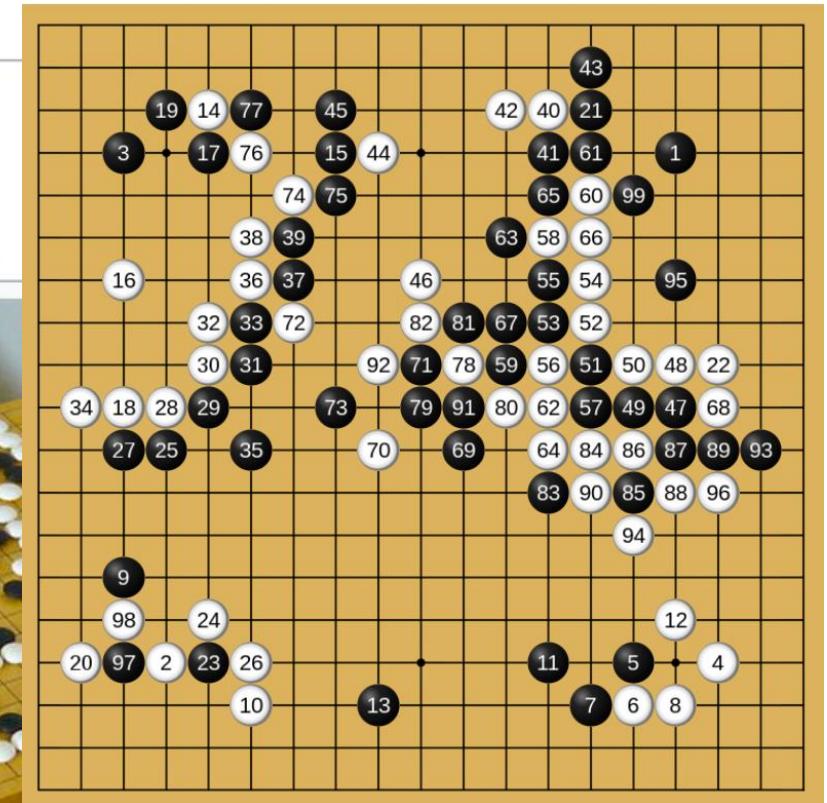
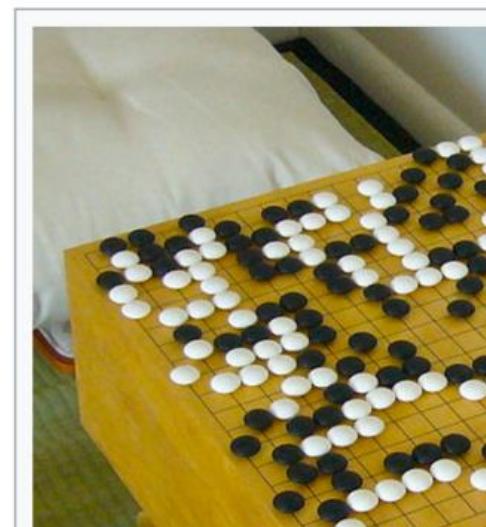
Go (game)

From Wikipedia, the free encyclopedia

This article is about the board game. For other things named "Go", see [Go](#).

Go (traditional Chinese: 圍棋; simplified Chinese: 围棋; pinyin: [wéiqí](#); Japanese: 囲碁; [rōmaji](#): [igo](#)^[nb 2]; Korean: 바둑; romaja: [baduk](#)^[nb 3]; literally: "encircling game") is an abstract strategy board game for two players, in which the aim is to surround more territory than the opponent.

The game was invented in [ancient China](#) more than 2,500 years ago, and is therefore believed to be the oldest board game continuously played today.^{[2][3]} It was considered one of the four essential arts of the cultured [aristocratic](#) Chinese scholars in antiquity. The earliest written reference to the game is generally recognized as the historical annal [Zuo Zhuan](#)^{[4][5]} (c. 4th century BCE).^[6]



Alpha-Go vs Lee Sedol, 2016

What AI and robotics still cannot do? Can you see it?

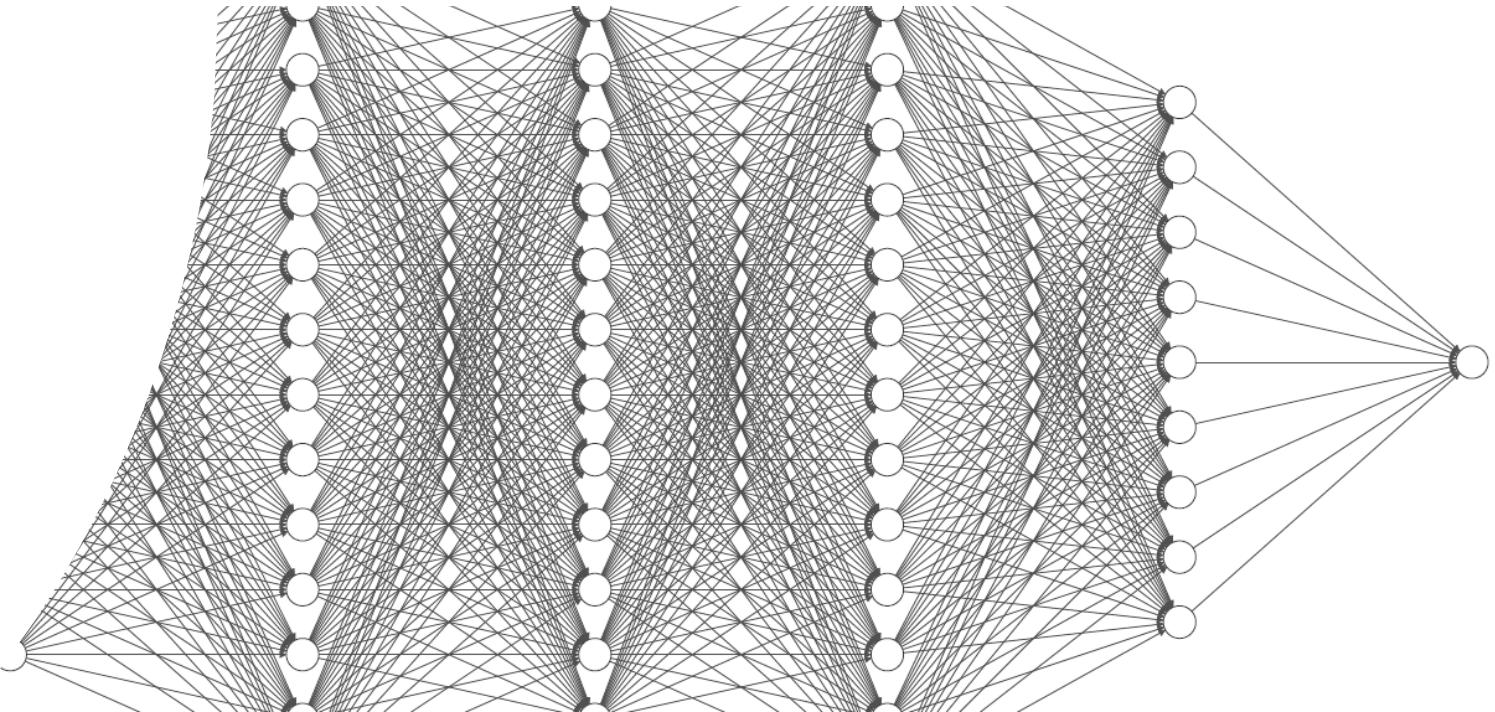
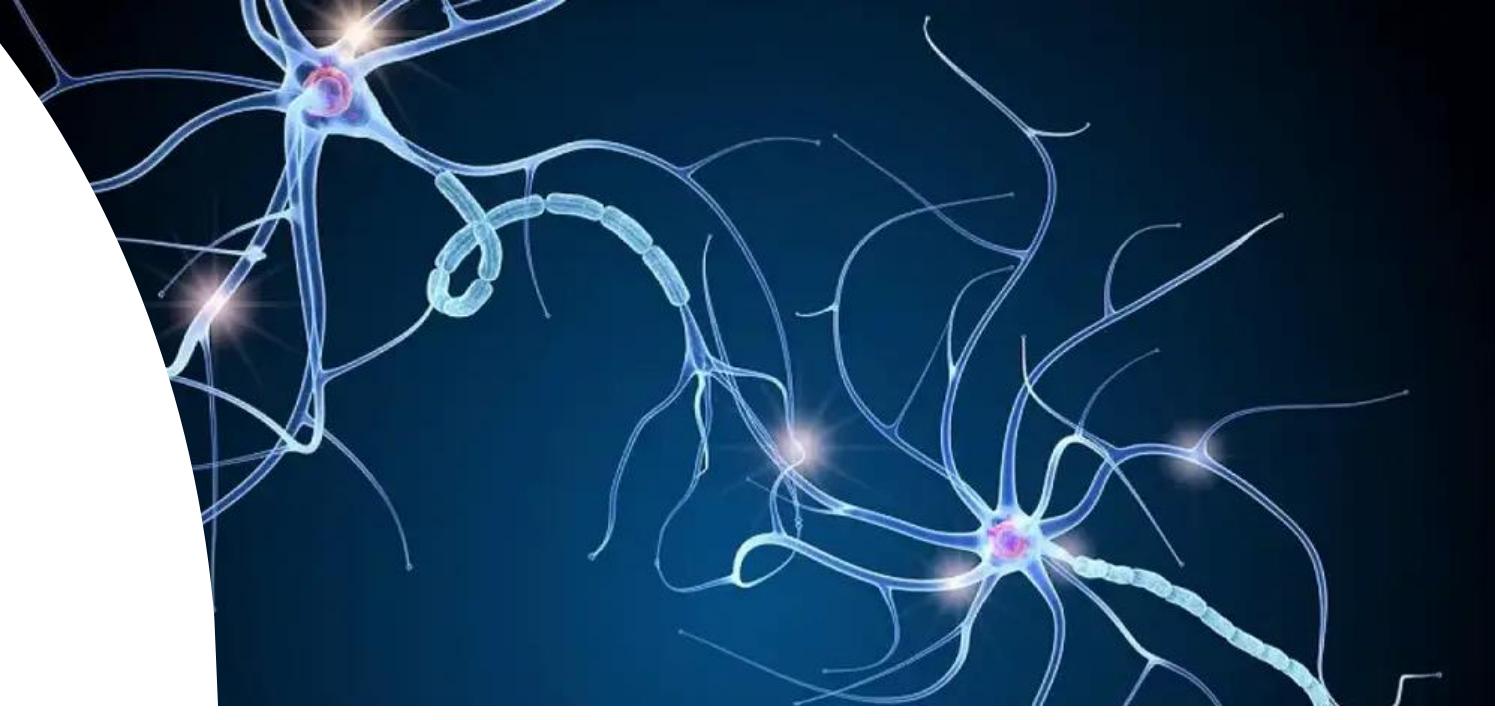


Reliable control of physical interaction is hard in real world.



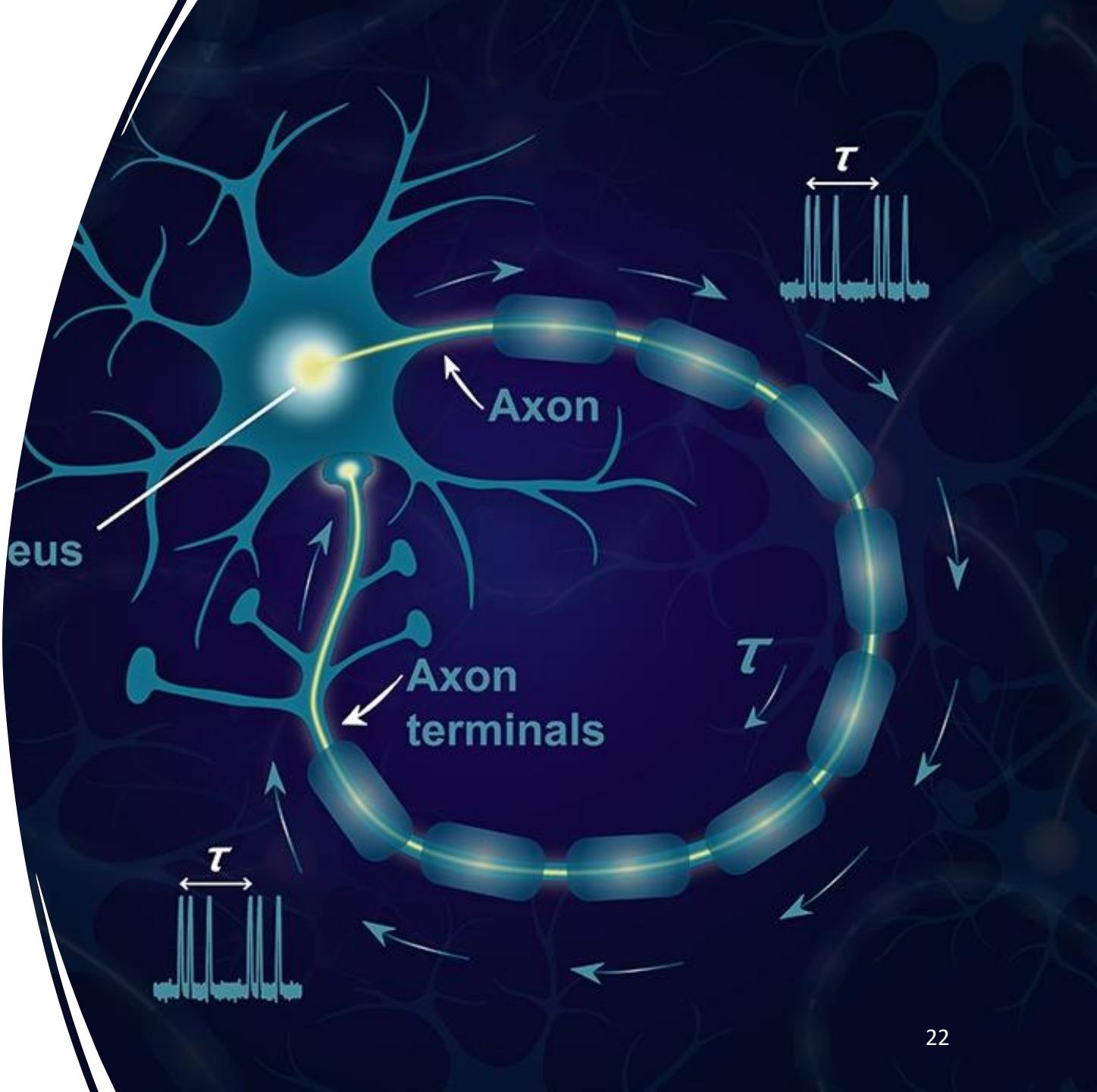
Machine learning

Neural Networks



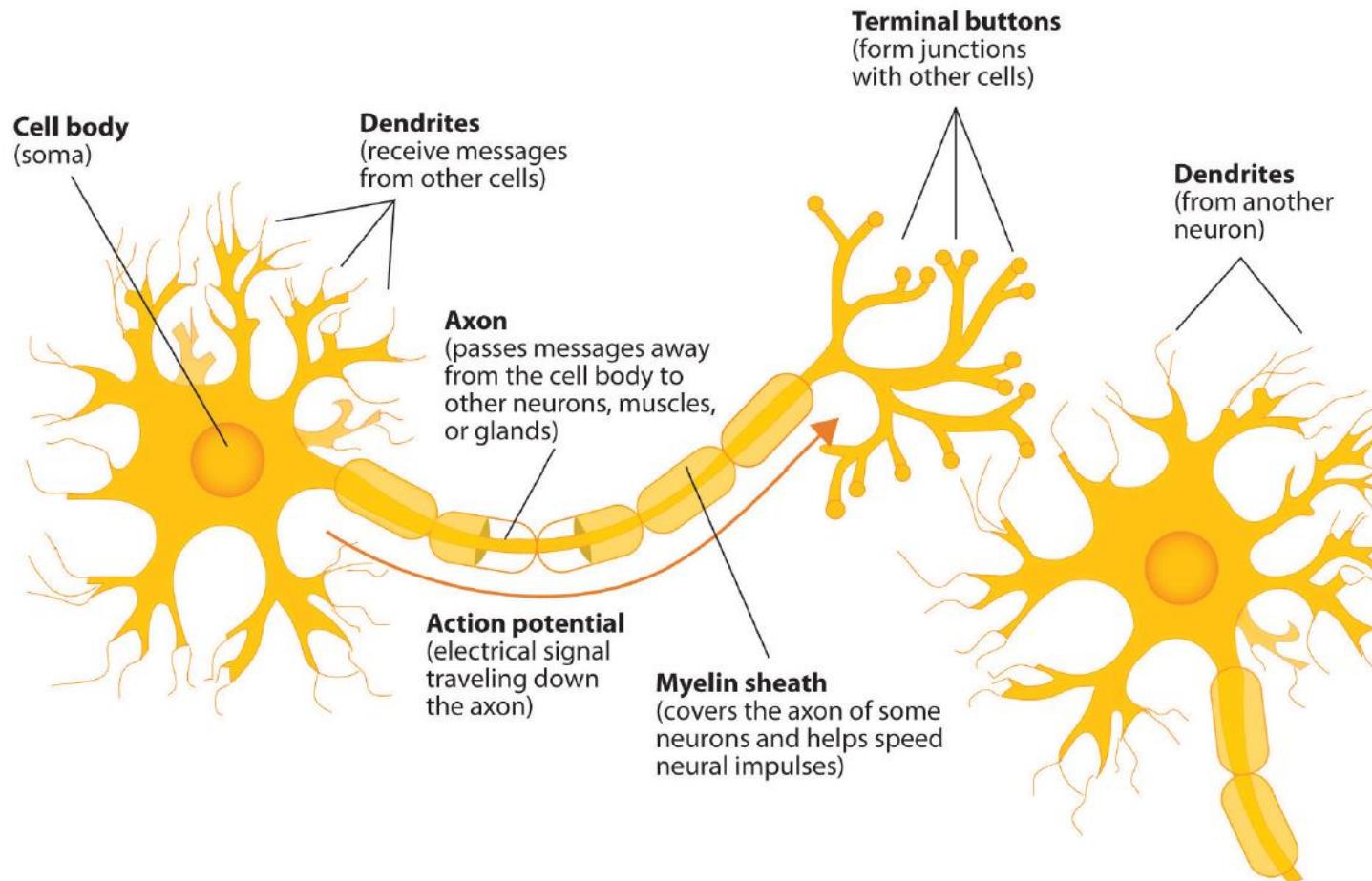
Artificial neural network

Neural Networks (NN), or more precisely, Artificial Neural Networks (ANN) are computing systems that are **inspired** by, but **not** identical to, **biological neural networks**.



