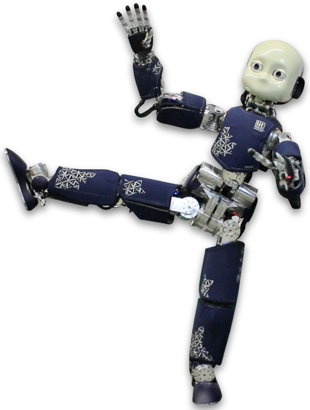
Introduction to Multi-Layer Perceptrons (MLP)

Applied Machine & Deep Learning (190.015)

Univ.-Prof. Dr. Elmar Rueckert

Telefon: +43 3842 402 - 1901
Email: teaching@ai-lab.science



WO AUS FORSCHUNG ZUKUNFT WIRD

Chair of Cyber-Physical-Systems

1st Week:

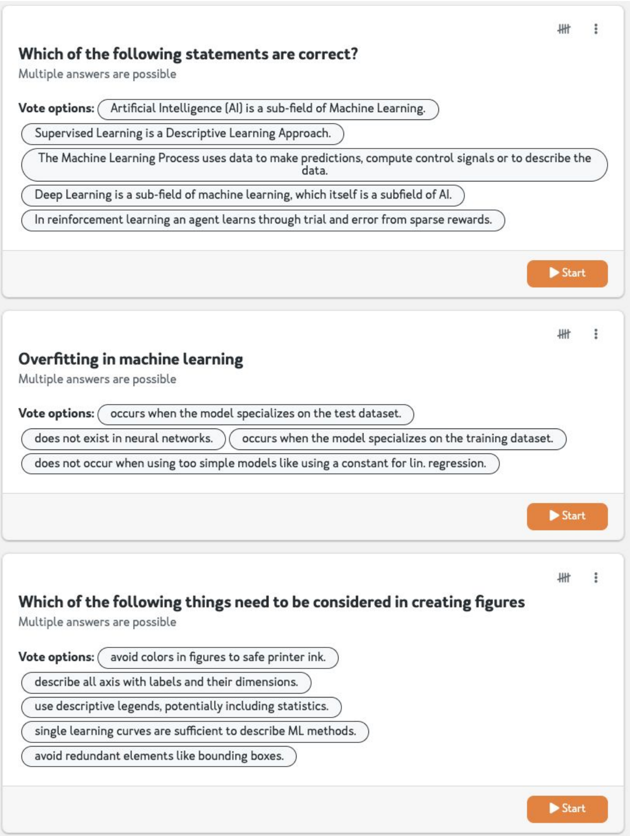
	MON 02.10.2023	TUE 03.10.2023	WED 04.10.2023	THUR 05.10.2023	FRI 06.10.2023
Topic	Intro to ML Organisation	Neural Networks	Representation Learning	Robot Learning	AML Projects
9 am					
:15					
:30					
:45					
10 am					
:15	Quizz on ML	Quizz on Neural Nets	Introduction to Deep Representation Learning		Quizz on AML
:30	Introduction to ML	Introduction to Multi-Layer-Perceptrons			Project Topic Presentations
:45					
11 am	15 min Break	15 min Break	JupyterHub NB on Rep. Learning		
:15	Statistics, Model Validation, Figures & Evaluations	Handout on Neural Networks using playground.tensorflow			Team Ass., Git Repos & Wiki Instructions
:30			30 min Break		
:45					AML Summary
12 pm	30 min Lunch Break	30 min Lunch Break	Curiosity (MLPs), Imagination (Dreamer) and Information (Empowerment)		
:15				Quizz on Robotics	
:30	Course Organisation & Grading	Introduction to CNNs		Introduction to Robot Learning	
:45					
1 pm	15 min Break	15 min Break	Quizz Summary		
:15	Python Programming with our JupyterHub	JupyterHub NB on MLPs CNNs		15 min Break	
:30				Handout on Robot Learning (Model Learning & RL)	
:45	Quizz Summary	Quizz Summary			
2 pm					
:15				15 min Break	
:30				Introduction to Mobile Robotics & SLAM	
:45					
3 pm				JupyterHub NB on Path Planning	
:15					
:30				Quizz Summary	
:45					

Legend

Quizz on ML	Online Quizz using https://tweedback.de
Course Content Presentation	Using google slides, etc.
15 min Break	Breaks to recover or to continue programming
Organisation & Instructions	Using google slides, etc.
Practical Exercise	Using online tools, our JupyterHub, etc.
Latest Research	State-of-the-art research

Quiz on Machine Learning

Chance to get **three bonus points:** https://tweedback.de/zxyk

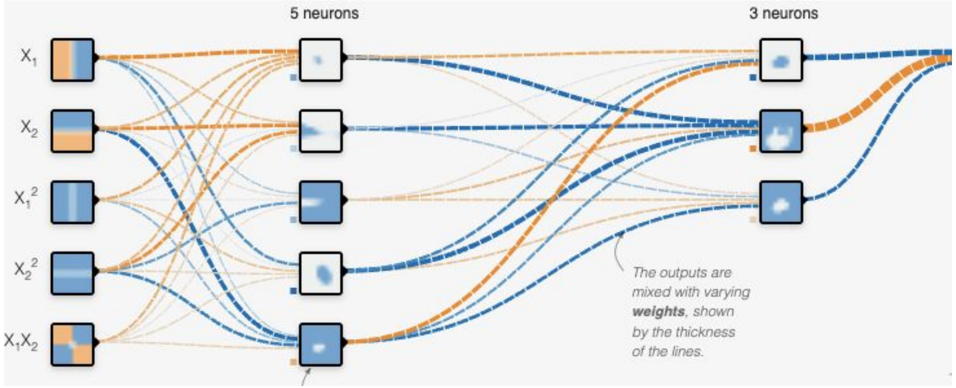


Outlook of this lecture

- Definition of Artificial Neural Networks
- The Human Brain
- The Neuron Model and Artificial Networks
- Selected Research Projects with Neural Networks
- Future Topics and how to study neural nets

Definition of Artificial Neural Networks

An **artificial neural network (ANN)** is a computational model inspired by the structure and function of the human brain, composed of interconnected nodes (neurons) that process and transmit information to perform tasks such as pattern recognition, data analysis, and decision-making.



