Introduction to Artificial Neural Networks

First introduced in 1943.
Looking at the brain's architecture on how to build an intelligent machine, which is sparked by artificial neural network. Machine learning models inspired by the networks of biological neurons found in our brains.

The idea of using biological analogies, such as neurons, in artificial intelligence (AI) has been a topic of debate.

ANN's are the core of deep learning.

Limitations of biological analogies

- **Oversimplification**: Biological systems are complex and cannot be fully replicated in AI systems.
- **Constraints**: Biological analogies can limit the design of AI systems, constraining them to mimic nature rather than exploring more efficient solutions.

Units better than neurons

- **Artificial neurons are already abstract**: The concept of artificial neurons is a significant simplification of biological neurons, so why not explore more effective and efficient units?
- **Alternative units can be more effective**: Research has shown that alternative units, such as support vector machines or random forests, can outperform traditional neural networks in certain tasks.
- **Flexibility and innovation**: Moving away from biological analogies can lead to more innovative and effective AI architectures.

ANN's are versatile, powerful, and scalable making them ideal to tackle large and highly complex ML tasks.

Connectionism is **an approach to the study of human cognition that utilizes mathematical models, known as connectionist networks or artificial neural networks**. Often, these come in the form of highly interconnected, neuron-like processing units.

Biological Neurons:

- Consist of:
    - Cell body (with nucleus)
    - Dendrites (branching extensions)
    - Axon (long extension)
    - Telodendria (branches at axon's end)
    - Synapses (small structures at telodendria's tips)
- Function:
    - Produce electrical impulses (action potentials) that travel along axons
    - Release chemical signals (neurotransmitters) at synapses

- Receive neurotransmitters from other neurons, triggering own impulses (or inhibition)

Branching extension called dendrites, a very long extension called the axon , The axom 's length may be a few times or a thousand times of the cell body. Biological neurons produce short electrical impulses called action potentials , which travel along the axons and make the synapses release chemical signals called neurotransmitters. When. a neuron receives a sufficient amount of these neurotransmitters within a few milliseconds, it fires its own electrical impulses.

Each neuron may be connected to thousands of other neurons. Highly complex computations can be performed by a network of simple neurons.

In short, biological neurons transmit and receive signals through electrical and chemical processes, allowing them to communicate with each other.