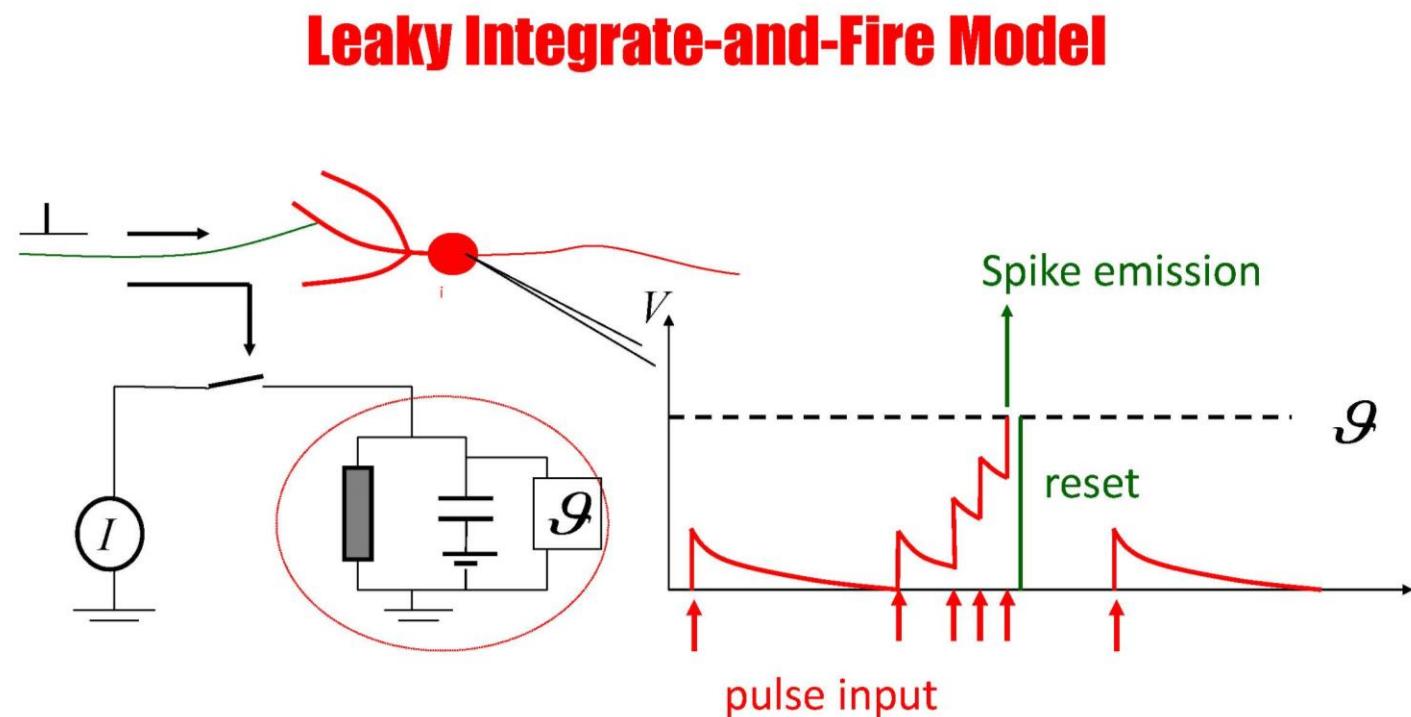
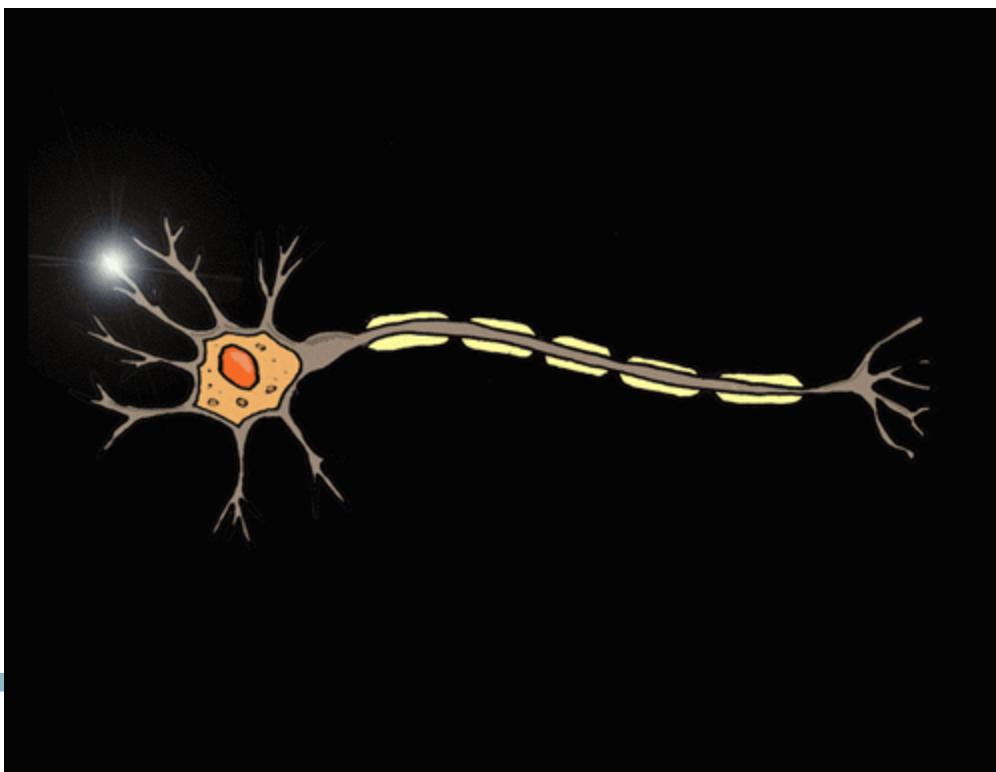
Biological neurons

- Neurons are a type of cell as the primary components of the nervous system that receives and sends signals.



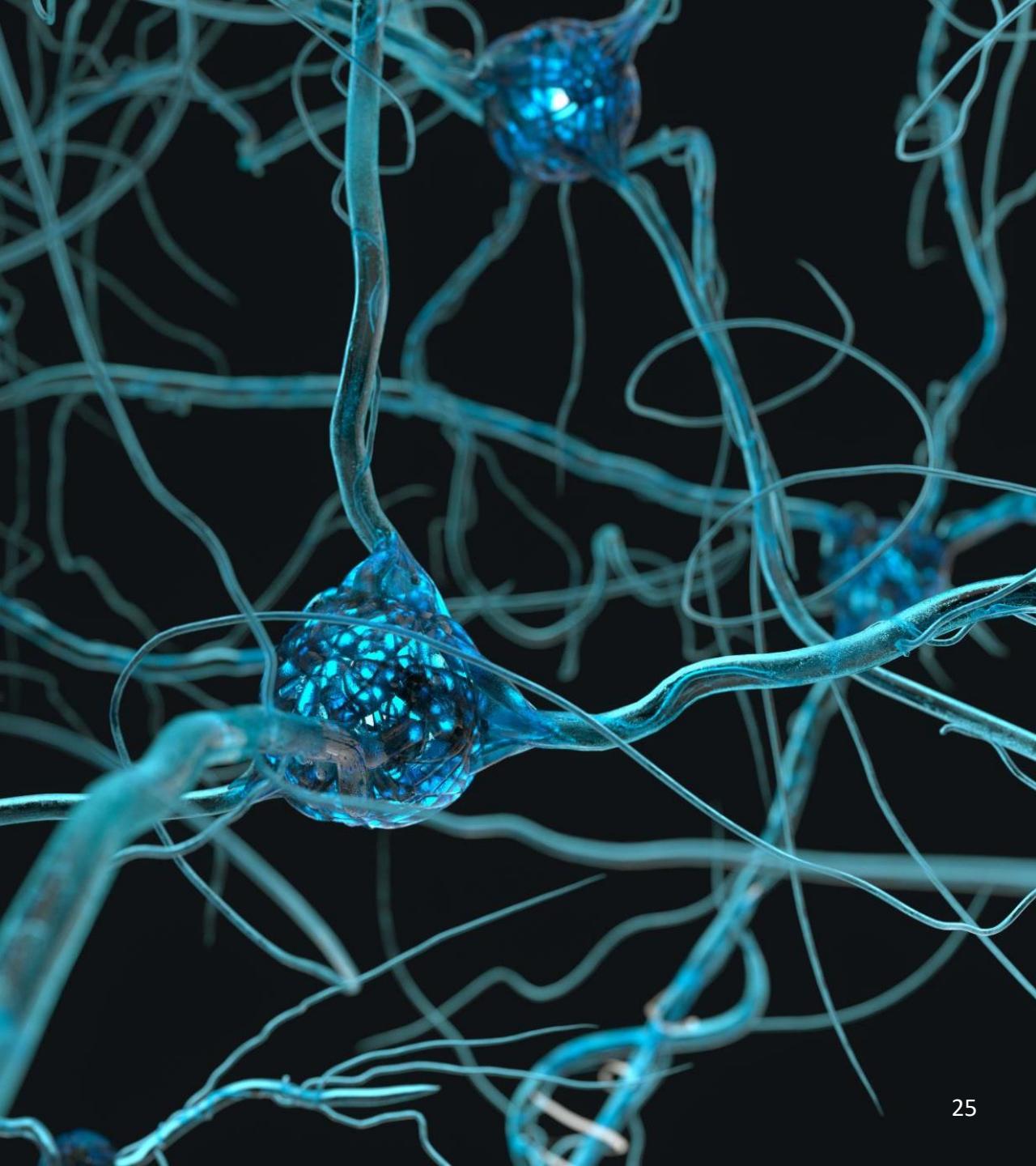
Neural model

- It is represented by an RC circuit with a threshold. Each input pulse (e.g. caused by a spike from a different neuron) causes a short current pulse. Voltage decays exponentially. If the threshold is reached an output spike is generated and the voltage is reset.



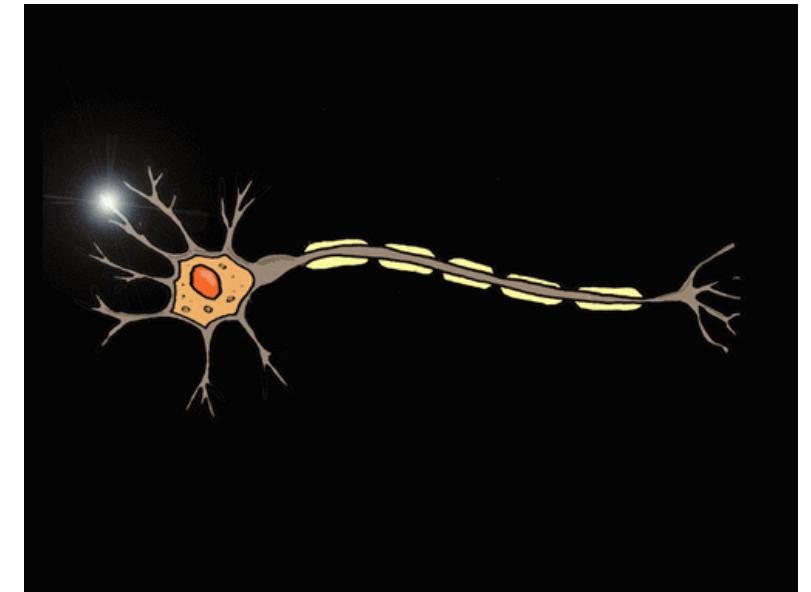
Neural model

- **Artificial Neural Network (ANN):** a computational system that consists of many interconnected units called artificial neurons.
- **Neurons:** These units, also referred to as nodes or neurons, are simple processors which operate in parallel.
- **Connection** between artificial neurons can transmit a signal from one neuron to another. Every neuron is connected with other neurons through a connection link. These connections, like the synapses in a biological brain, can transmit a signal (carry/pass information) to other neurons.

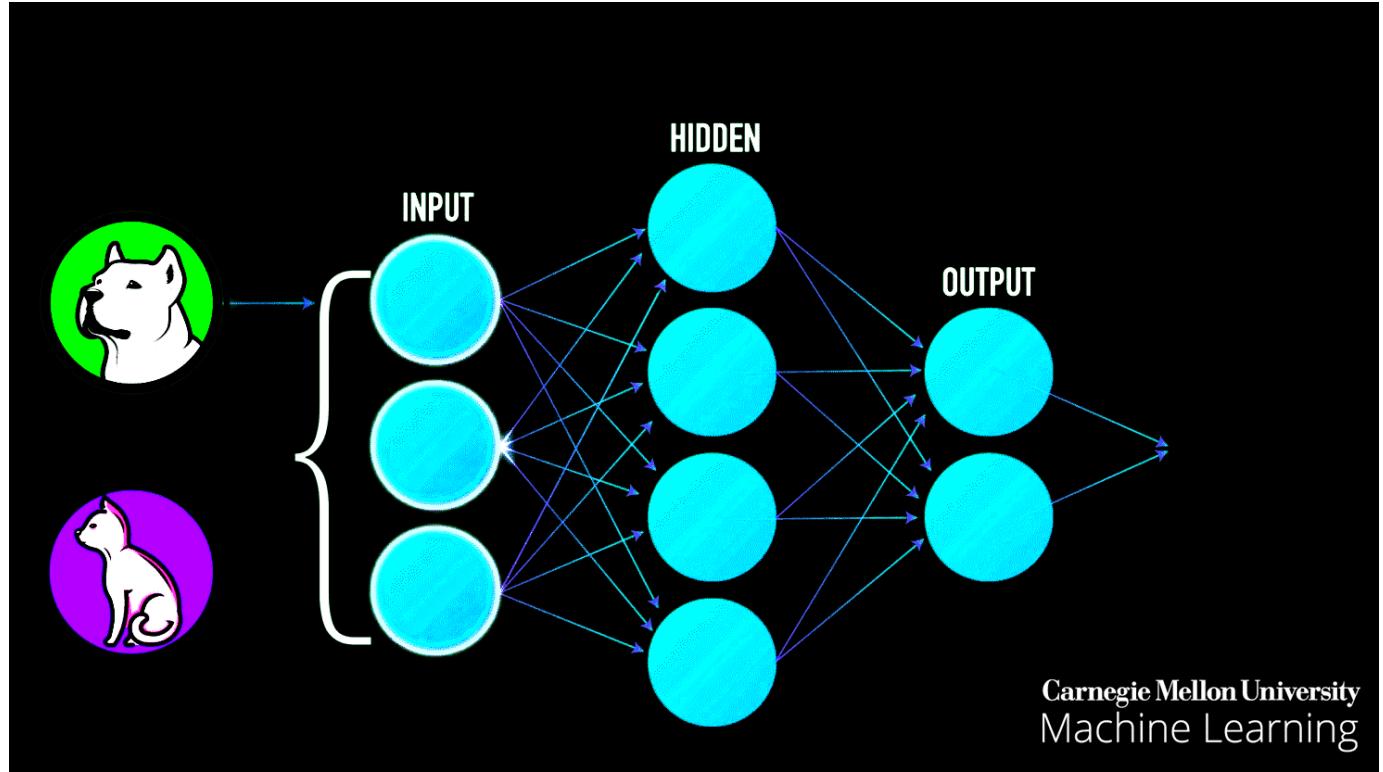
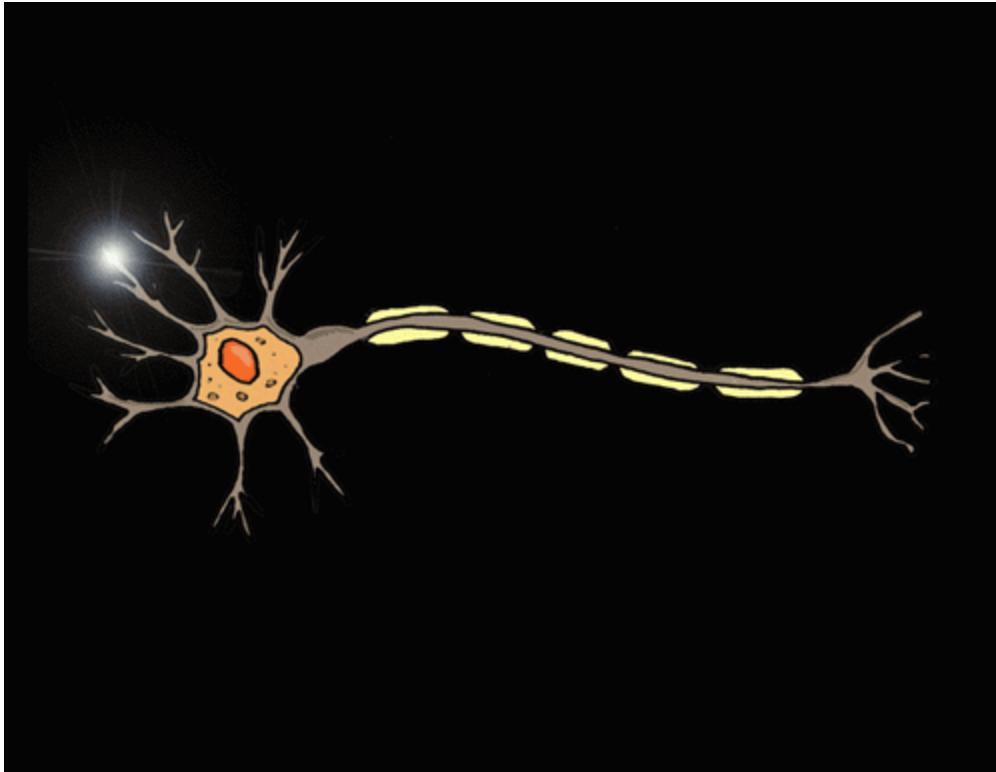


Neural model

- The “signal” at a connection is a real number, and the **output** of each neuron is computed by the non-linear function of the sum of **its inputs**. Neurons connections typically have **weights** that adjusts as learning proceeds. The weight increases or decreases the strength of the signal at a connection.
- Neurons have a threshold such that a signal is sent only if the aggregate signal goes beyond that **threshold**. Typically, neurons are aggregated into layers. Different layers perform different transformations on their inputs. Signals travel from the first layer (the **input layer**), to the last layer (the **output layer**).