Source: https://playground.tensorflow.org, 18.09.2023

[Interactive 3D Model ➪]

Facts:

- approx. 1400 grams

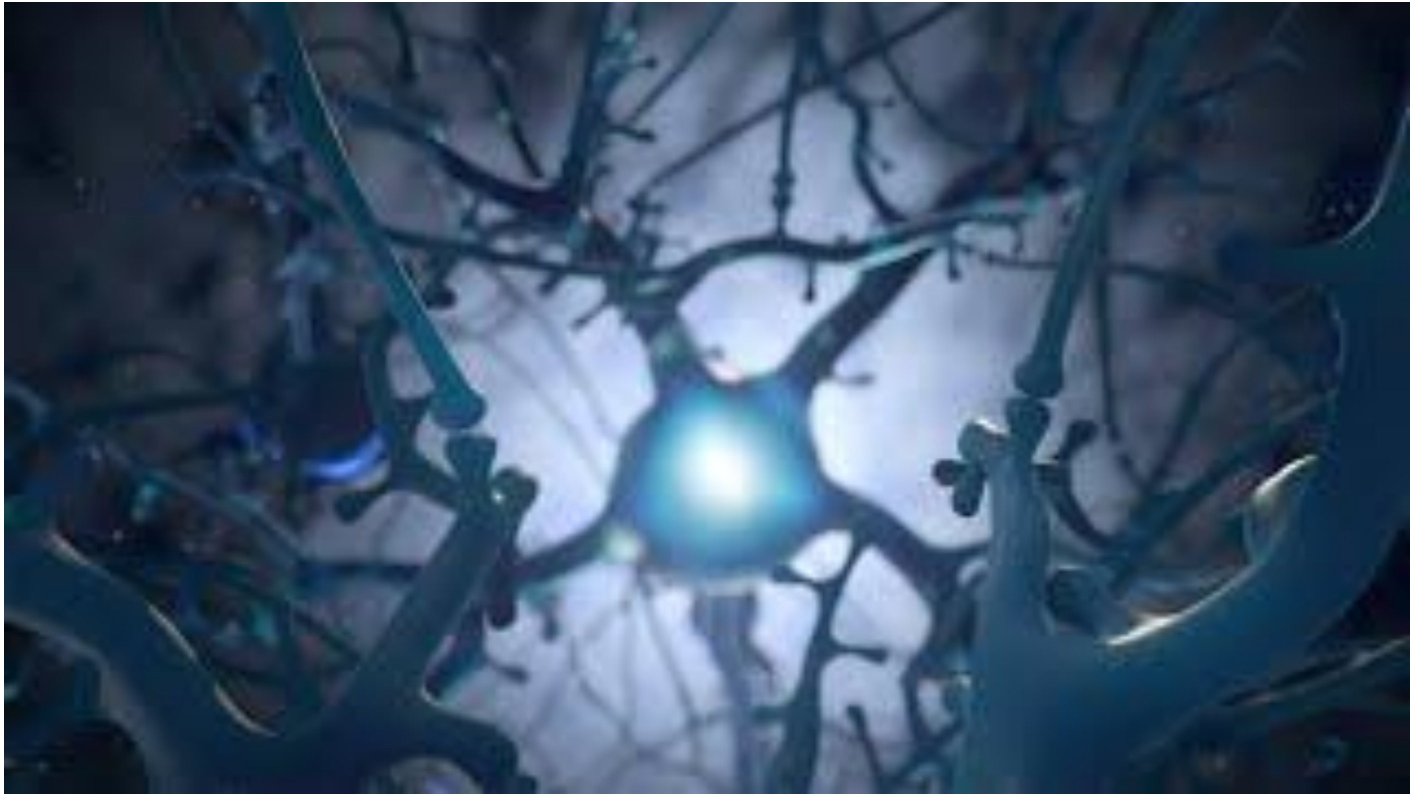
- 80 - 120 10^9 neurons
- 100 times more glial cells (support cells)

- A neuron is connected to up to 30.000 others
- ca. 10^{14} Synapses with a total length of 6 10^6 km (15.6x to the moon)

- 20% of all neurons are in the 2-4mm thick cerebellum (Großhirnrinde)

- Many different types of neurons with special purpose functions like pyramid cells, Purkinje cells, etc…

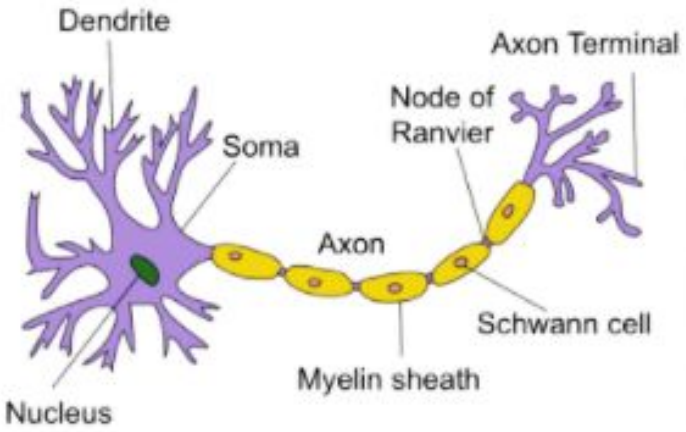
Modelling the most powerful universal AI system



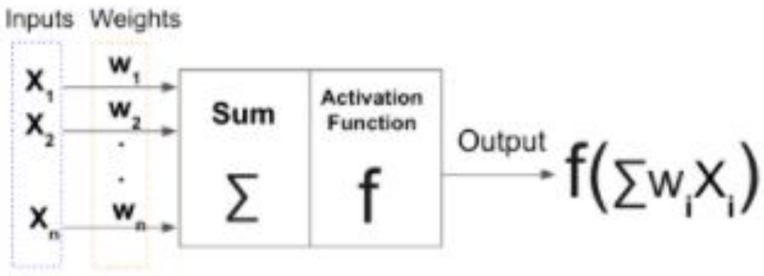
Source:
https://www.youtube.com/watch?v=hv7wYRndWpo visited last 03.10.2023.