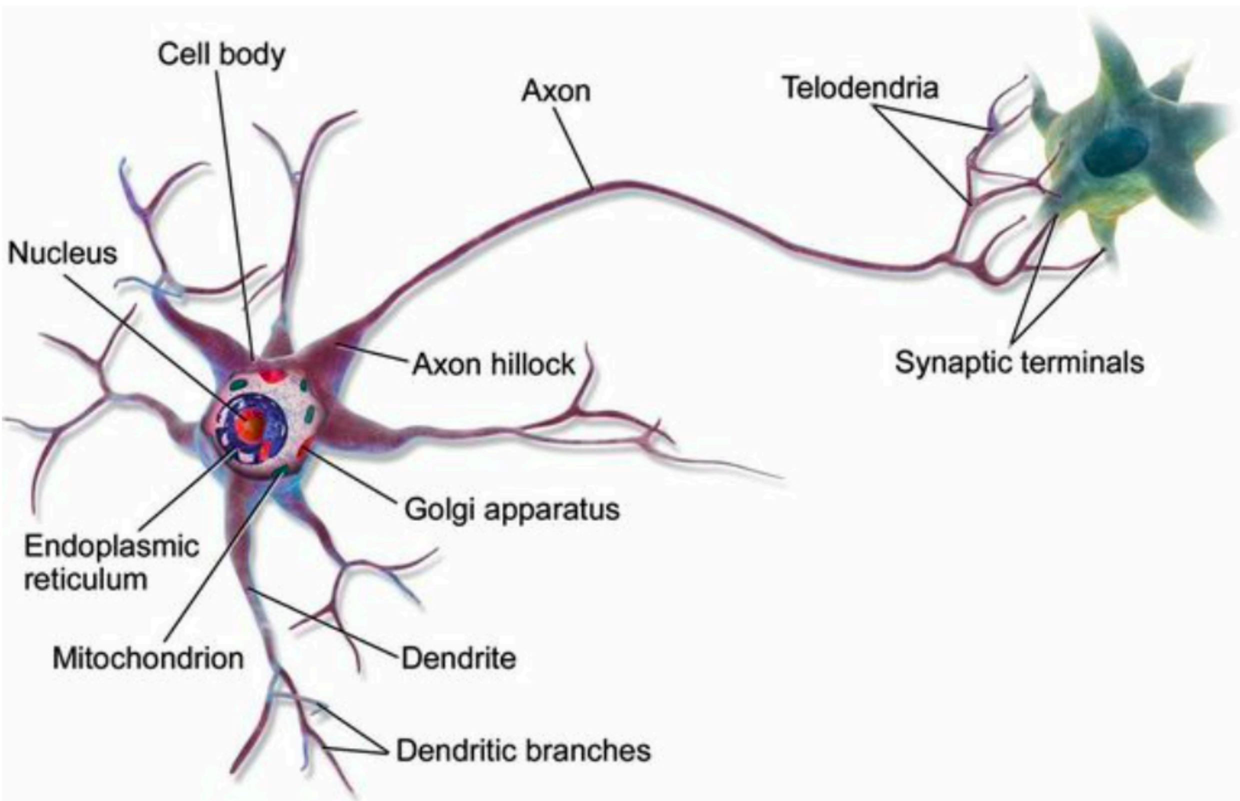
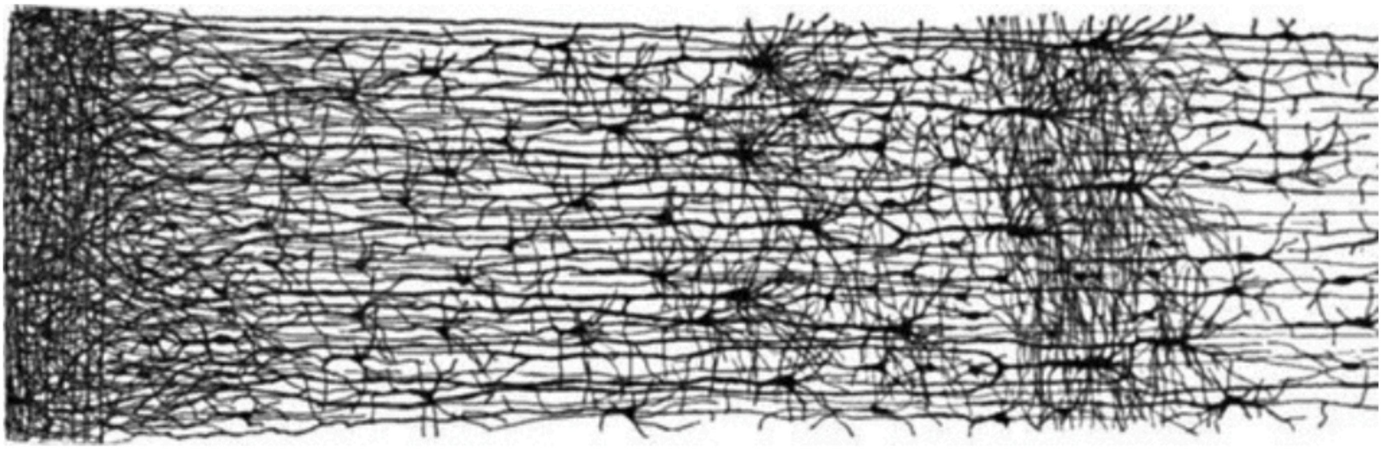


Figure 10-1. A biological neuron [4]

individual biological neurons seem to behave in a simple way, but they're organized in a vast network of billions, with each neuron typically connected to thousands of other neurons. Highly complex computations can be performed by a network of fairly simple neurons.