Neural model

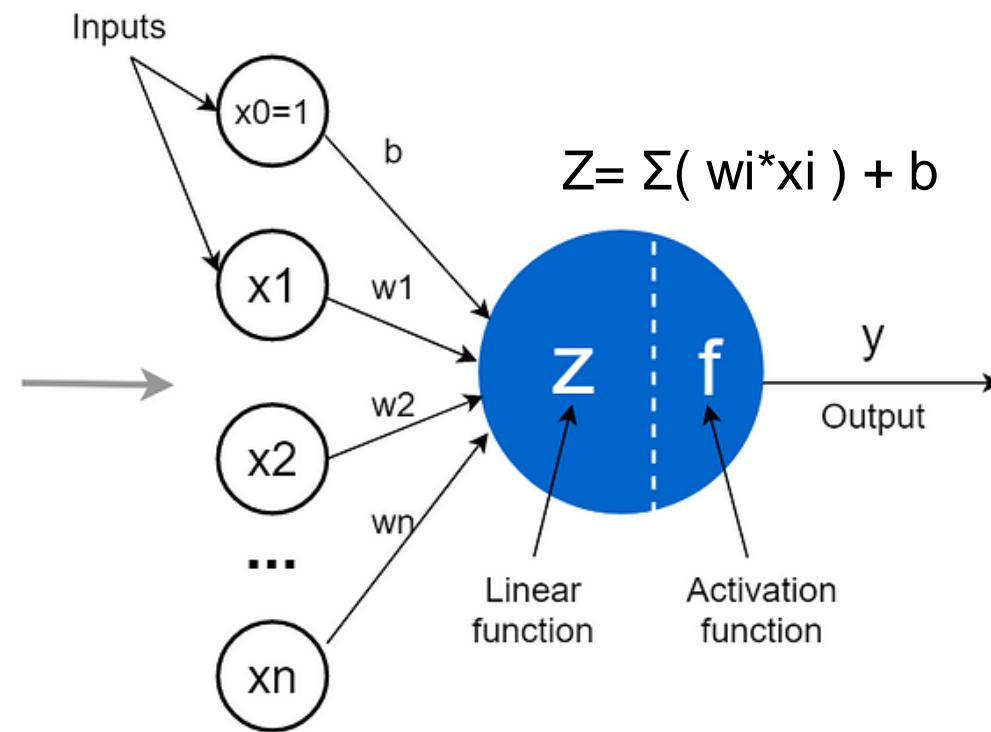
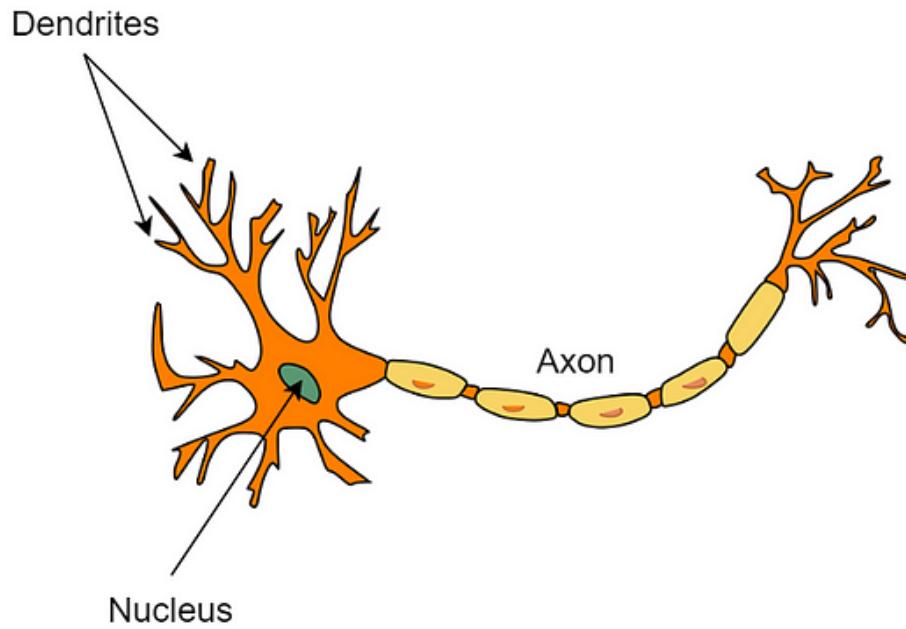


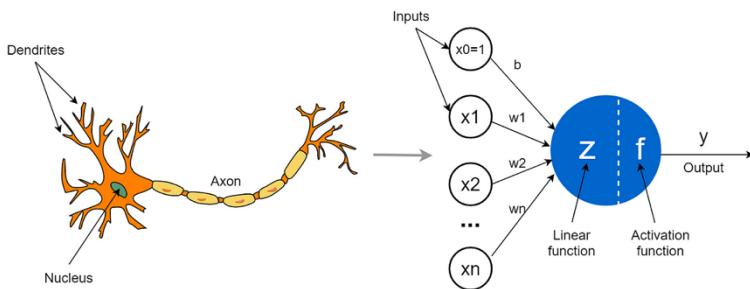
Key concepts

- **Artificial Neural Networks (ANN)**: Computational systems with interconnected units called artificial neurons.
- **Neurons**: Simple processors operating in parallel.
- **Connections**: Transmit signals from one neuron to another.
- **Signal**: A real number computed by a non-linear function of the sum of its inputs.
- **Weights**: Adjust the strength of the signal at a connection.
- **Layers**: Neurons are typically aggregated into layers, performing different transformations on their inputs.
- **Signal Flow**: Signals travel from the input layer to the output layer, possibly traversing layers multiple times.

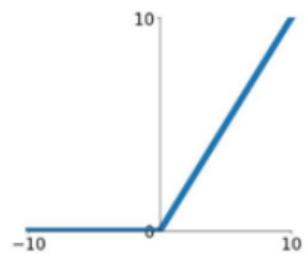
Artificial neural network

- A neural model is a mathematical model that simulates the behavior of neurons in the brain. It uses algorithms to learn from data and make predictions or decisions based on that data.

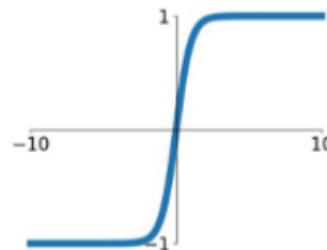




ReLU
 $\max(0, x)$



tanh
 $\tanh(x)$



Activation functions

- Sigmoid - vanishing, asymmetric
 - Tanh - vanishing, symmetric
 - Relu (rectified linear unit) - asymmetric, non-vanishing, but subject to death
 - Leaky relu - asymmetric, non-vanishing, undying
 - maxout - asymmetric, non-vanishing, double params
 - Elu (exponential linear unit) - asymmetric, non-vanishing, smooth
-
- Rule of thumb:
 - use **ReLU** for perception/hidden layer
 - **Tanh** or **sigmoid** for final/output with physical saturations, eg motor actions

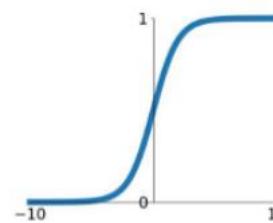
Activation functions

- A neural model is a mathematical model that simulates the behavior of neurons in the brain. It uses algorithms to learn from data and make predictions or decisions based on that data.

Activation functions

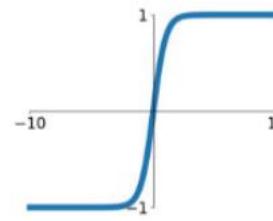
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



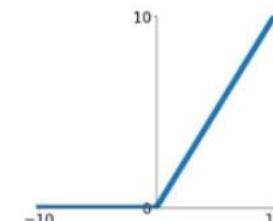
tanh

$$\tanh(x)$$



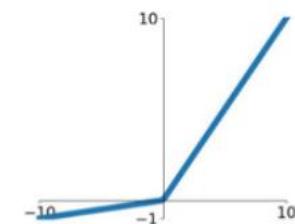
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

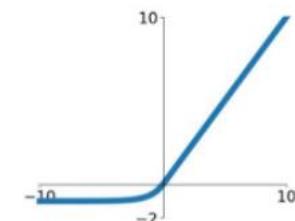


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

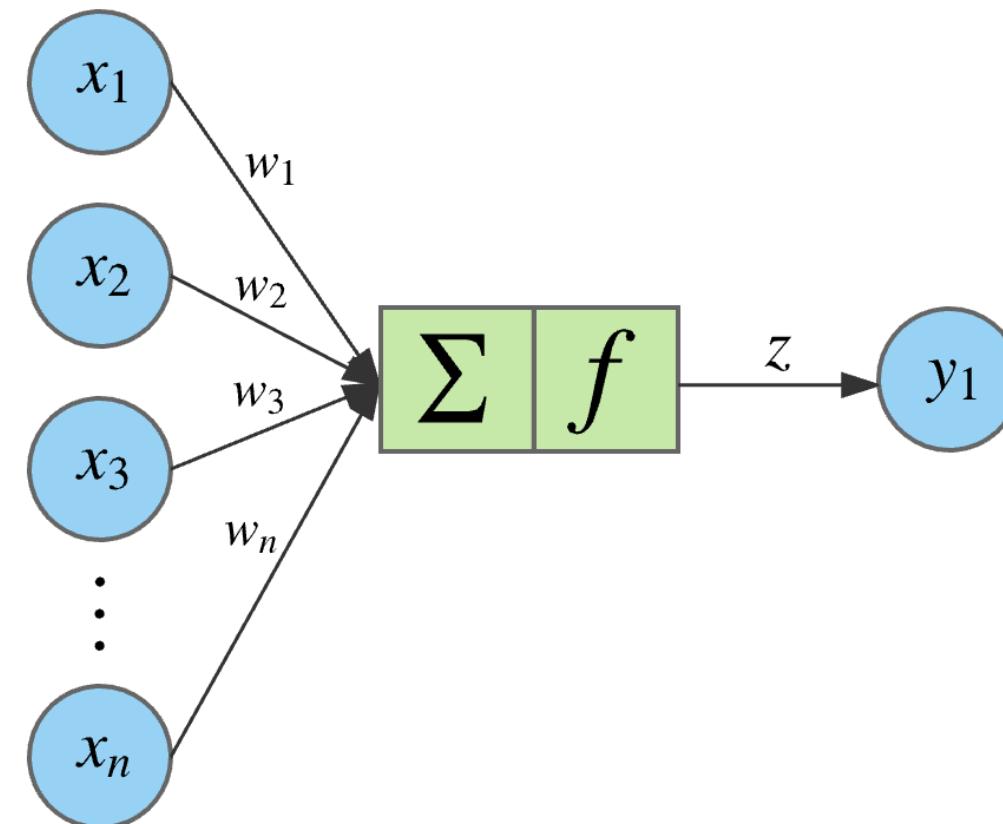
ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



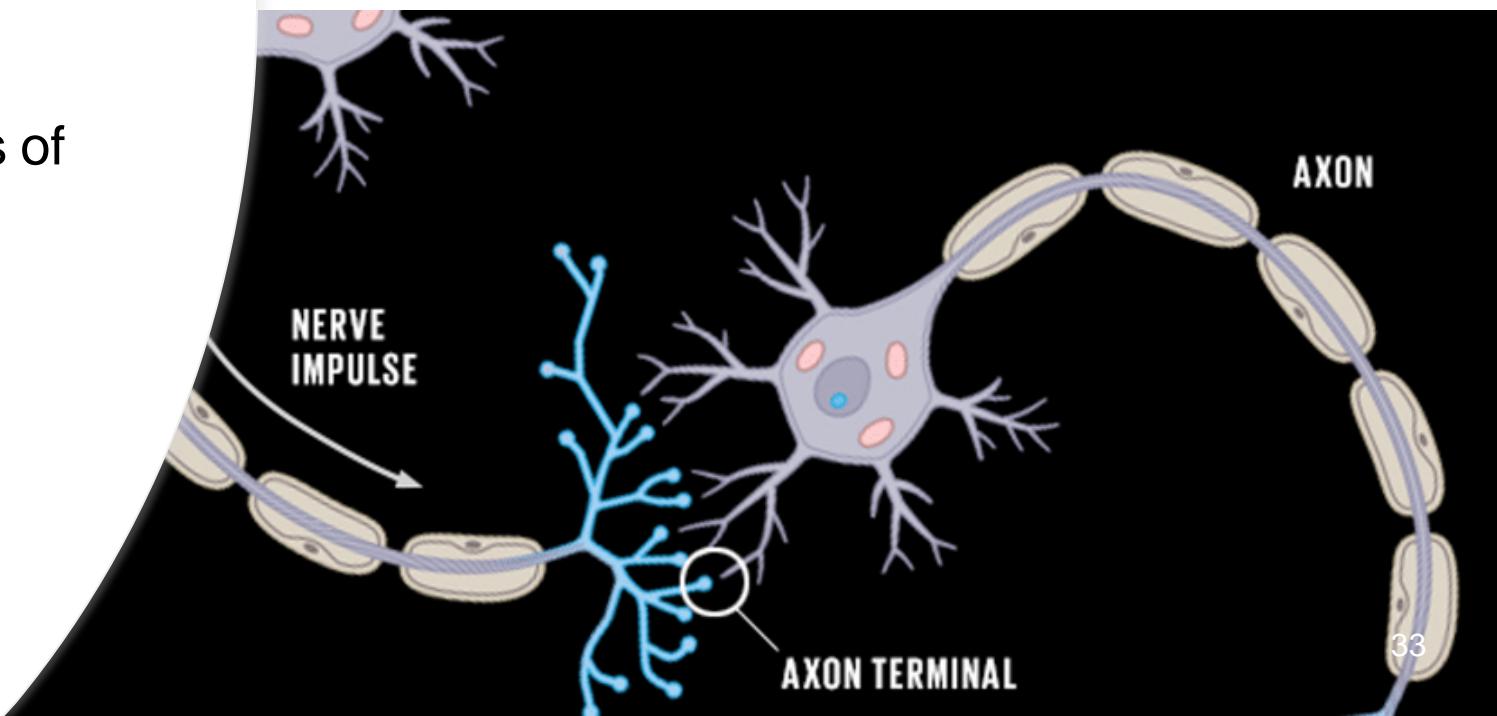
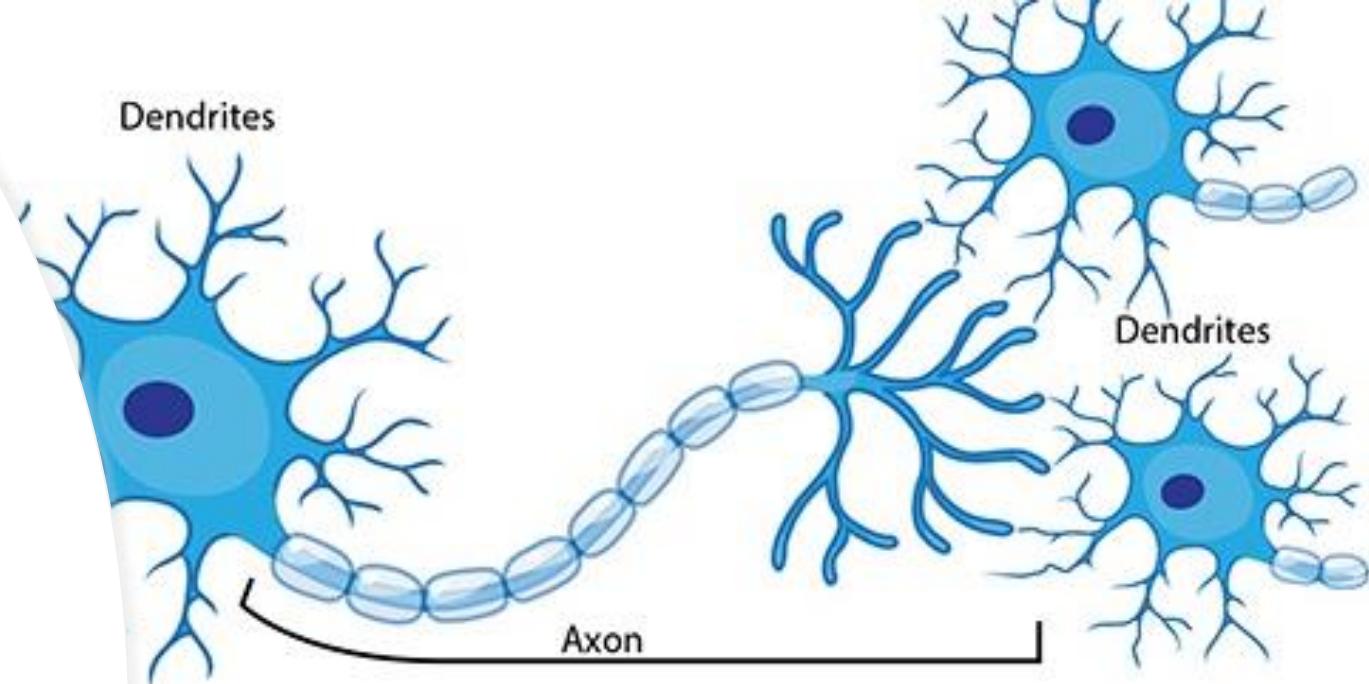
Multi-Layer Perceptrons