Modelling the most powerful universal AI system

This neuron model is a strong abstraction!



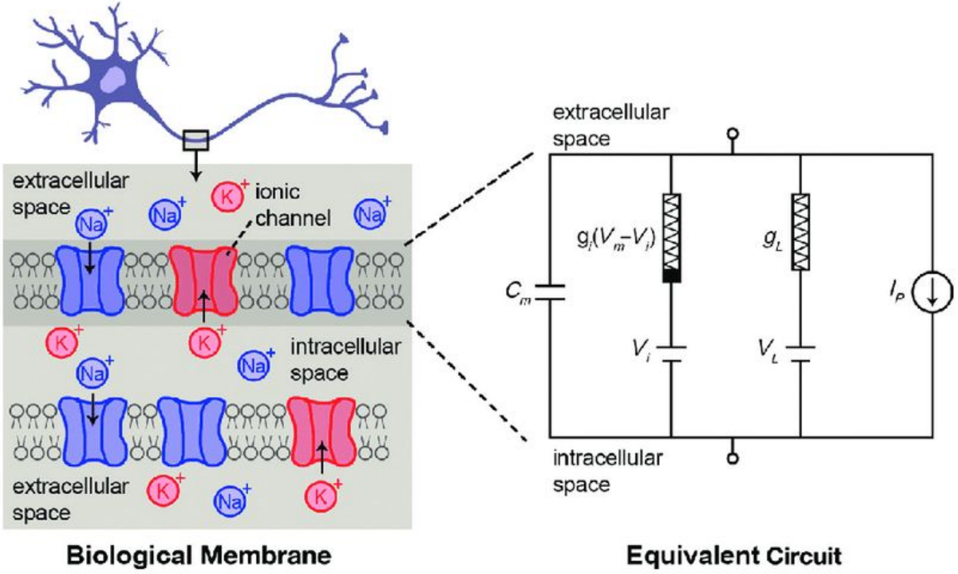
Structure of a typical neuron
(source: Wikipedia)



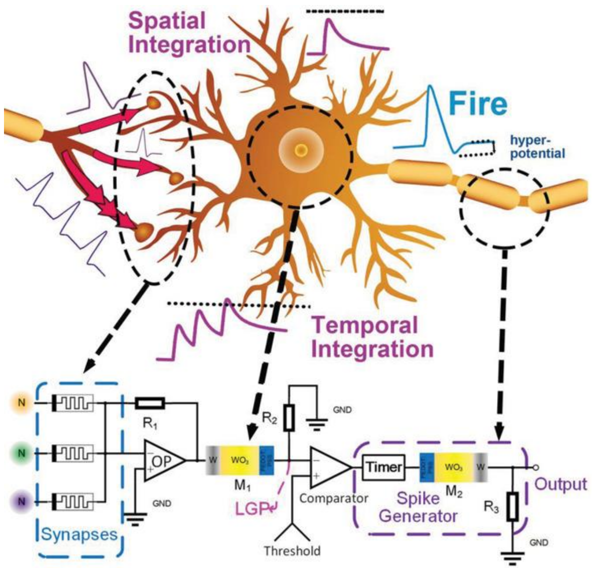
Structure of artificial neuron

Source: https://news.sophos.com/en-us/2017/09/21/man-vs-machine-comparing-artificial-and-biological-neural-networks/ and Wikipedia, visited last 18.09.2023.

The **Hodgkin-Huxley Model** is a more realistic approximation that models the biophysical characteristic of cell membranes.

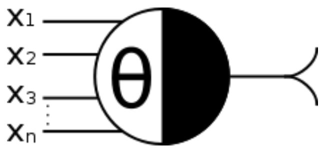


Van De Burgt, Y., & Gkoupidenis, P. (2020). Organic materials and devices for brain-inspired computing: From artificial implementation to biophysical realism. *MRS Bulletin*, *45*(8), 631-640.

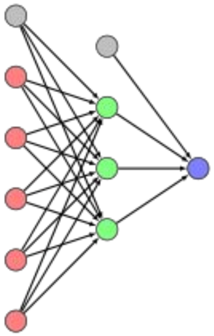


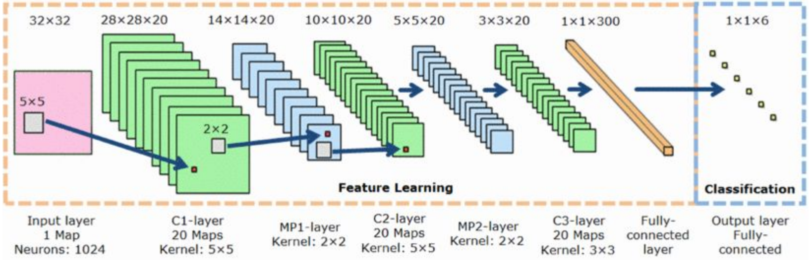
Huang, H. M., Yang, R., Tan, Z. H., He, H. K., Zhou, W., Xiong, J., & Guo, X. (2019). Quasi-Hodgkin–Huxley Neurons with Leaky Integrate-and-Fire Functions Physically Realized with Memristive Devices. *Advanced Materials*, *31*(3), 1803849.