Multiple layers in a biological neural network (human cortex)

Logical Computation

The McCulloch-Pitts artificial neuron is a simple model of a biological neuron with one or more binary inputs and one binary output.

It activates its output when more than a certain number of its inputs are active. This model works like a switch, where enough active inputs turn the output on, and not enough keep it off.

Despite its simplicity, the McCulloch-Pitts neuron can be connected with others to form networks that can compute any logical proposition, making it a powerful building block for solving problems and making decisions.

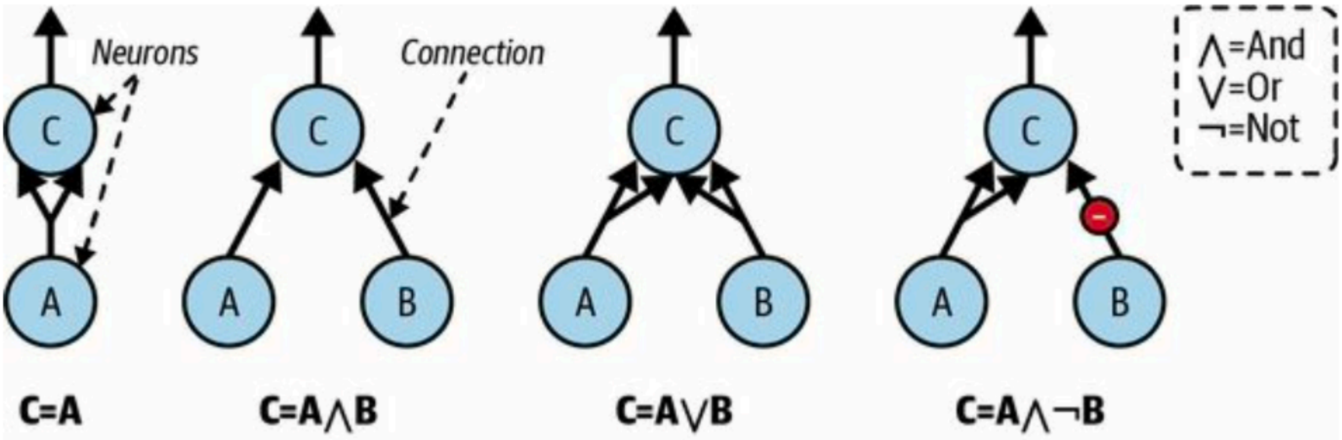
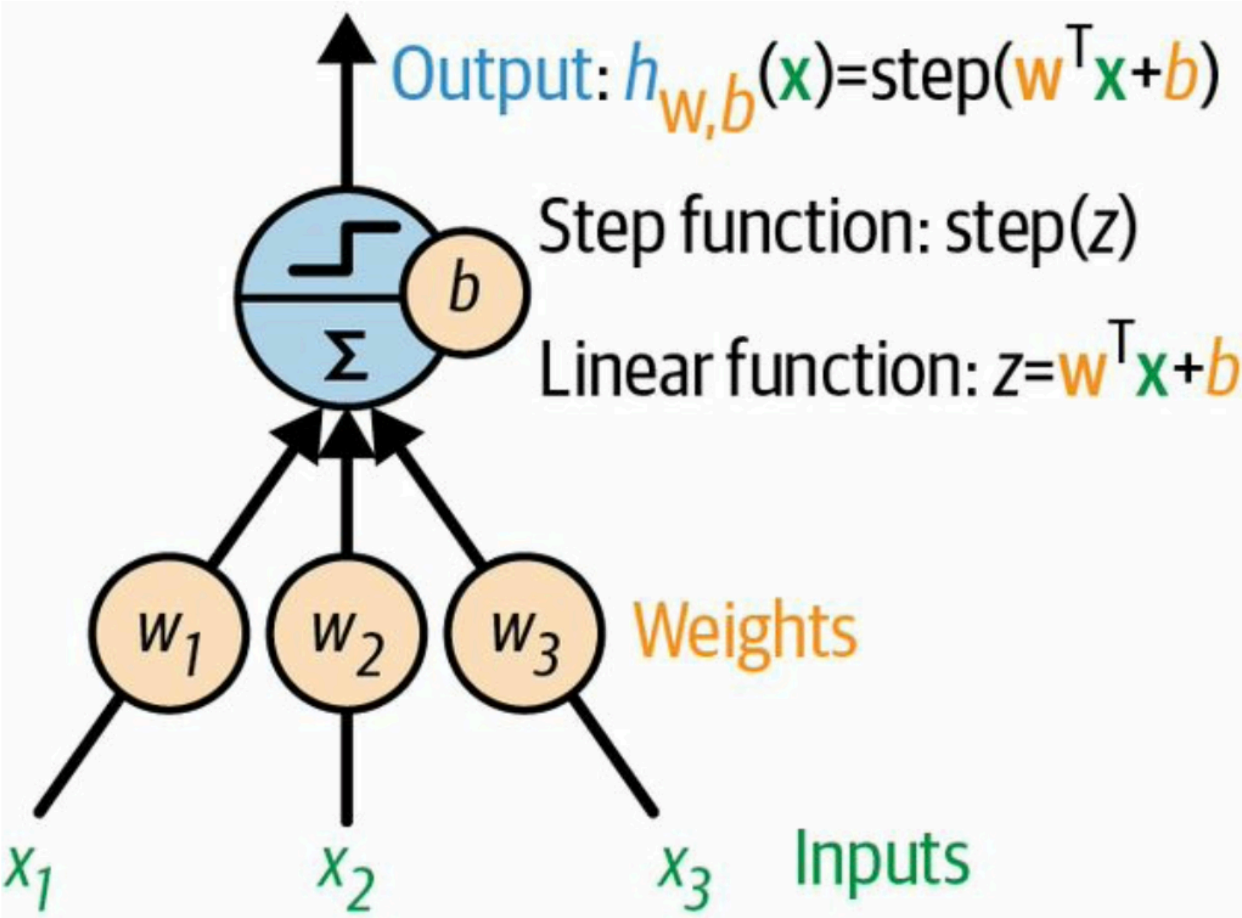