- Single perceptrons and equivalent linear models fit only a linear boundary
- Incapable of extracting more complex features in the original input space



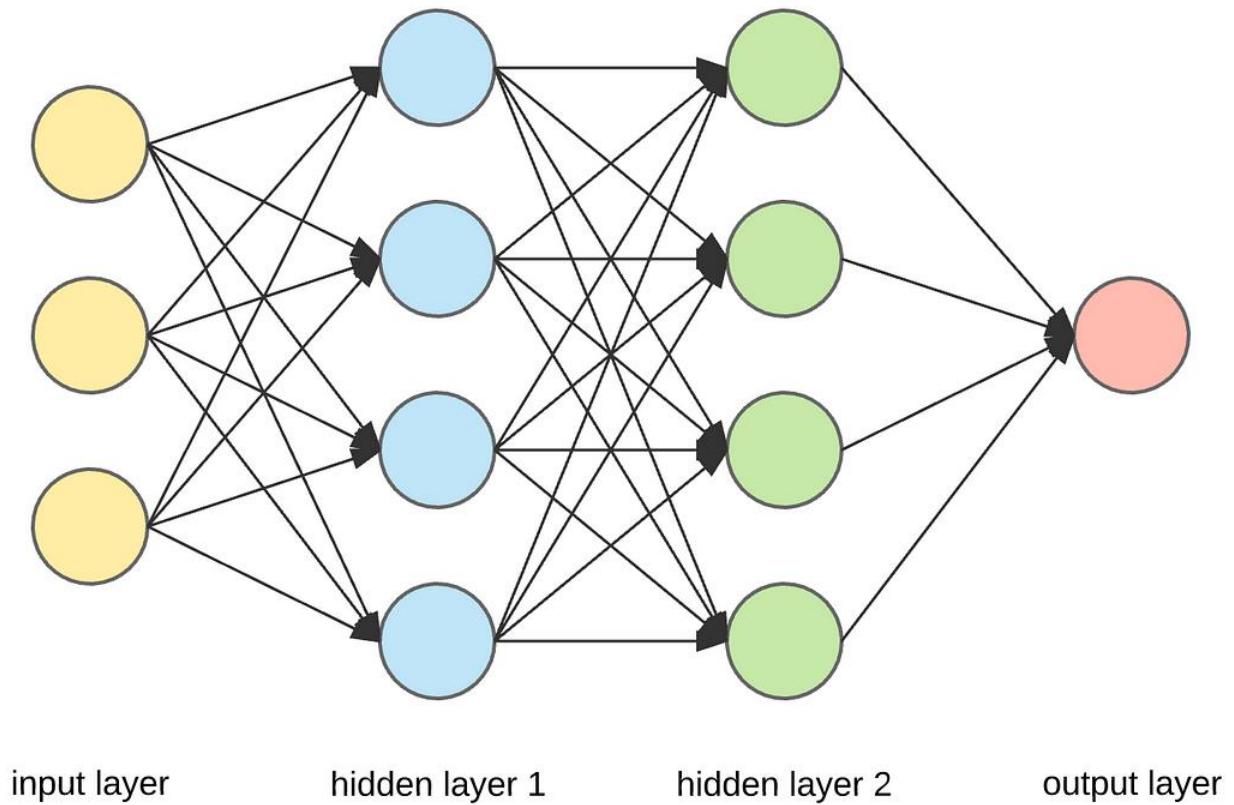
Interconnection between neurons

- The perceptron only sends signal in **one direction**, and it is called **Feed-Forward Neural Network**.
- The other type, **Recurrent Neural Network (RNN)**, can use their internal state to process sequences of inputs (so feedback is possible).

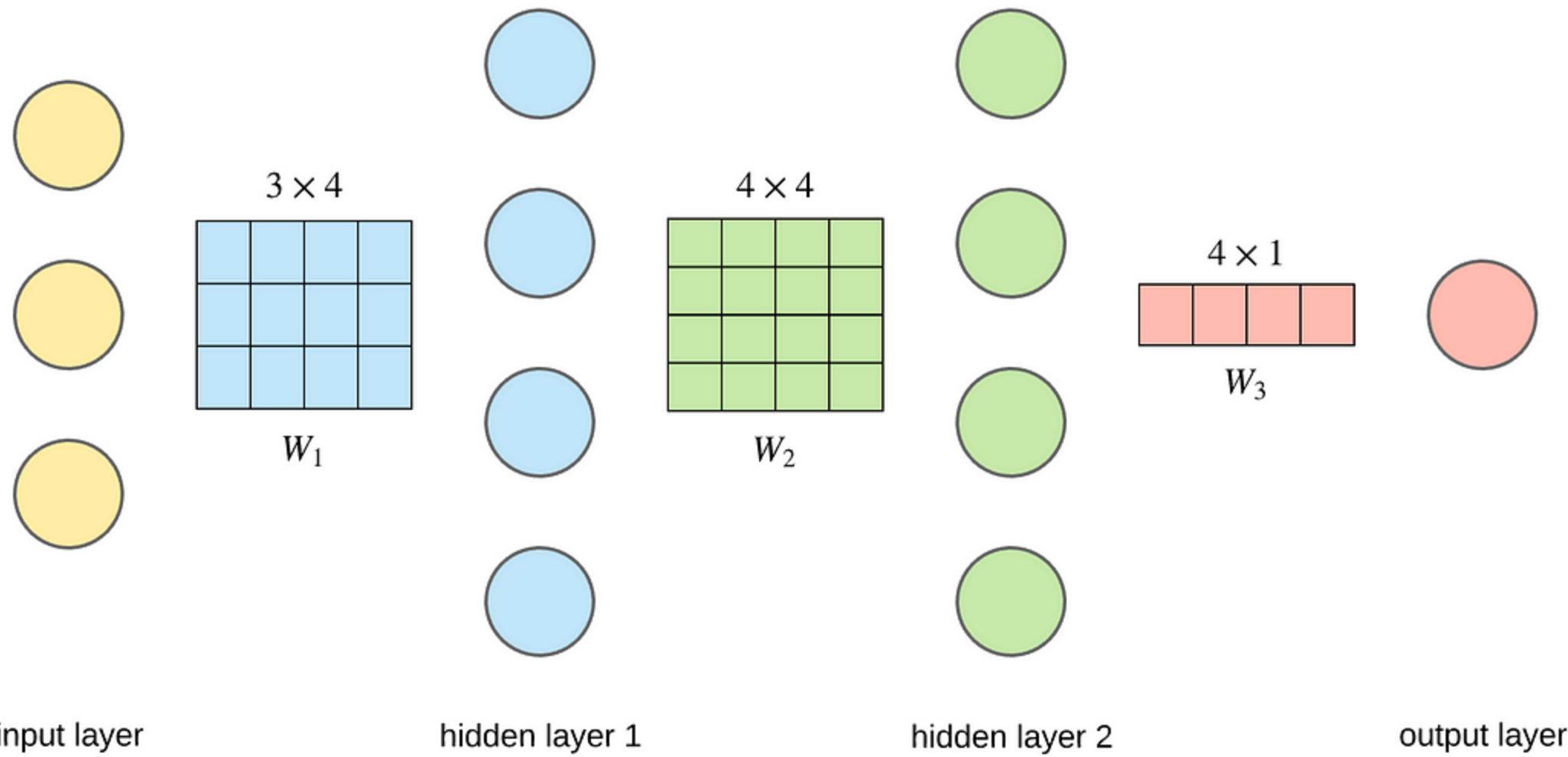


Hidden layers

- Single layer NN has limited “features” that can be aggregated
- What if we add an intermediate “hidden” layer of neurons that take the previous layer’s outputs as its inputs?
- → Multi-Layer Perceptron (MLP)



MLP & hidden layers



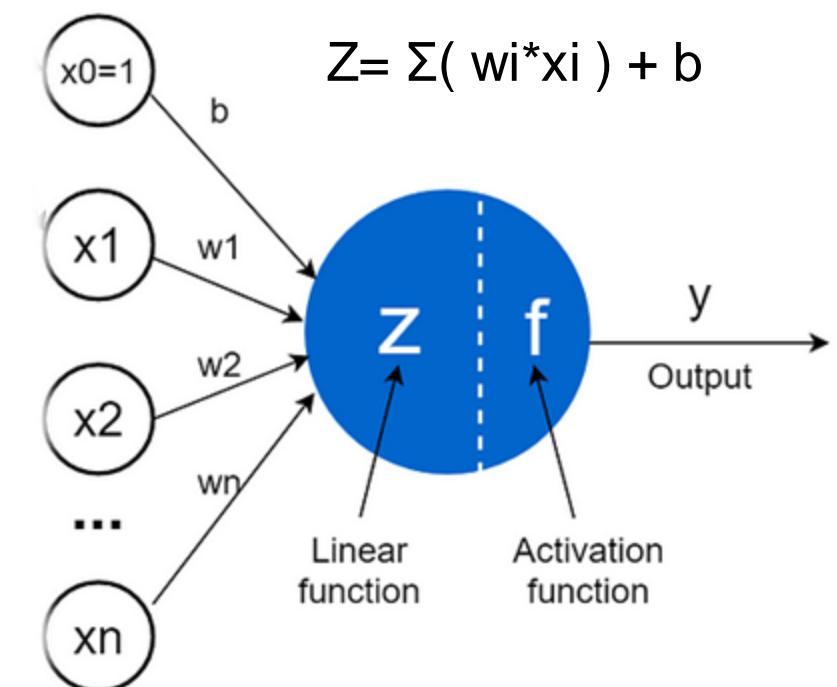
Updating weights/biases by backpropagation

- Gradients are computed using backpropagation, the weights in the neural network can be updated using **gradient descent** (refer the lecture: advanced optimization): adjust the weights in the direction that reduces the error.
- Rule to update the weight by gradient descent:

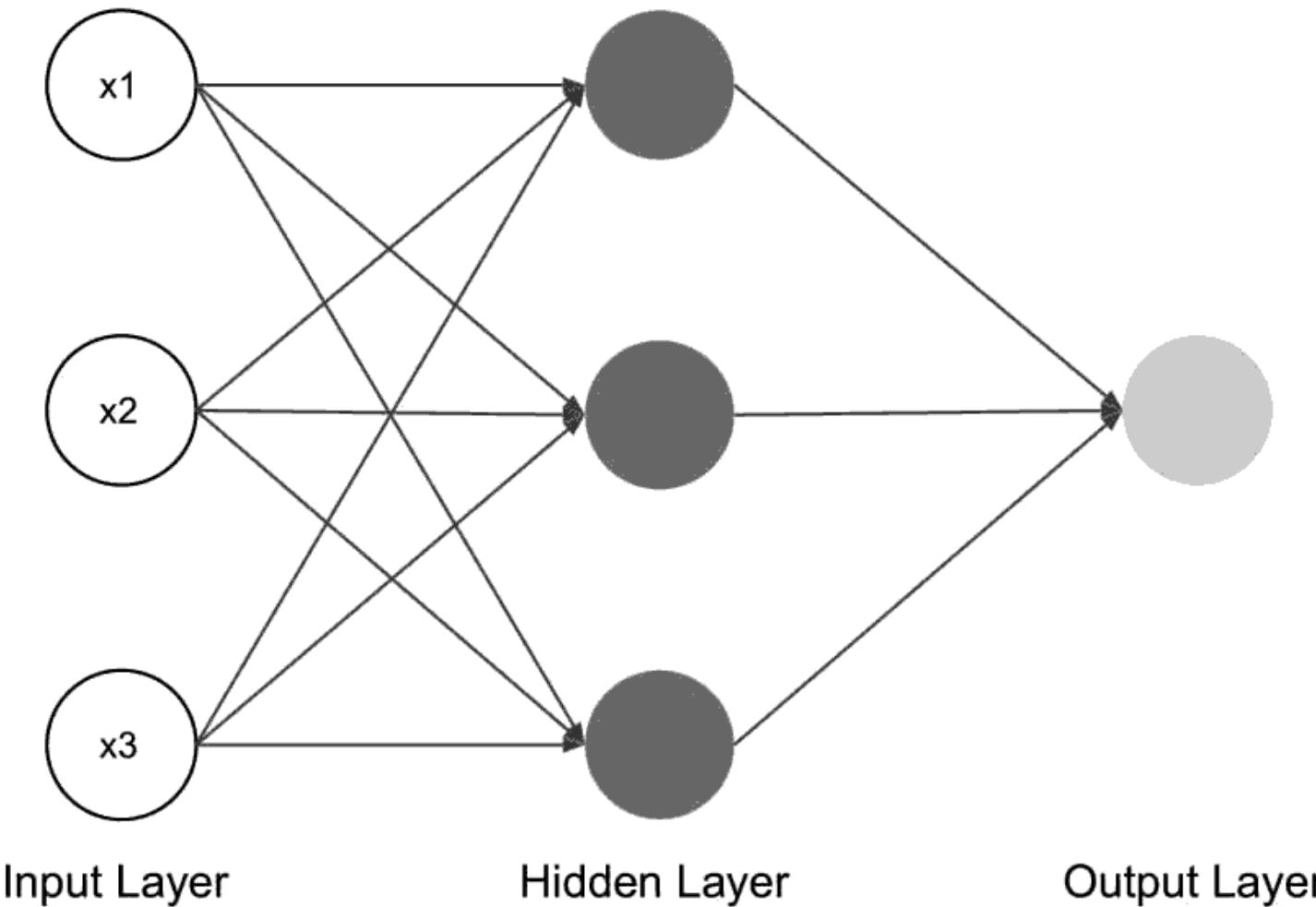
$$w = w - \alpha \cdot \frac{\partial E}{\partial w}$$

- Similarly, the bias can be updated as:

$$b = b - \alpha \cdot \frac{\partial E}{\partial b}$$



Feedforward



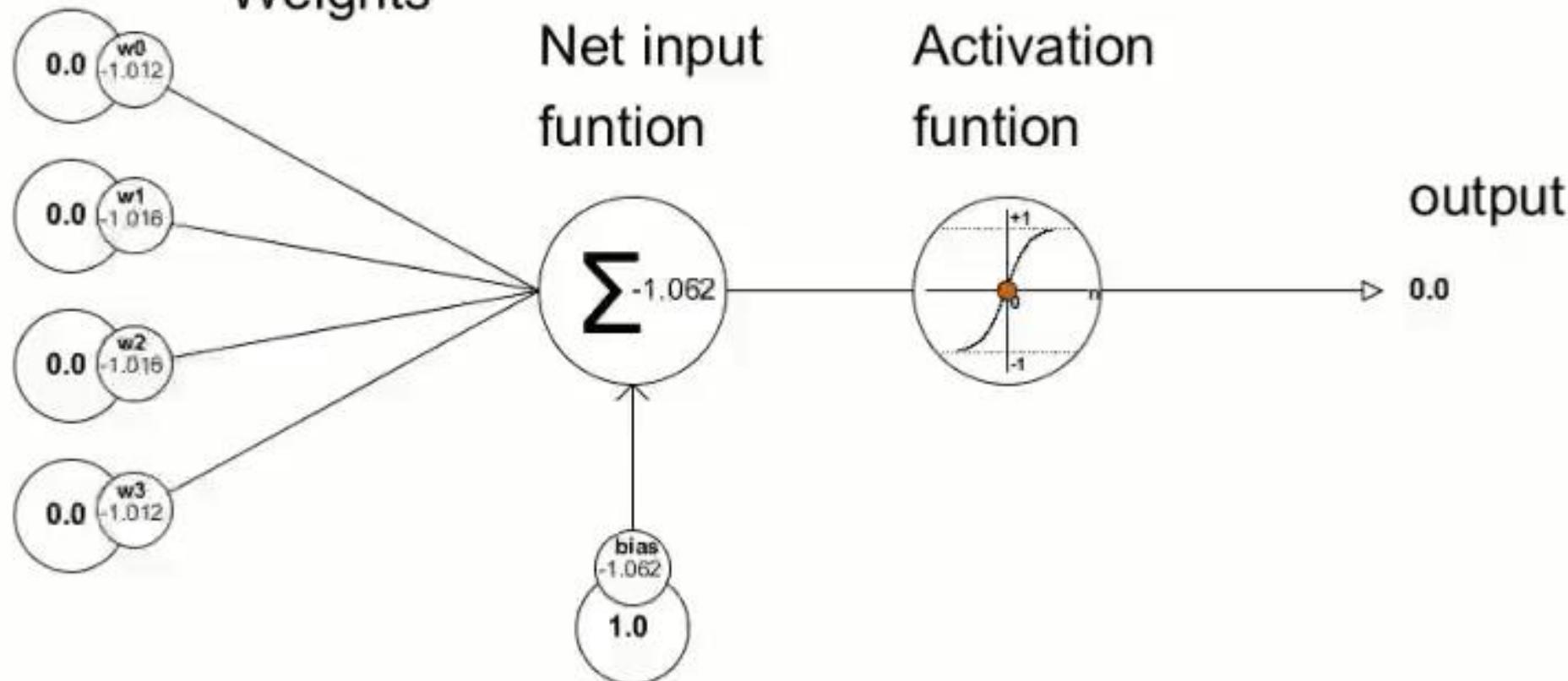
Inputs

Weights

Net input
function

Activation
function

output



Summary

- NNs are very versatile and capable
- Can learn arbitrarily complex decision functions and regression landscapes
- Tend to be black box, implicit and uninterpretable
- Need care to train (reward function, hyper parameters)
- Most of the machinery for deep learning has been in place since the 1980s
- Recent **renaissance** due to computational speed, data availability, and subtleties such as initialisation and activation

Summary

The linear model used in artificial neural networks is a **significant simplification** of the complex electrobiochemical processes in neural computation. The comparison between real neurons and the "weighted sum + non-linearity" of artificial neurons, such as perceptrons, is merely an analogy.