Short History



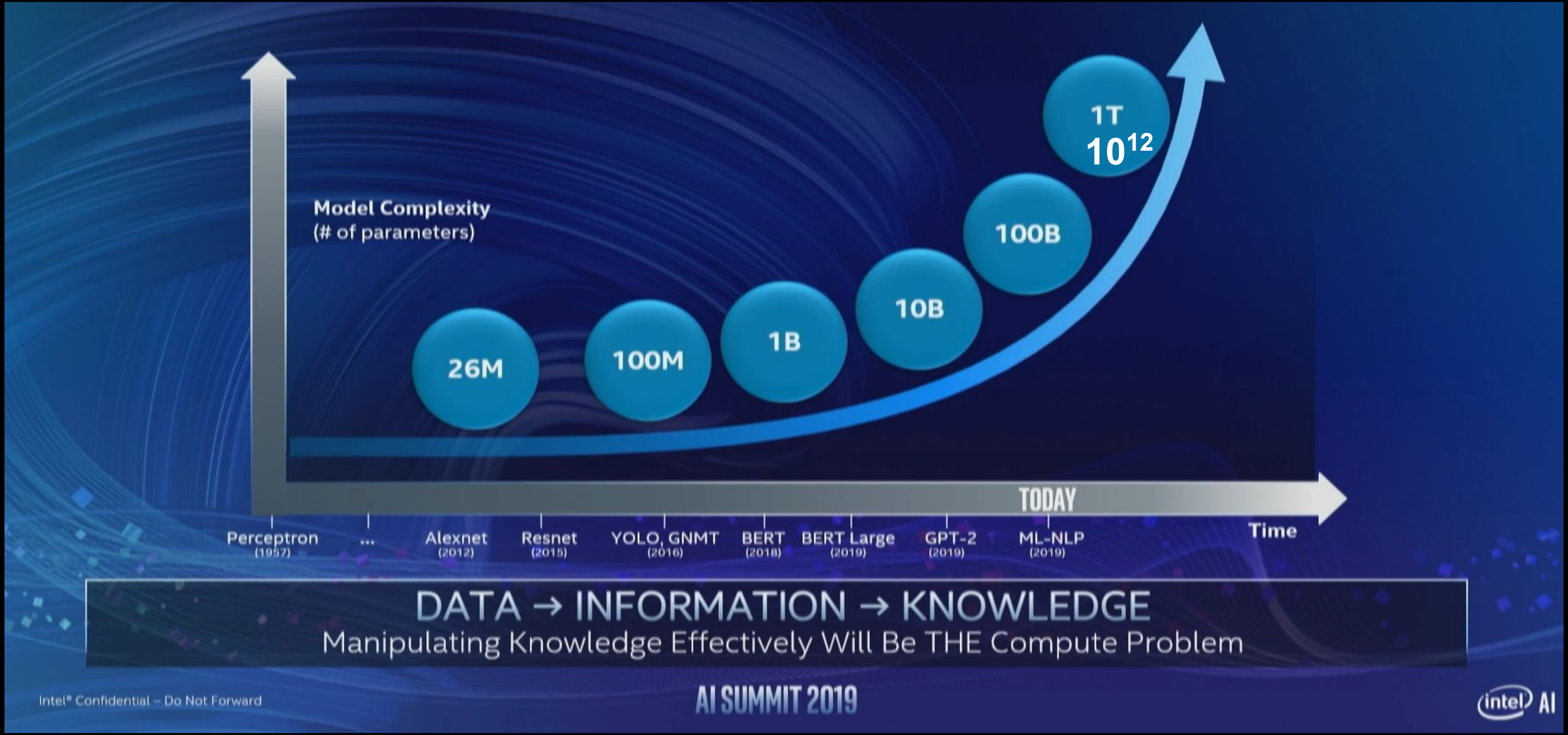
- 1943 Warren **McCulloch**, Walter **Pitts:** Binary cell model for information processing.
- 1949 Donald **Hebb**: „what fires together, wires together“.
- 1962 **Rosenblatt**: proof of Convergence in binary classification.
- 1969 **Minsky** & **Papert**: perceptrons can not learn non-linearly separately target functions.
- 1970s **Werbos, Rumelhart, McClelland, Hinton:** multi-layer perceptrons can represent any nonlinear target function, but there is no guarantee of convergence to the minimum.
- 1989 **Cybenko:** MLP can rep. any nonlinear target function **with only one hidden layer**
- 2000s, 2010s **Deep Learning:** Using large nets with many layers, sophisticated update rules and regularization strategies.





Todays Nets approach the complexity of our human brain but can only solve a single task (NLP in LLMs)

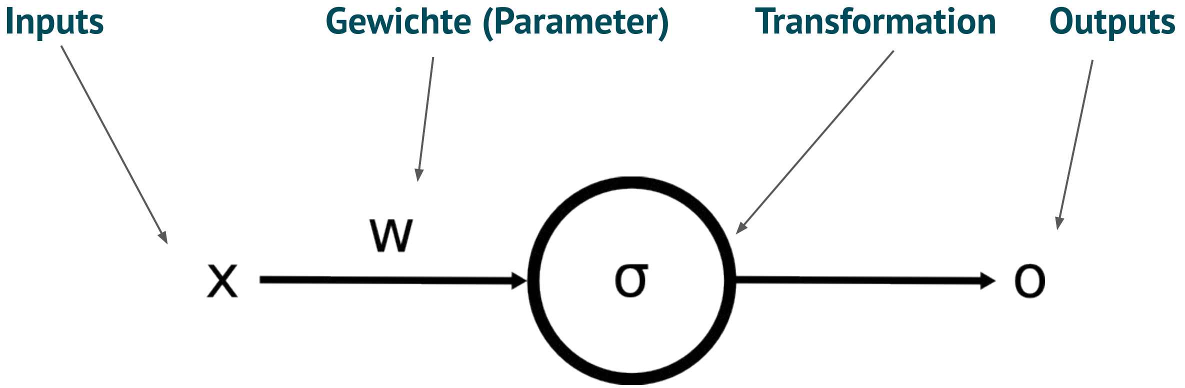
Human brain >>
10^{14} Synapse + chemical processes



Source: https://www.datanami.com/2019/11/13/deep-learning-has-hit-a-wall-intels-rao-says/, last visited 18.09.2023.