Figure 10-3. ANNs performing simple logical computations

ANN's are considered now because of availability of more training data, computational power, funding , good and better algorithms and the local optima is almost as good as the global optima .

Perceptron

One of the simplest ANN Architecture, It is based on a artificial neuron called a Threshold logic unit (TLC) or a linear threshold unit.
The input and outputs are numbers and each input connection is associated with a weight. The TLU conputes a linear function of its inputs and then applies a step function to the result. (It's like logistic regression but instead of a logistic function it uses a step function).



The most common step function used in perception is the Heaviside step function.

Equation 10-1. Common step functions used in perceptrons (assuming threshold = 0)

$$\text{heaviside}(z) = \begin{cases} 0 & \text{if } z < 0 \\ 1 & \text{if } z \geq 0 \end{cases} \qquad \text{sgn}(z) = \begin{cases} -1 & \text{if } z < 0 \\ 0 & \text{if } z = 0 \\ +1 & \text{if } z > 0 \end{cases}$$

A simple TLU can be used for linear binary classification. It computes a linear function , if the result exceeds a threshold , it outputs the positive class, else the negative class.

A perceptron is composed of one or more TLU's organized in a single layer, where every TLU is connected to every input. Such is called a fully connected layer or a dense layer.

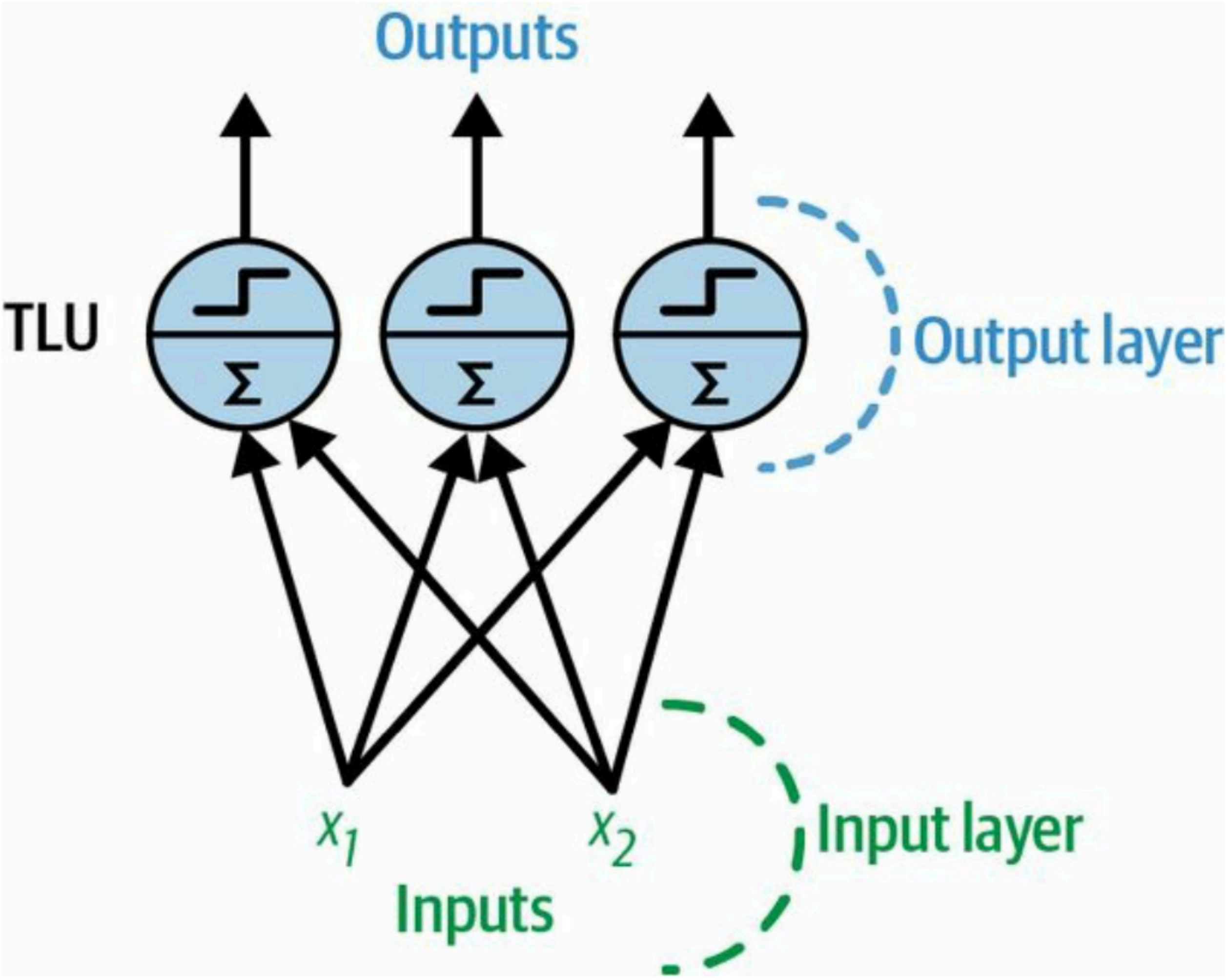


Figure 10-5. Architecture of a perceptron with two inputs and three output neurons

This perceptron can classify instances into 3 different binary classes , which makes it a multilabel classifier.

Equation 10-2. Computing the outputs of a fully connected layer

$$h_{W,b}(X) = \phi(XW + b)$$

X is the matrix of input feature, row per instance and col per feature.
W is the weight matrix containing all the connection weight. row per input and col per neuron
b is the bias term vector
ø is activation function where neurons are TLU .

Training of a Perceptron

When a biological neuron triggers another neuron often, the connection between these two neurons grows stronger. Cells that wire together fire together.
Perceptrons are trained using a variant of this rule that takes into account the error made by the network when it comes to prediction.
Perceptron learning rule reinforces connections that help reduce the error. For each instance it makes a wrong call , it reinforces the weights from that input that would have countrified to the correct prediction.