However, there is some validity to the connectionist model that suggests the **power of the brain lies in its connectivity**, as a substantial amount of the brain's processing and learning appears to occur at synapses.

(Please check out recent interview with Geoffrey Hinton, after he left Google:
<https://www.bbc.co.uk/news/av/world-us-canada-65453192>)

In essence, artificial neural networks only metaphorically resemble biological brains. Despite this, the metaphor has **proven to be quite fruitful and has led to numerous advancements in the field of artificial intelligence**.

Learning paradigms

Different learning paradigms

- Machine learning paradigms: the way how **algorithms learn from data**.
- Different learning paradigms:
 - **Supervised Learning:** Learning from labeled data, where the algorithm is trained using input-output pairs.
 - **Unsupervised Learning:** Learning from unlabeled data to find hidden patterns or intrinsic structures.
 - **Semi-Supervised Learning:** A hybrid approach using both labeled and unlabeled data for training.
 - **Reinforcement Learning:** learn optimal policies by interacting with the environment, collecting/observing the state-action pairs, receiving rewards or penalties. Unlike supervised and unsupervised learning, RL doesn't rely on labeled data.

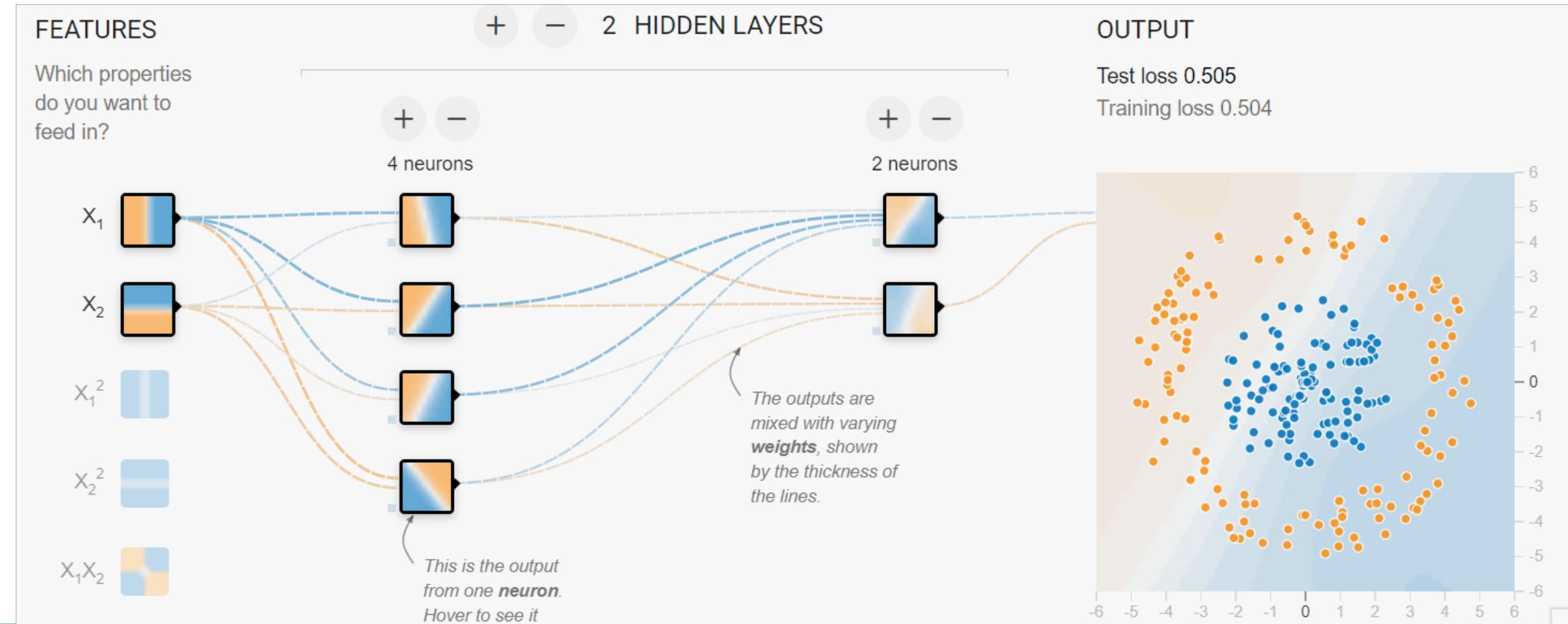
Training and inference

In machine learning, the process is divided into two main phases: **training** and **inference**.

- **Training phase:** a model is fed a dataset so that it can “learn” to recognize patterns, make predictions, and generally understand the dataset. The training phase is computationally **intensive** and requires a lot of computation.
- **Inference phase (deployment):** the trained model is used to make predictions based on new data. This is where the model applies the learned/trained neural network to new data. The inference phase is less computationally intensive than the training phase, but it needs to be fast and efficient, especially in real-time applications.

Supervised learning (using feedforward neural networks)

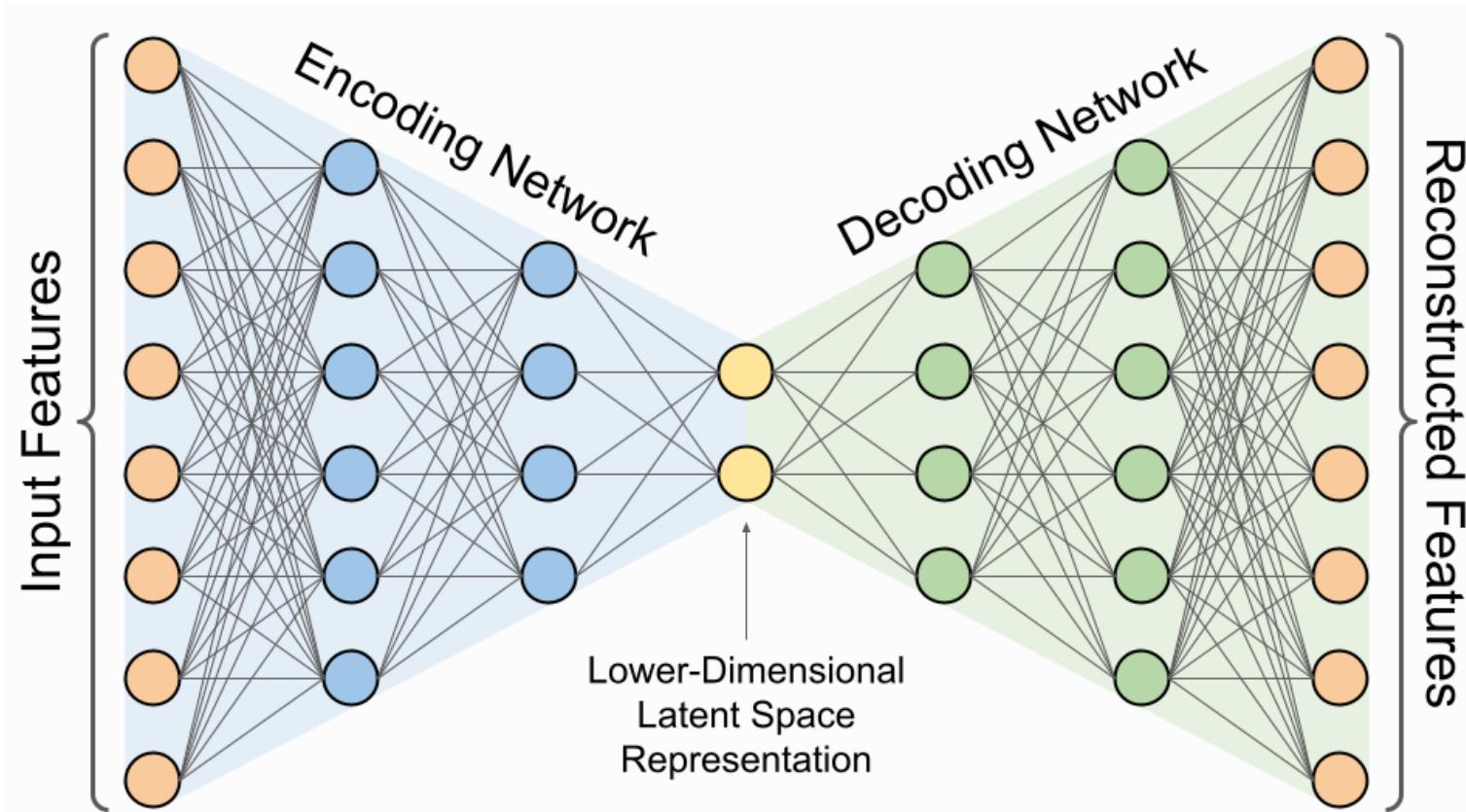
- Training neural networks to classify data (80% training, 20% testing)



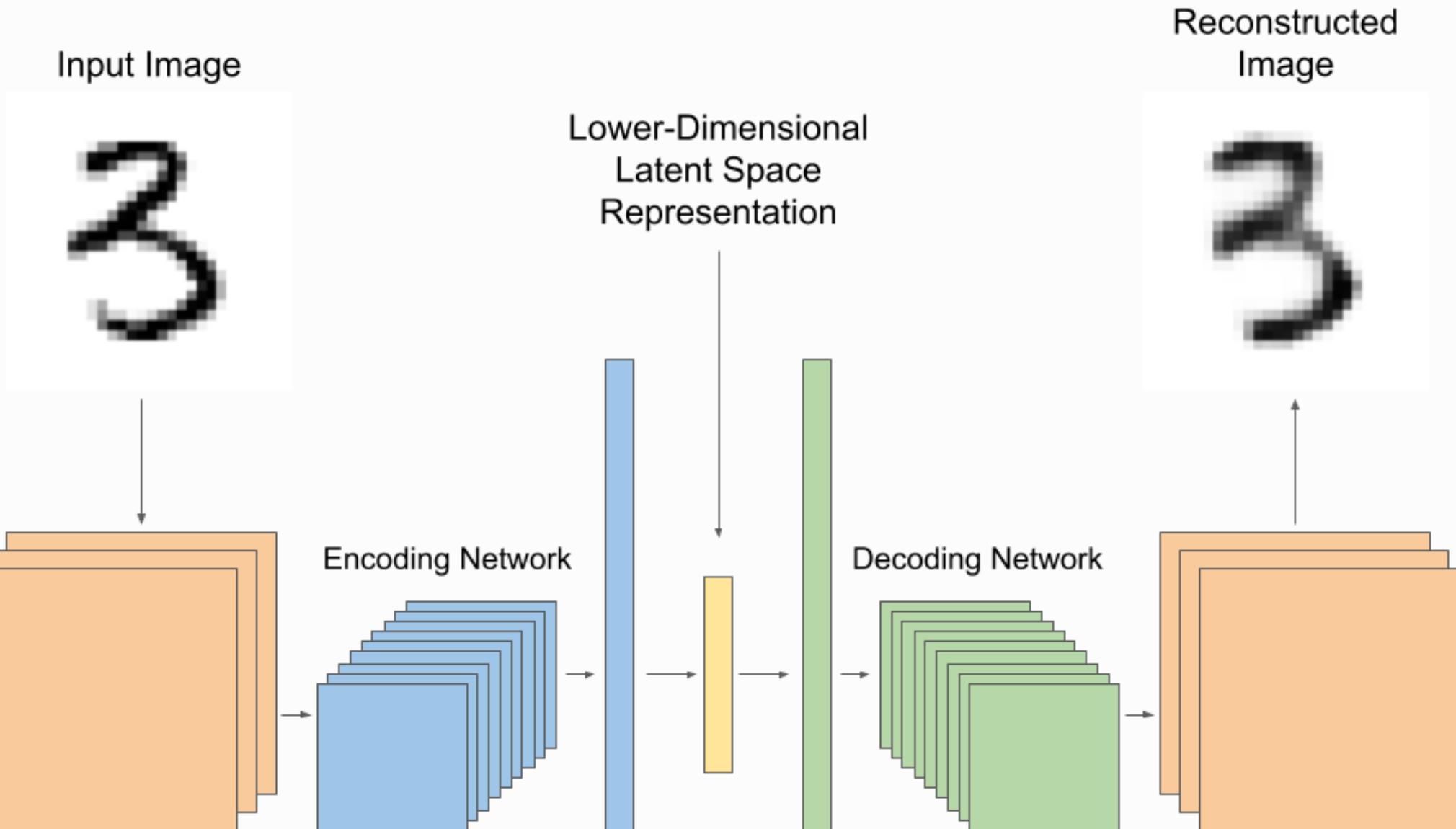
Unsupervised learning (Autoencoders)

Autoencoders

- Autoencoders are a type of neural network architecture that use their own inputs as labels, and reconstruct their input data at the output layer
- It learns to encode and decode the data, and to learn a compressed representation, called a **latent vector or latent space**, of the input data.



Autoencoders

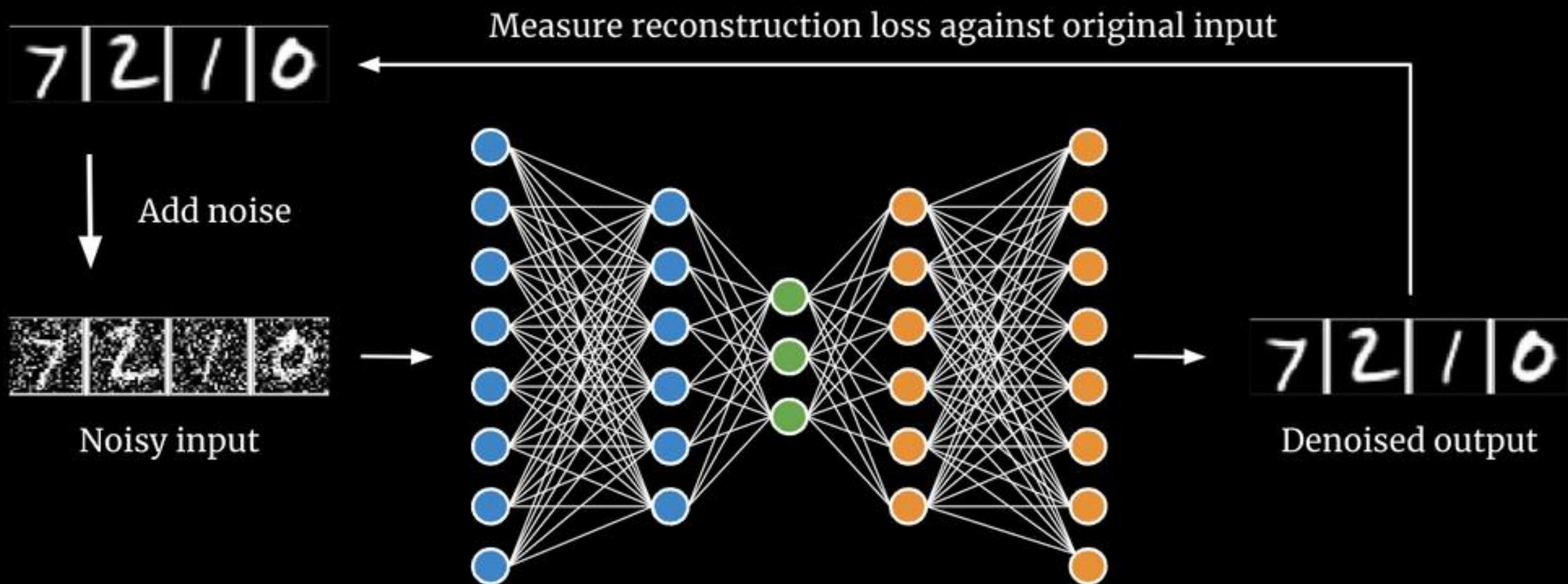


Denoising Auto Encoders (DAE)

- Denoising autoencoders are a specific type of autoencoder designed to **remove noise or corruption from input data**.
- Main idea: train the model to reconstruct the clean original data from a corrupted or noisy input.
- Usefulness: **noise removal and feature learning**.
- Applications in image denoising, speech enhancement, data pre-processing, and data quality improvement – removal of noise or reconstruct corrupted signals is essential for accurate analysis.