A simple Neuron Model: The Perceptron

$$o = \sigma(x\ w)$$

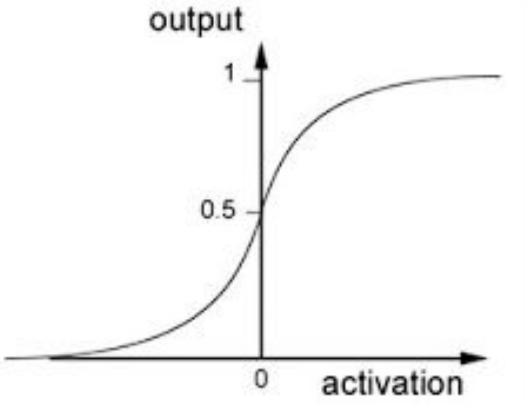


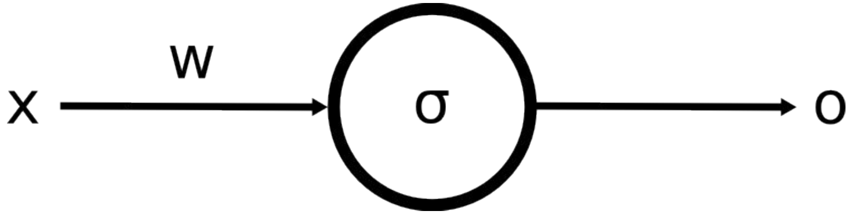
The ‘transformation’ of ‘activation’ function introduces a nonlinearity!