The decision boundary is linear so perceptron is incapable of learning complex pattern. However if the training instances are linearly separable the algorithm will converge. This is the perceptron convergence theorem.

Some limitations of perceptrons can be eliminated by stacking multiple perceptrons . The resulting ANN is called a multilayer perceptron. (MLP). An MLP can solve XOR . Perceptrons do not output a class probability unlike logistic regression classifiers. They also don't use regularization by default, but they may train a bit faster.

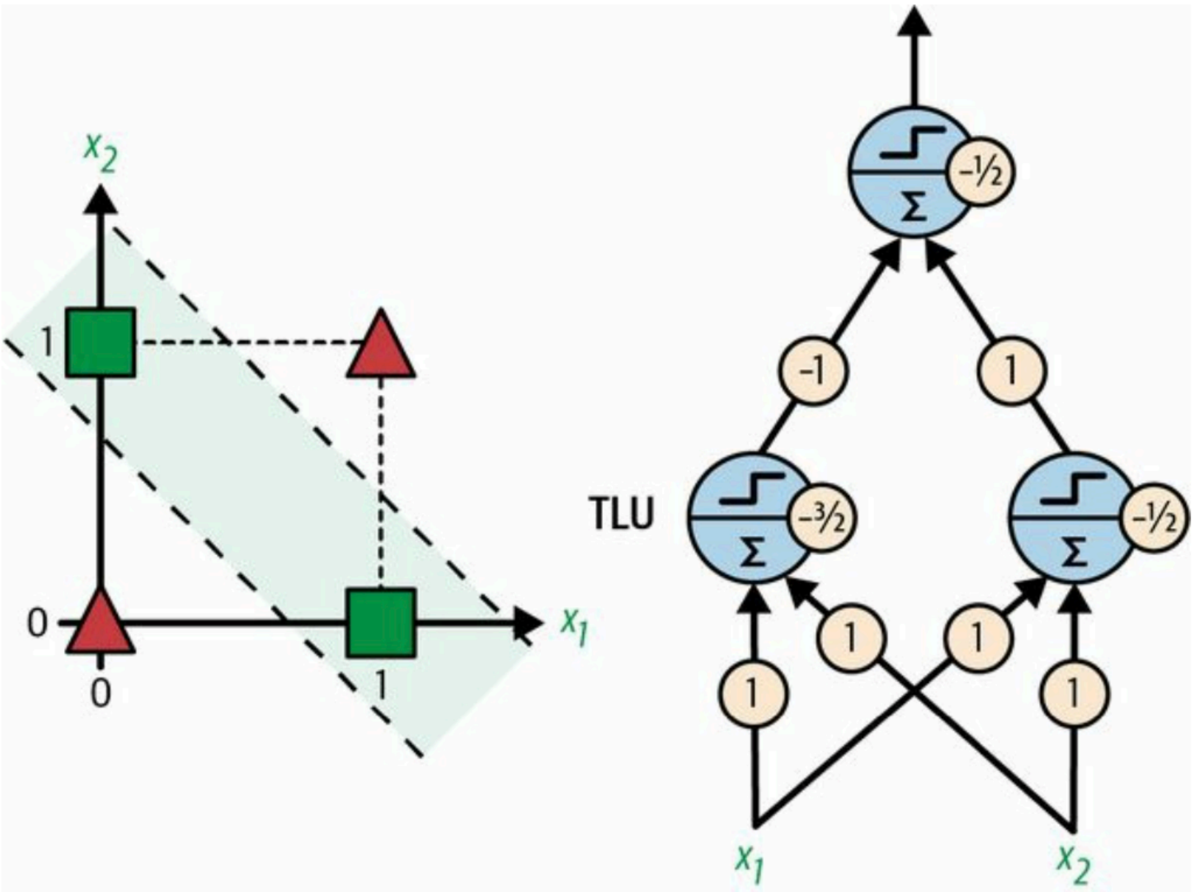


Figure 10-6. XOR classification problem and an MLP that solves it