Denoising Auto Encoders (DAE)

- Autoencoders learn to capture the **essential features** of the input images, ignore the noises as unimportant features.

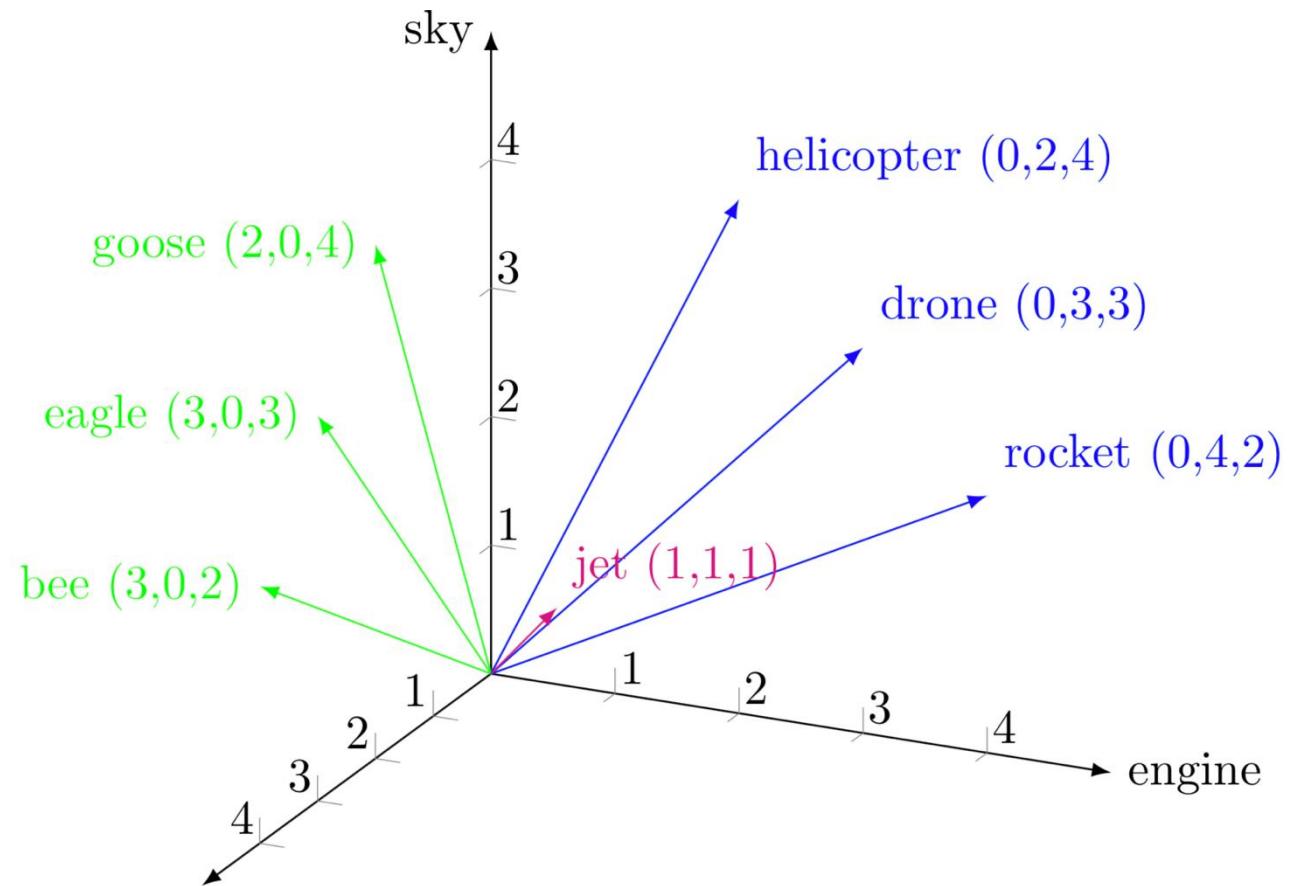


Let's see latent space
(feature embedding)
via NLP example



Text embedding

- Text embedding is the process of representing textual data in a numerical format.
- It captures semantic relationships between words and enables machine learning models to understand meaning and context.
- A numerical form that captures the semantic meaning

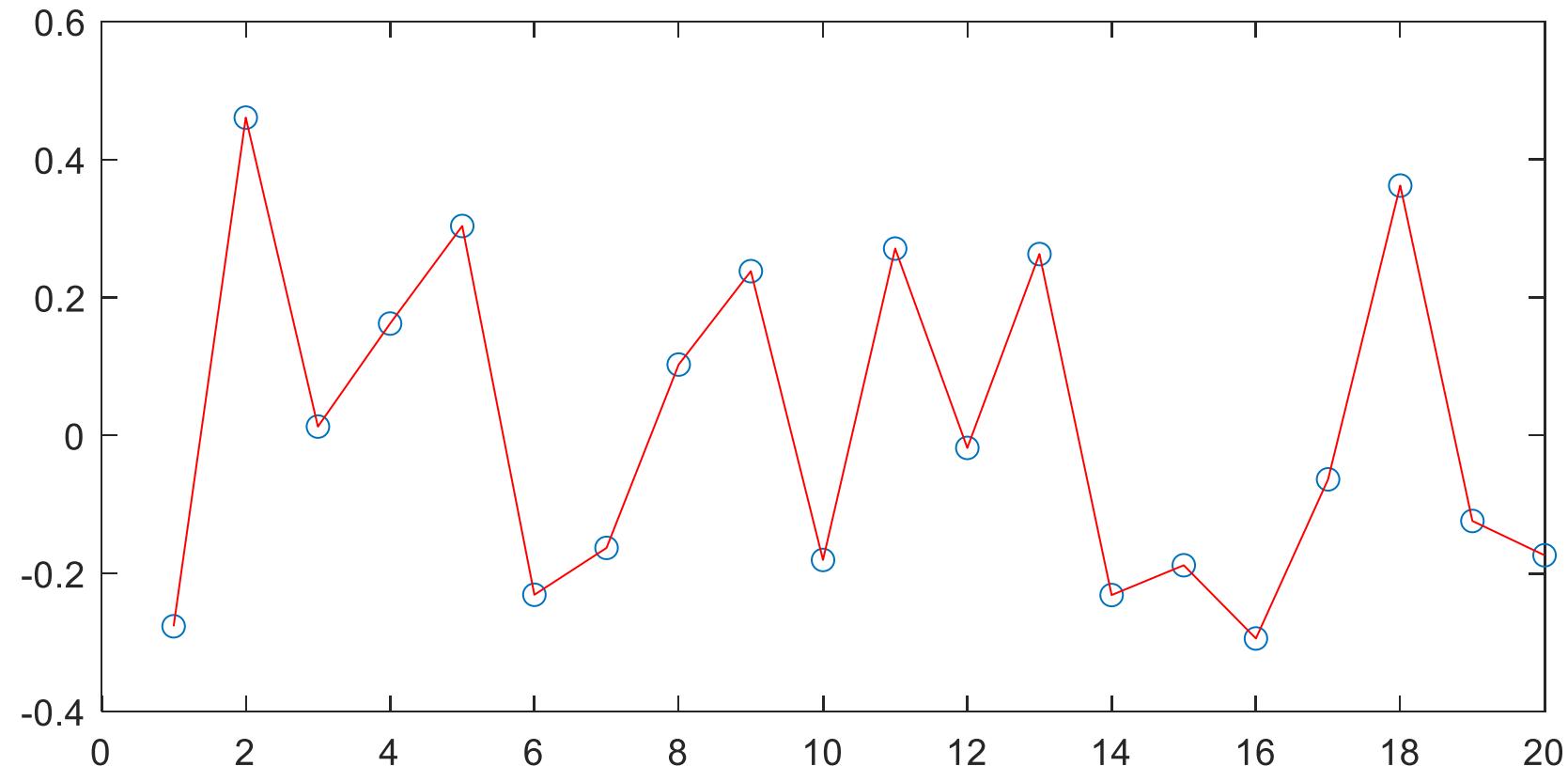


Text embedding

- Text embeddings represent words, sentences, or documents in a numerical form that captures their semantic meaning.
- Text embeddings provide a way to transform **raw text data** into **a continuous vector space (latent space / latent vector)**, where words or sentences with similar meanings are located close to each other.
- Typical models: Word2Vec, GloVe, or BERT, that learn to encode words or sentences into dense **numerical representations**, taking into account their contextual relationships and semantic similarities.
- Each embedding vector corresponds to a word, sentence, or document.
- Once trained, these embeddings can be used as input features for various NLP.

Intuitive example: text embedding

- What is the meaning for information of the following data?
- [-0.2765046, 0.4608382, 0.0129819, 0.162292, 0.303784, -0.2308206, -0.1628694, 0.1028627, 0.2382612, -0.1803477, 0.2709867, -0.0179272, 0.2630633, -0.2312355, -0.1880319, -0.2941769, -0.0634862, 0.3623612, -0.1238407, -0.1735657]



Intuitive example: text embedding

- Decompressed this **latent vector** using BERT-based sentence encoding method:

“Life is a quality that distinguishes matter with biological processes from matter without, and is characterized by the capacity for growth, reaction to stimuli, metabolism, energy transformation, and reproduction. Various forms of life exist, such as plants, animals, fungi, protists, archaea, and bacteria, and biology is the science that studies them.”

Reward function

Reward function

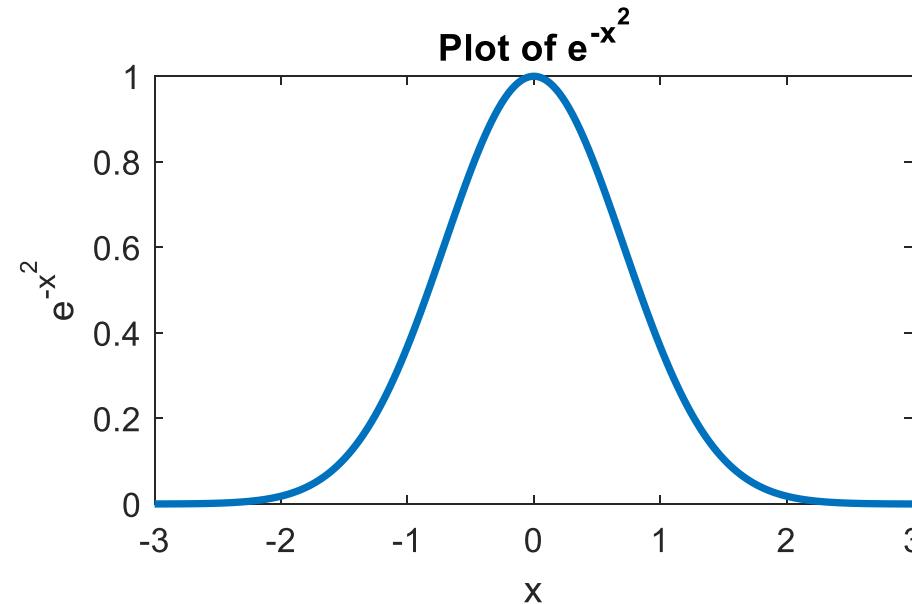
- In machine learning, especially in RL, a reward is a feedback signal (numerical quantities) from the environment to the agent, indicating the quality and suitability of its actions (positive or negative).
- **Reward function:** how rewards are assigned based on the agent's actions and the environmental state.
- **Goal:** learn an optimal policy that maximizes cumulative rewards over time.
- The reward function typically involves different physical quantities, where each term/quantity referred as **reward terms**.
- Overall, reward function guides the agent's learning of exploration and exploitation, encouraging positive actions and discouraging negative ones.

Reward design

- Previously, in the inverse kinematics (IK) problem, we have
- Objective function for optimization – as a **minimization objective**

$$f(\mathbf{q}) = \|\mathbf{F}(\mathbf{q}) - \mathbf{y}^d\|_{QP}^2 + \|\mathbf{O}(\mathbf{q}) - \mathbf{vec}^d\|_{Qr}^2 + \|\mathbf{q} - \mathbf{q}_0\|_{Qj}^2$$

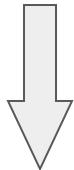
- **Quiz:** so, what the reward should be, if we would like to use reinforcement learning to learn the optimal policy? Ie, how to convert to **maximization** problem?



Reward design

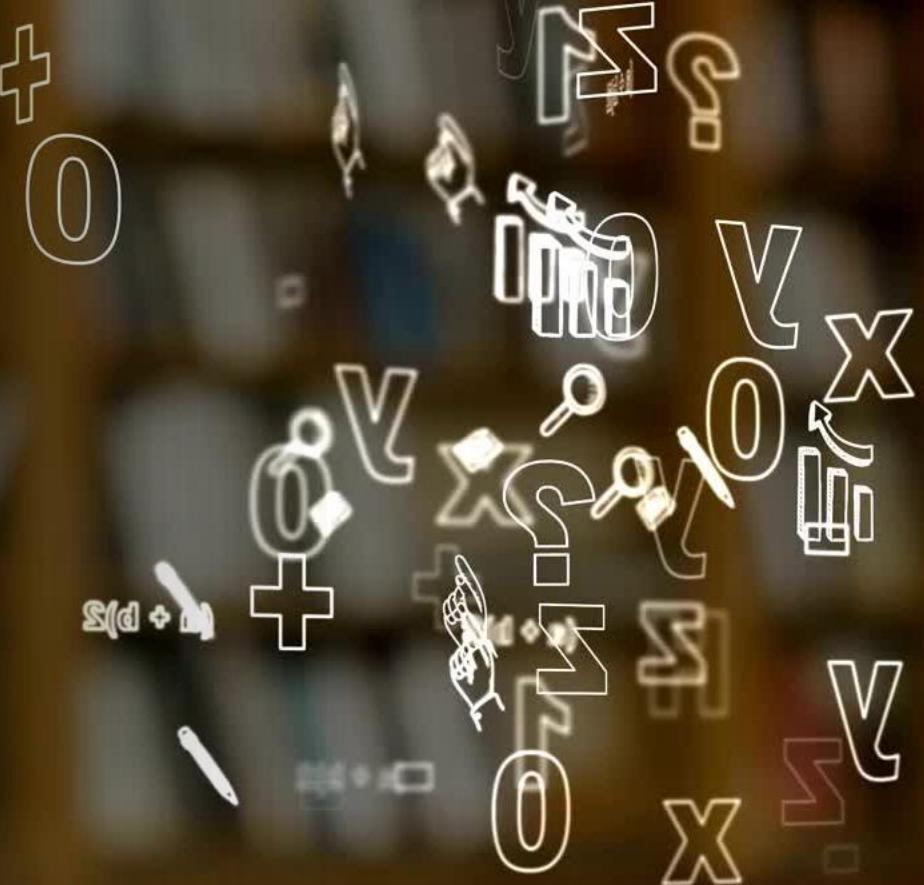
- Previously, in the ***optimization*** lecture (the inverse kinematics problem), we have the **Objective Function** for optimization

$$f(\mathbf{q}) = \|\mathbf{F}(\mathbf{q}) - \mathbf{y}^d\|_{Q_P}^2 + \|\mathbf{O}(\mathbf{q}) - \mathbf{vec}^d\|_{Q_r}^2 + \|\mathbf{q} - \mathbf{q}_0\|_{Q_j}^2$$

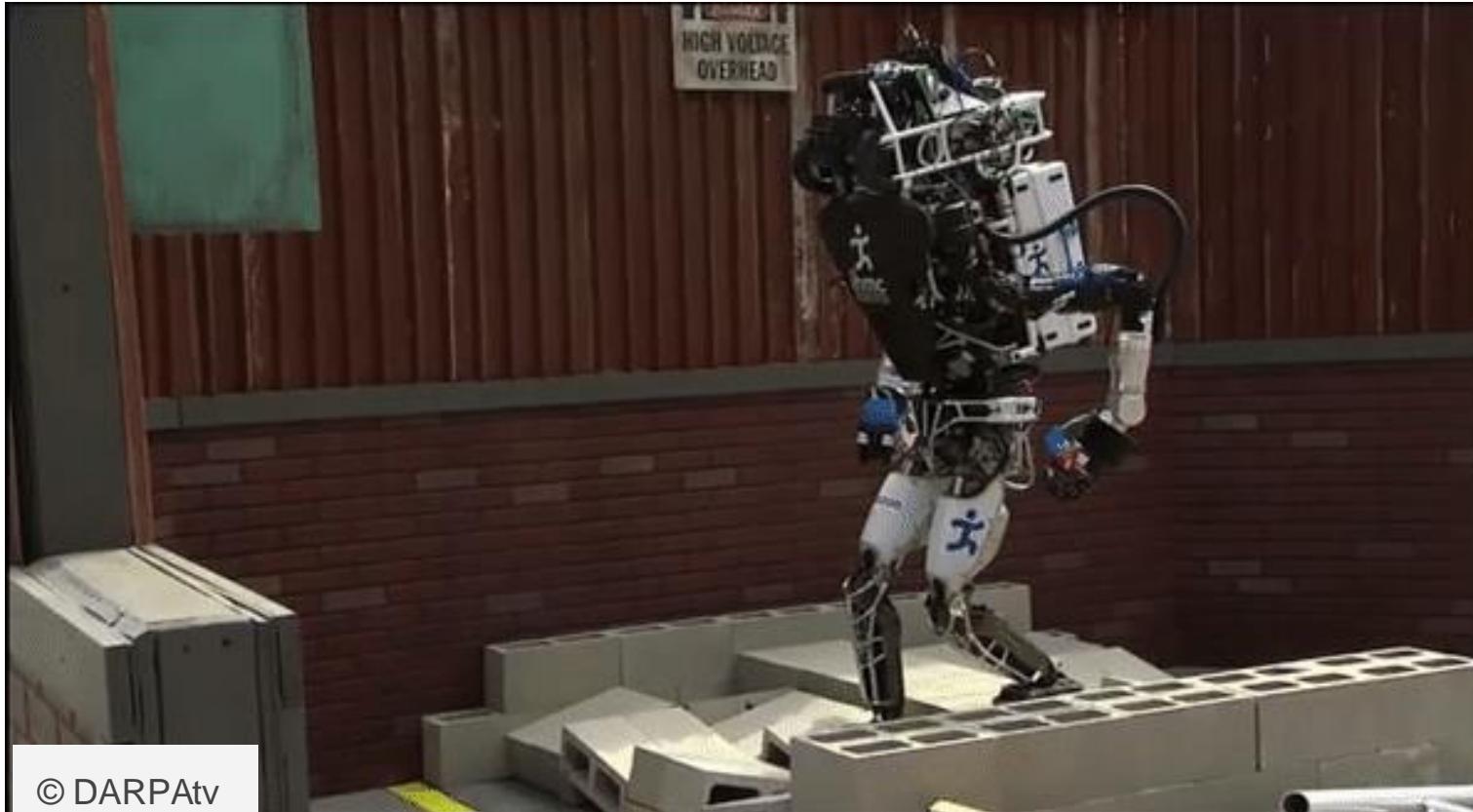


$$\mathbf{R} = Q_p e^{-(F(q) - y^d)^2} + Q_r e^{-(O(q) - \text{vec}^d)^2} + Q_j e^{-(q - q_0)^2}$$

Real case studies
& applications of
different learning
techniques



What are missing?