$$o = \sigma(x\ w)$$

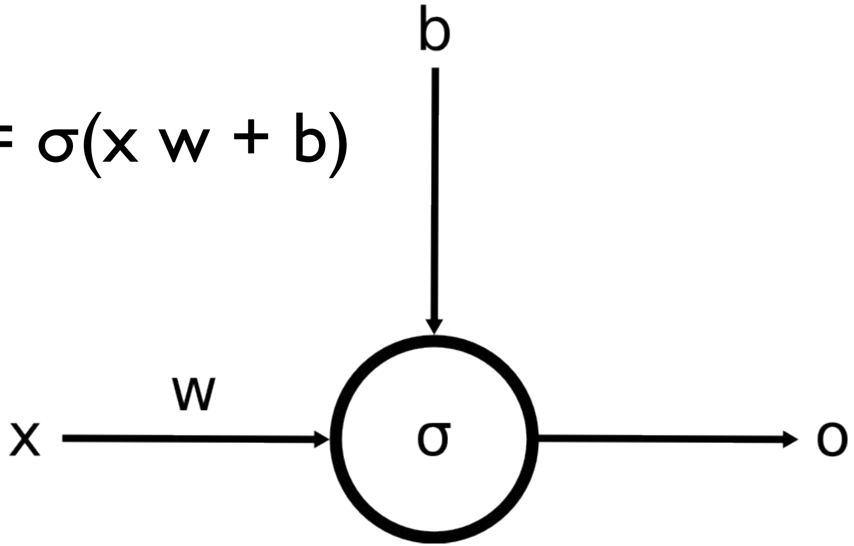
$$\sigma(x) = 1/(1+\exp^{-x})$$

$$\boldsymbol{\sigma'(x) = \sigma(x)(1-\sigma(x))}$$



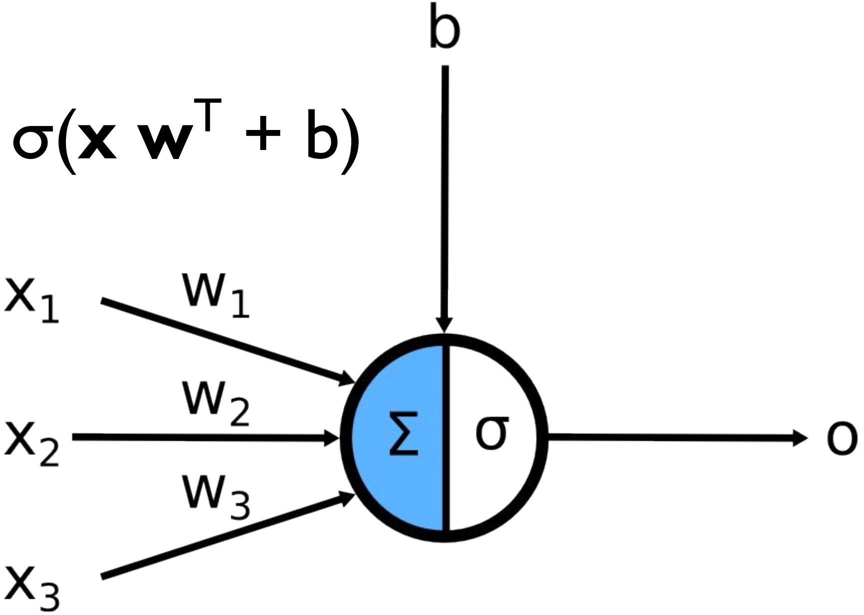


$$o = \sigma(x\,w + b)$$



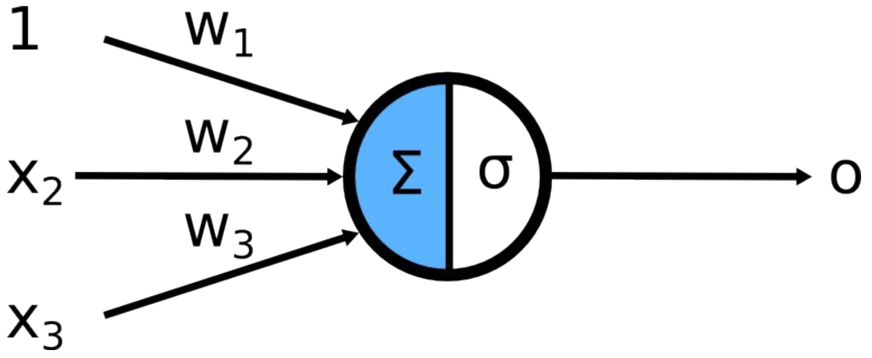
We use a compact vector notation

$$o = \sigma(\mathbf{x}\,\mathbf{w}^T + b)$$

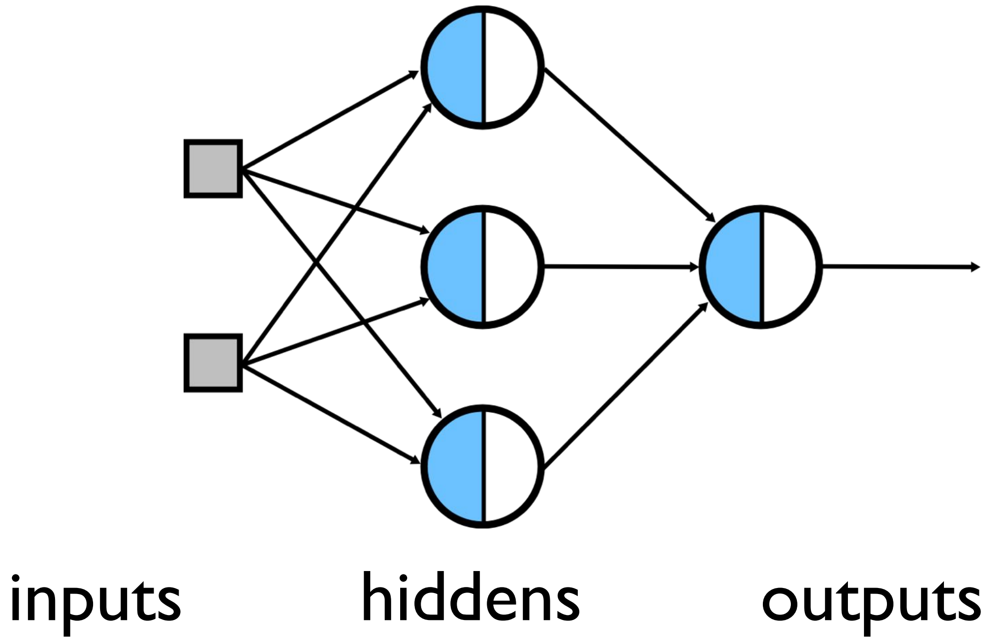


Augmented Inputs with a preceding 1 to get rid of the bias parameter!

$o = \sigma(\mathbf{x}\,\mathbf{w}^T)$ $\qquad$ $\mathbf{x} = [1, x_1, x_2, \ldots]$

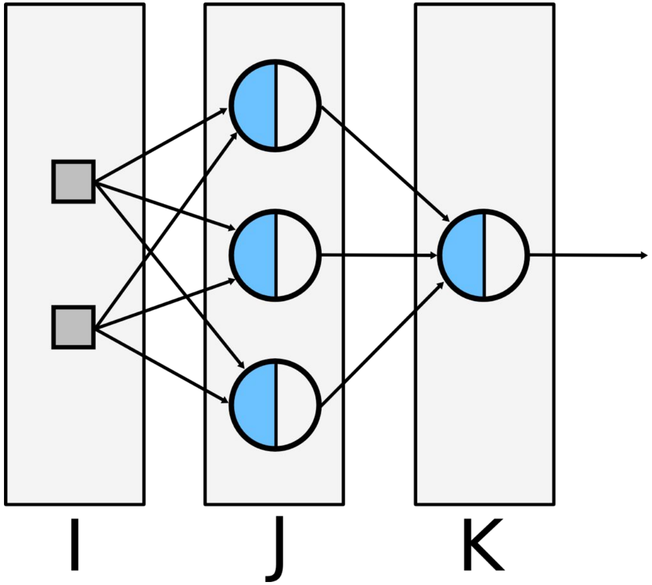


Extension to a Multi-Layer Perceptron Network

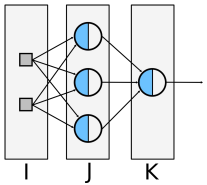


The Learning Algorithm

I, J and K denote the number of neurons in the respective layers in the three layer network.



The Learning Algorithm



- The goal is to minimize an objective (or cost function):

$$E = \frac{1}{2}\Sigma_{k \in K}(o_k - t_k)^2$$

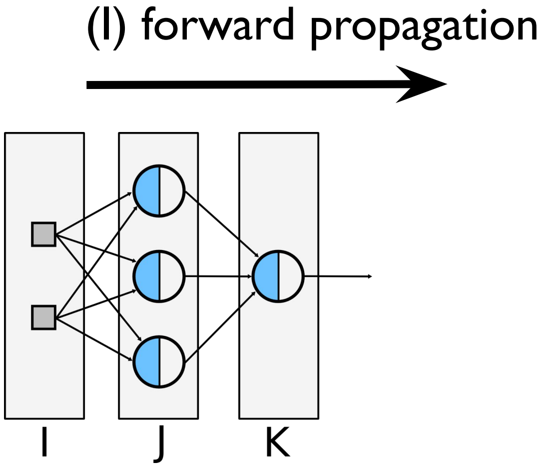
- via iterative updates (converges to a local minima):

$$\frac{\partial E}{\partial w^l_{jk}}$$

- The trick is to do that efficiently for all layer.

The Learning Algorithm

It is a two step process using a **forward information pass** and a **backward error propagation step**.

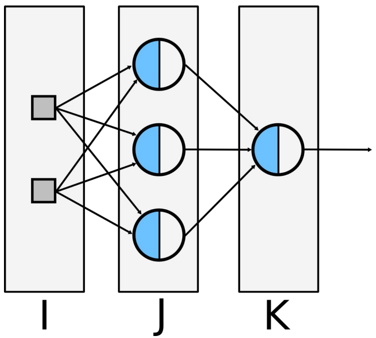


$$E = \frac{1}{2}\Sigma_{k \in K}(o_k - t_k)^2$$

The Learning Algorithm

It is a two step process using a **forward information pass** and a **backward error propagation step**.