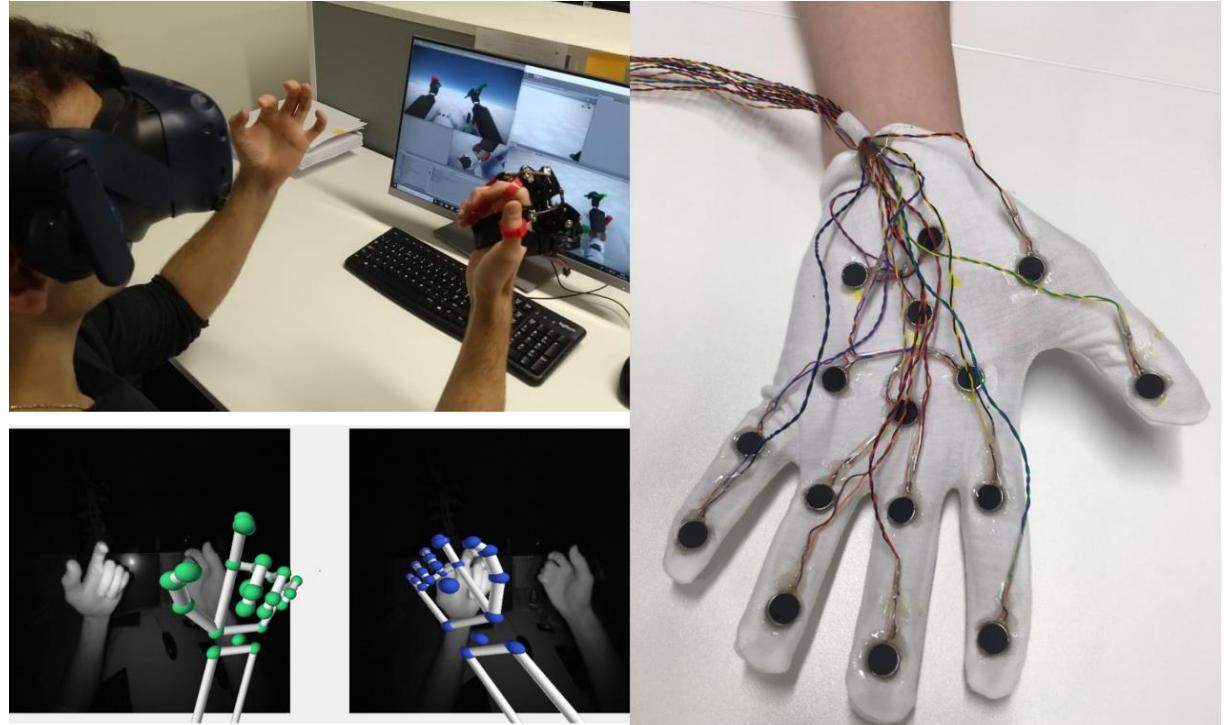
1. Operators manually put footsteps for robots, which introduces human errors
2. No use of hands during locomotion
3. No reaction while falling, no autonomous behaviors
4. :
- :

Supervised learning (from demonstration)

1

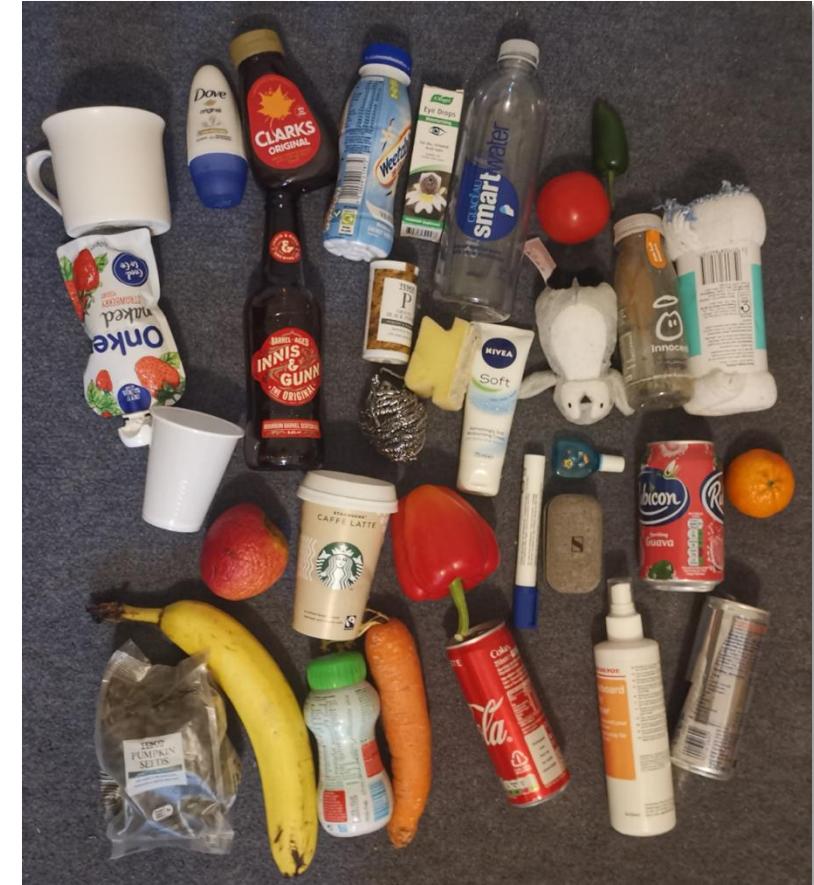
Learning from demonstrations

- Immersive tele-manipulation to demonstrate manipulation tasks
- Optical hand tracking to map human-robot motions
- Additional haptic gloves for contact sensing
- Generate demonstration data



Learning Adaptive Grasping From Human Demonstrations,
IEEE/ASME Transactions on Mechatronics, (Volume: 27, Issue: 5, October 2022)
<https://ieeexplore.ieee.org/abstract/document/9714723>

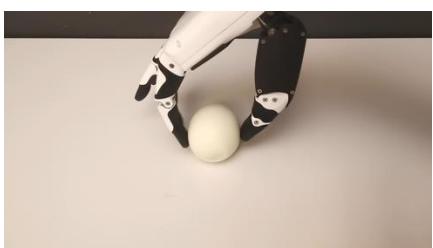
Learning from few demonstrations



Collect dataset for training: input-output pairs

Autonomous grasping by learning

- To achieve **autonomous, continuous, adaptive** grasping of a wide range of objects in different **shapes, stiffness**.



Extracting human policies: self-adaptive grasping

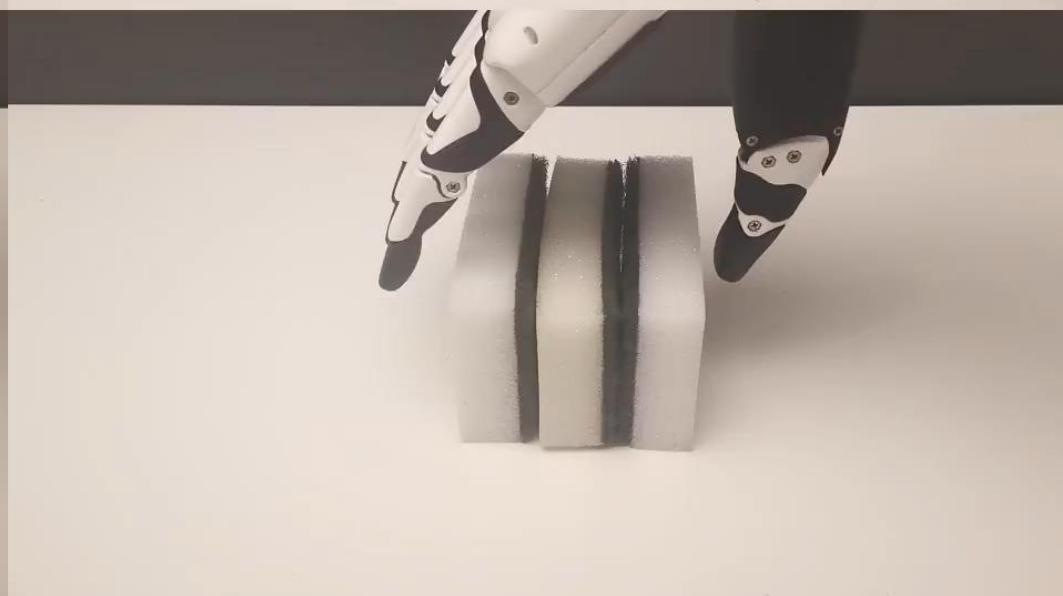
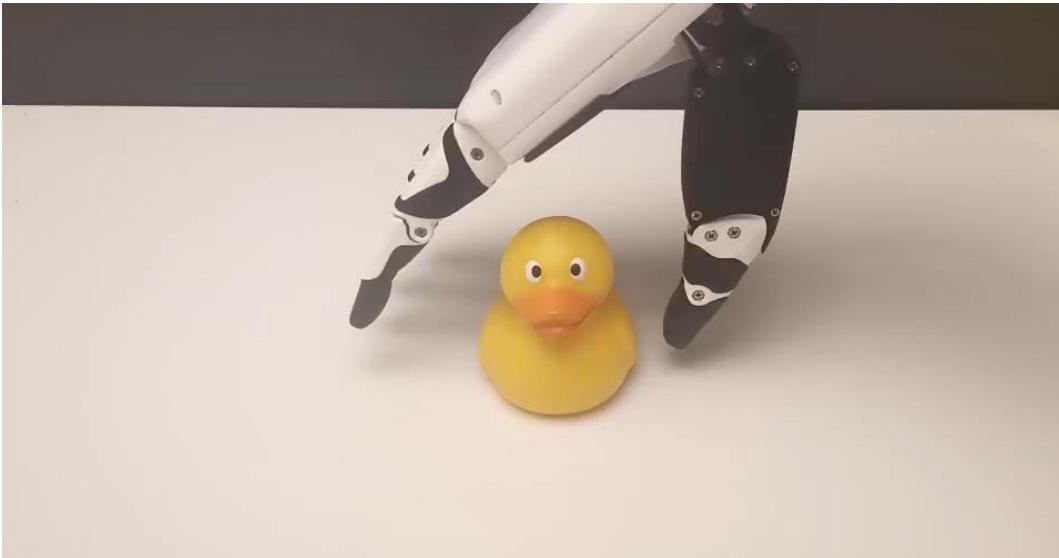


Pre-programmed Grasping Control



Self-adaptive Grasping

Grasping a wide range of unseen objects





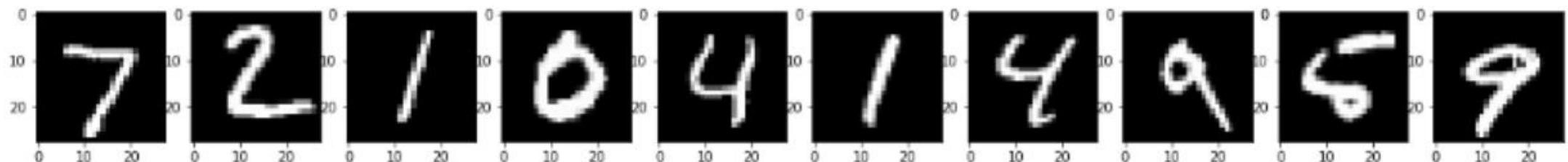
Unsupervised learning



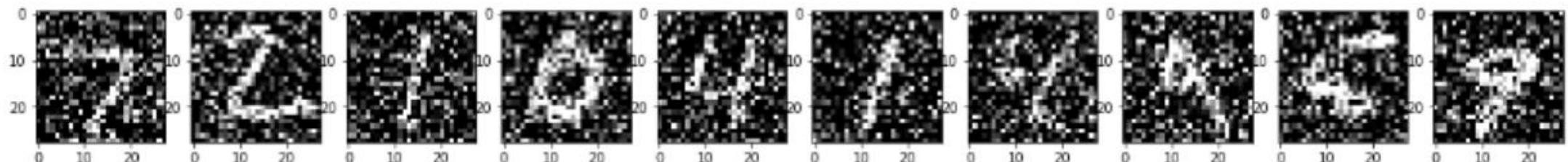
Denoising Auto Encoders (DAE)

- For DAE, the autoencoder is specifically trained to reconstruct the original, clean image from the noisy version, effectively **denoising** the image.

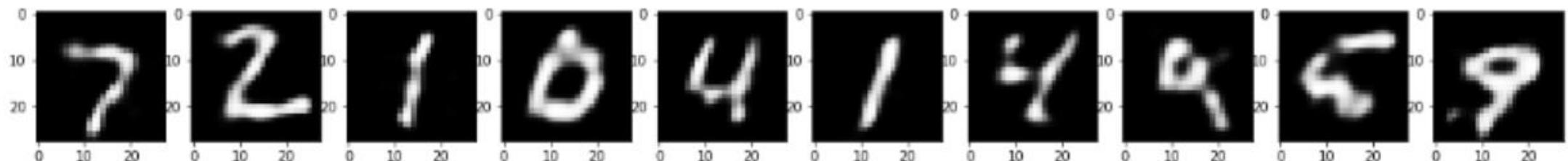
Original Images



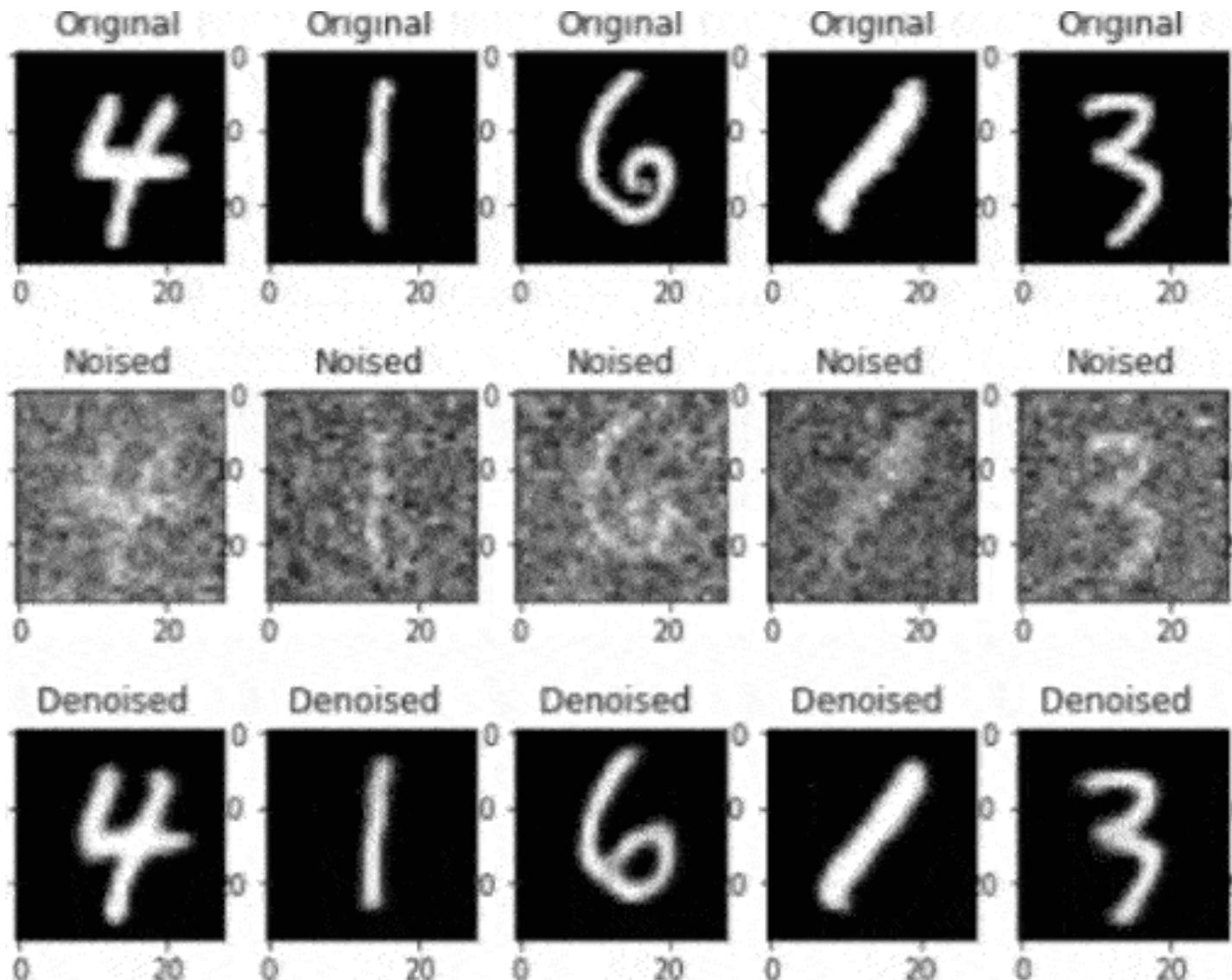
Noisy Images



Reconstruction of Noisy Images



Convolutional Denoising Autoencoder – example





Reinforcement learning



Signal processing in learning based motor skills

AAAS [Become a Member](#)

Log In [ScienceMag.org](#) Search

Science Robotics

Contents ▾ News ▾ Careers ▾ Journals ▾

Read our COVID-19 research and news.