(I) forward propagation



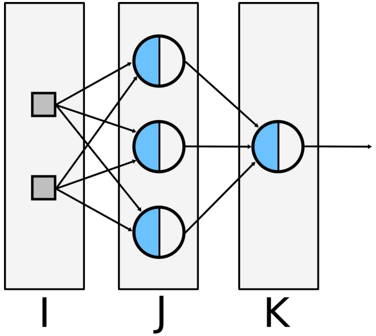
$$E = \frac{1}{2} \Sigma_{k \in K} (o_k - t_k)^2$$

$$\frac{\partial E}{\partial w^l_{jk}}$$

(II) error back-propagation

The Learning Algorithm

The updates in output and all hidden layers are treated separately!



1. Compute the outputs of all neurons in all layer.

   $o = \sigma(x\ w)$

2. For each output neuron, compute the error and the gradient.

   $$\delta_k = (o_k - t_k)\,\sigma(x_k)\,(1 - \sigma(x_k))$$

3. For each neuron in the hidden layer(s) compute the weight update to the output layer.

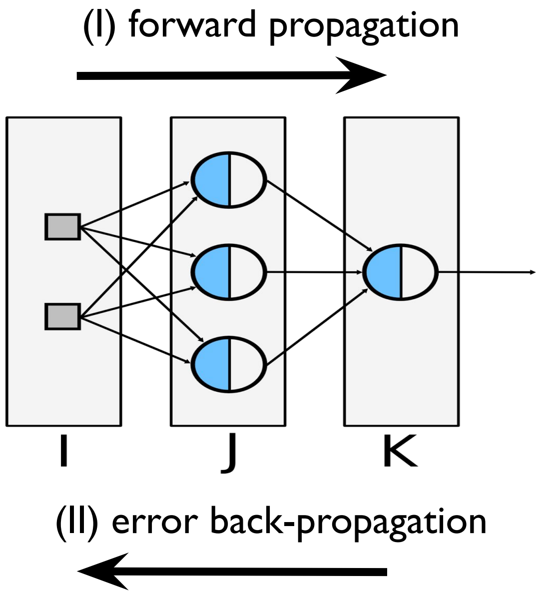
   $$w_{jk}(t+1) = w_{jk}(t) - \eta\,\delta_k\,o_j$$

4. For each neuron in the hidden layer(s) compute the weight update to the previous layer till the input layer is reached.

   $$\delta_j = \Sigma_{k \in K} \delta_k w_{jk}\,(1 - o_j)\,o_j$$

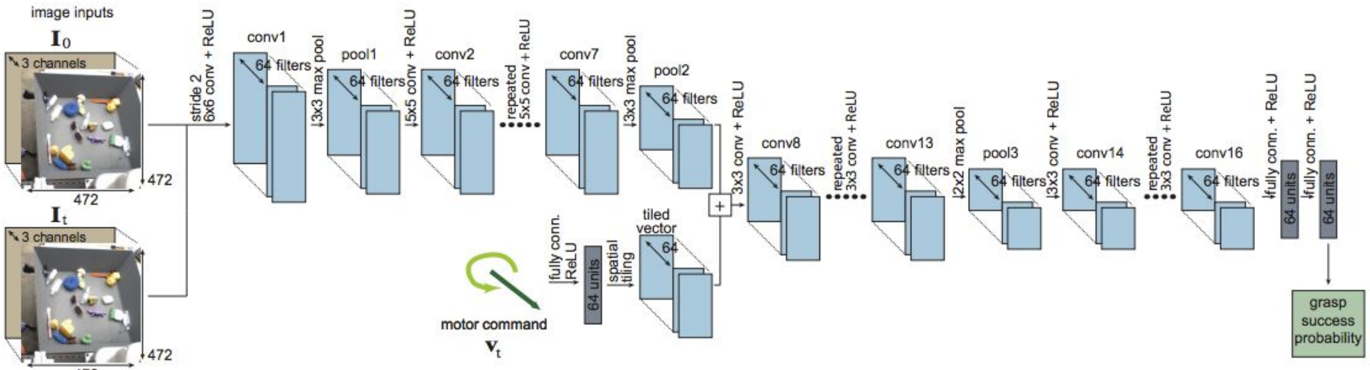
   $$w_{ij}(t+1) = w_{ij}(t) - \eta\,\delta_j\,o_i$$

Extensions



- Transformations: lineare, tanh, Gaussian, etc.
- Stochastic Gradient Descent.
- Minibatch updates.
- Regularizations (L1, L2, Dropout).
- Early stopping (when the test error starts to increase).
- and many more…

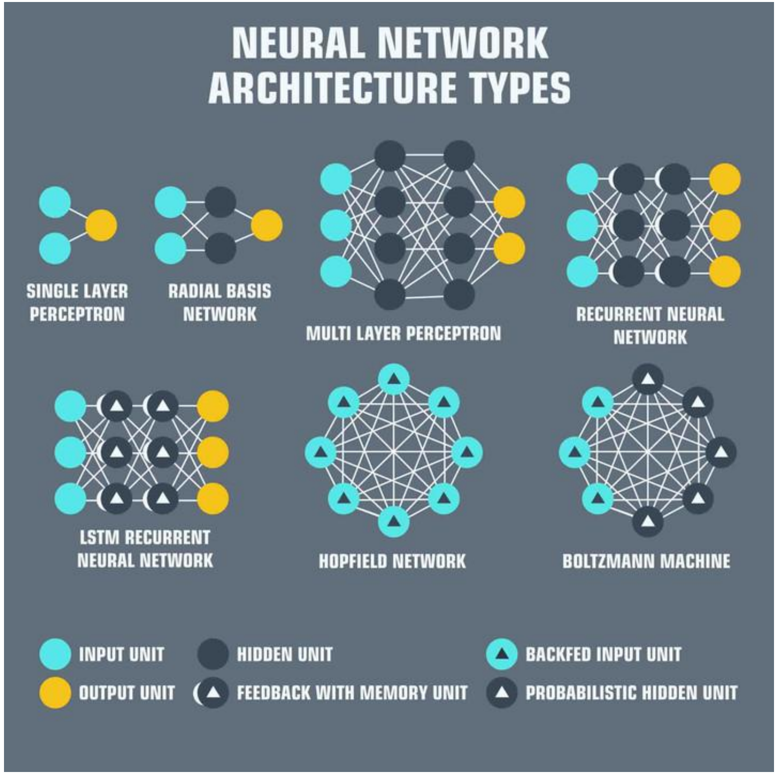
MLPs are special variants of „Deep Networks“