Teaching a robot dog new tricks

Hierarchical motor control using deep reinforcement learning

KAI YUAN AND CHRISTOPHER MCGREAVY

A robotic dog (Atlas) is shown walking alongside a real dog. The robotic dog has sensors attached to its body. A close-up view of the robotic dog's leg mechanism is also shown on the left side of the main image.

About The Cover

Science
Robotics

DECEMBER 2020

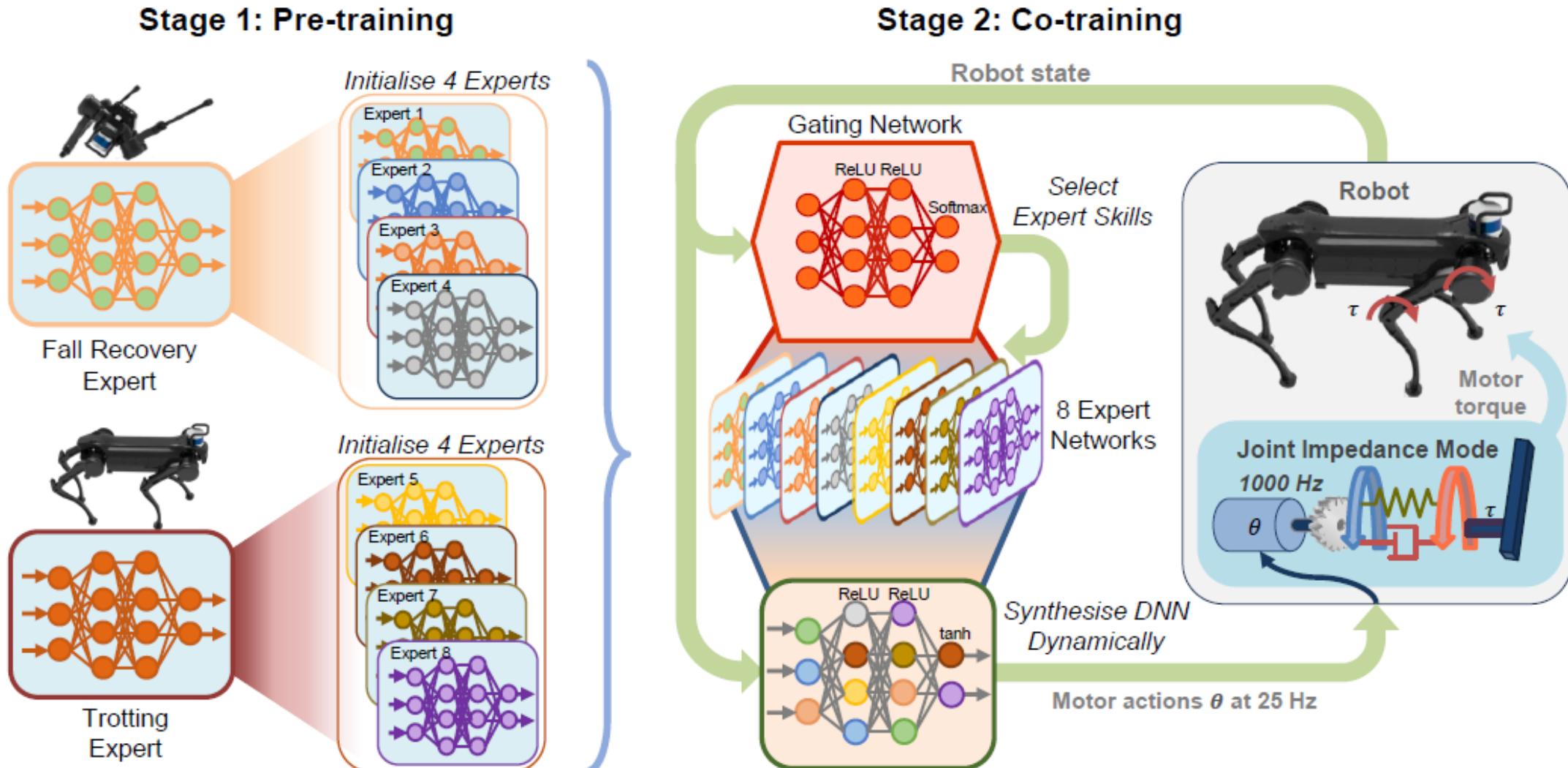


AAAS

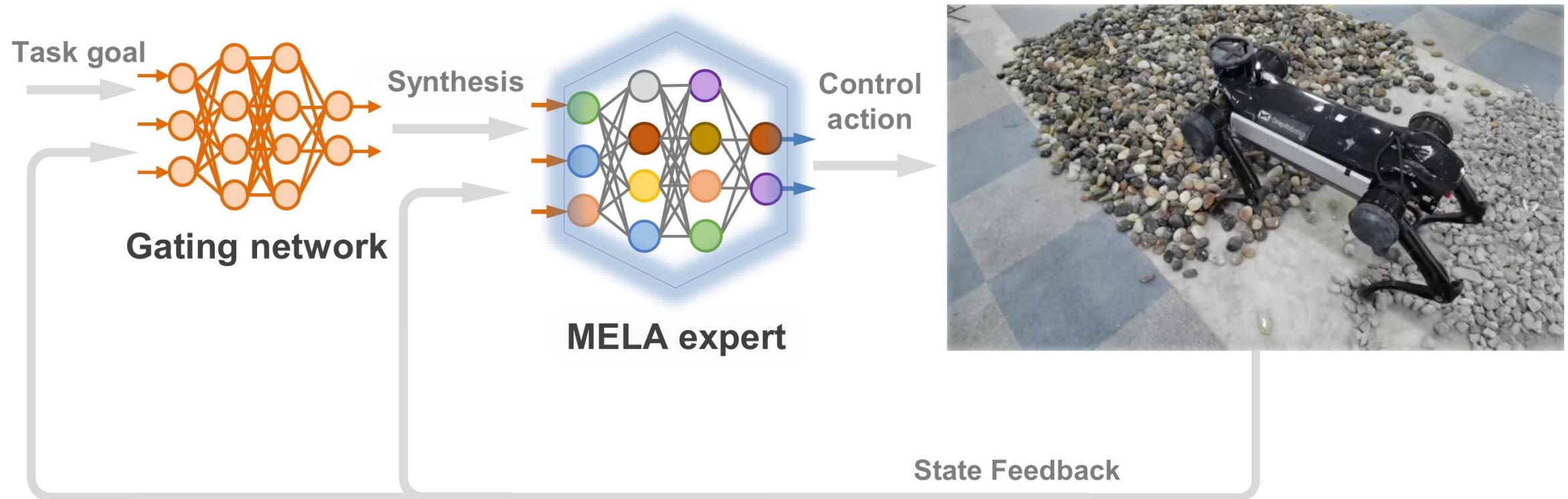
Multi-Expert Learning of Adaptive Legged Locomotion, Science Robotics, vol 5 (49), 2020

<https://www.science.org/doi/full/10.1126/scirobotics.abb2174>

Two-stage MELA Training



Multi-Expert Learning Architecture (MELA)



Multi-expert Learning of Adaptive Legged Locomotion, **Science Robotics**, vol 5 (49), 2020

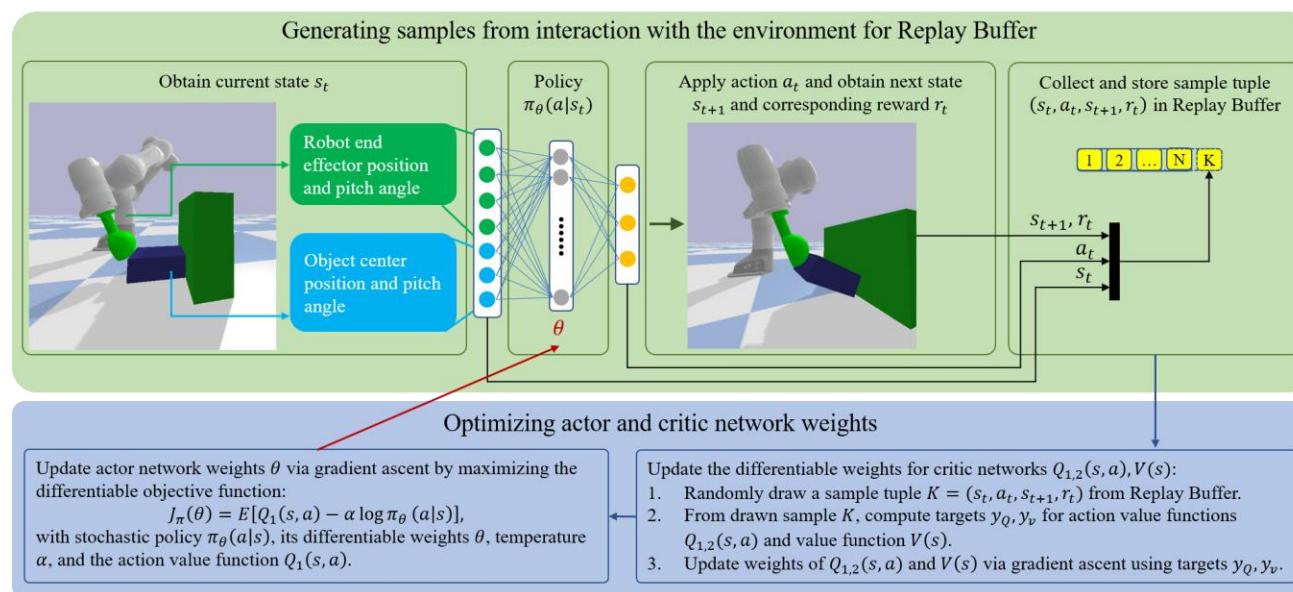
Real-world deployment



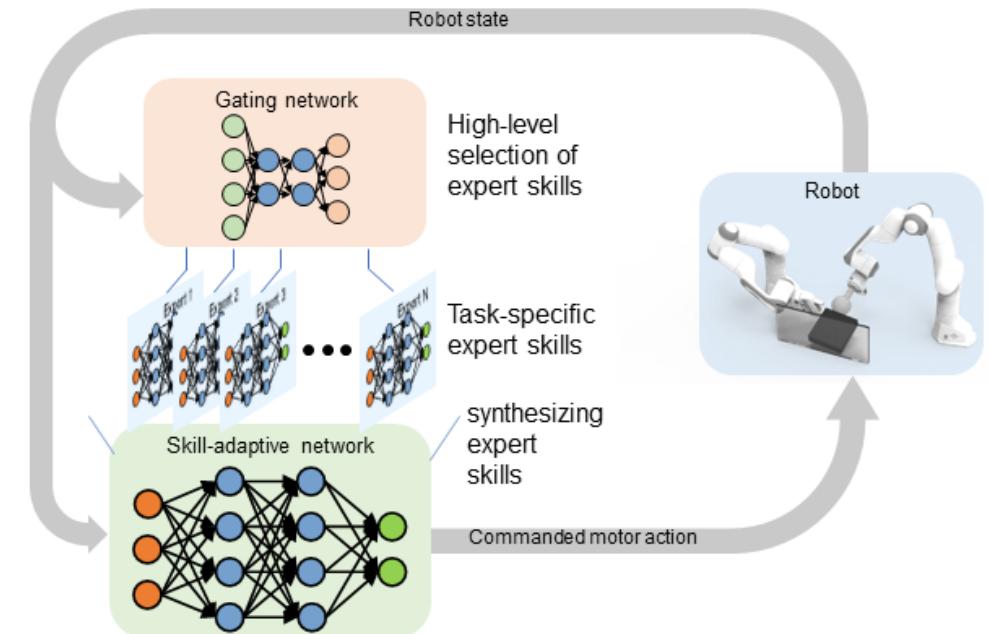
Real-world scenarios are challenging with large uncertainties

Two-stage RL training of a collaborative policy

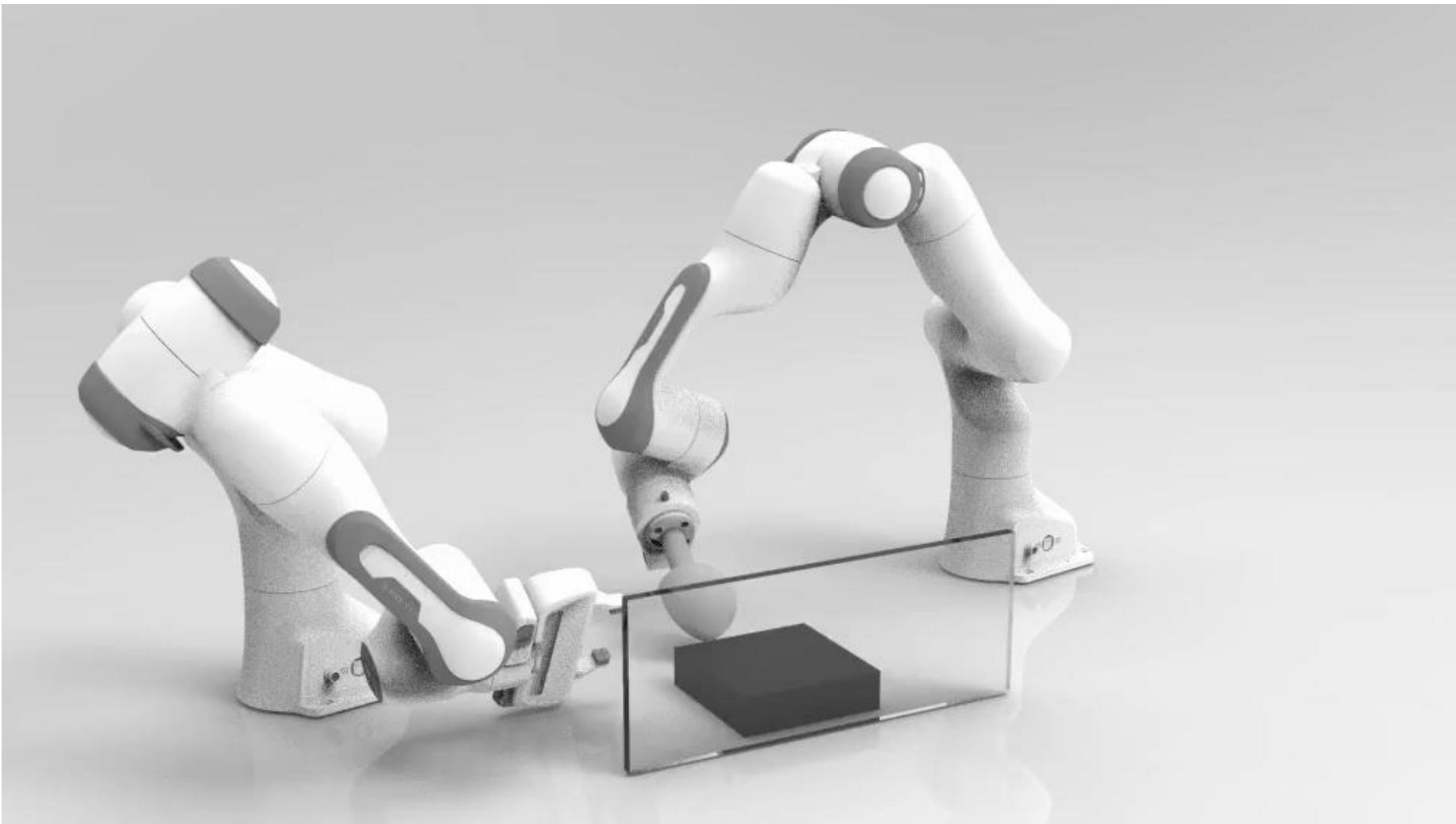
Stage 1: pretrain experts via Deep Reinforcement Learning



Stage 2: co-train experts in Multi-Expert Learning Architecture



Collaborative manipulation & grasping



Multi-expert synthesis for versatile locomotion and manipulation skills
Frontiers in Robotics and AI 9, 970890
Free access: <https://www.frontiersin.org/articles/10.3389/frobt.2022.970890/full>

Takeaway messages

- Definition and concepts (AI, ML, Deep learning - similarities and differences)
- Turing Test
- Common learning paradigms/techniques (4 types)
- Fundamentals of neural networks:
 - Activation functions
 - Backpropagation
 - Reward design
- Understand the basic use of suitable techniques to solve the problems