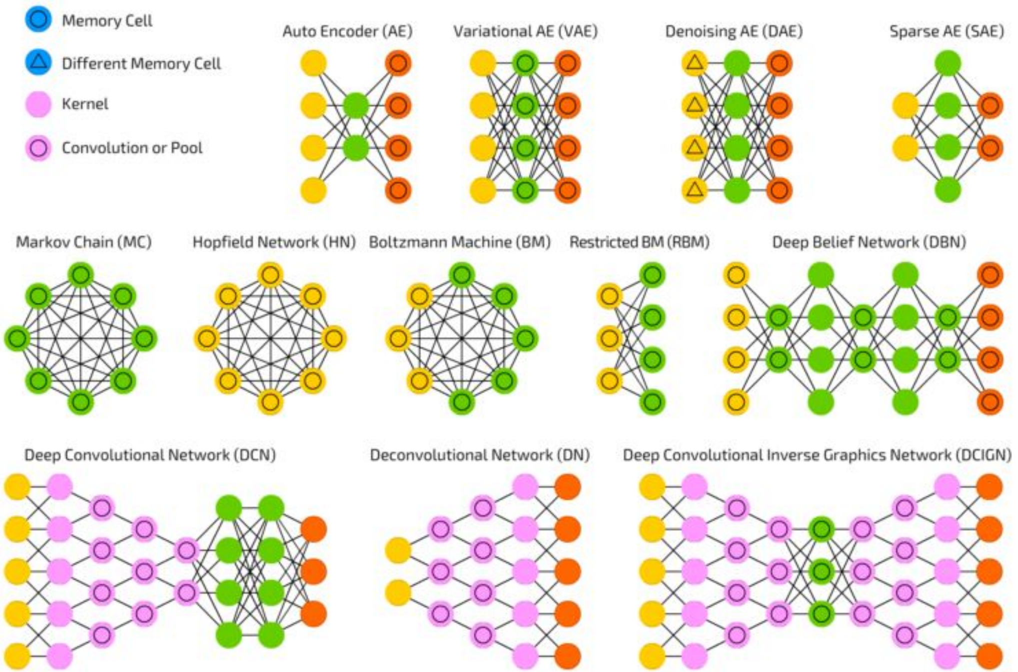


- pre-training (autoencoder) to avoid the vanishing gradient problem

- pruning to avoid overfitting

Many different Types of Neural Networks

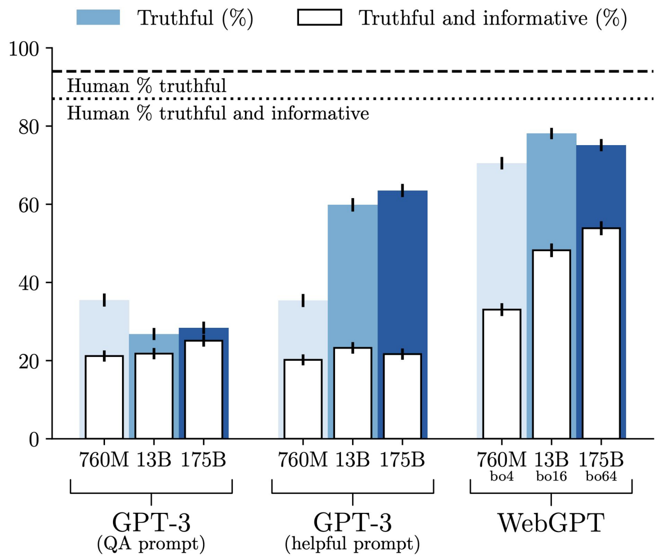




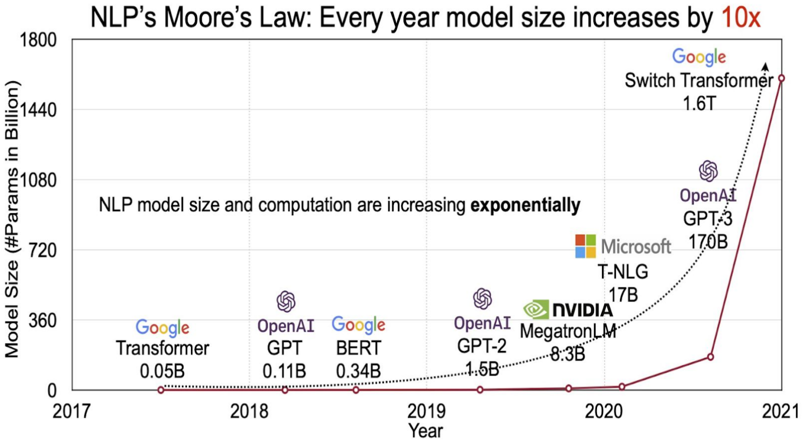
Fjodor van Veen from Asimov institute compiled a wonderful cheatsheet on NN topologies.

From https://www.linkedin.com/pulse/3-types-neural-networks-ai-uses-naveen-joshi/, visited on 03.10.2023.

The future of Large Language Models

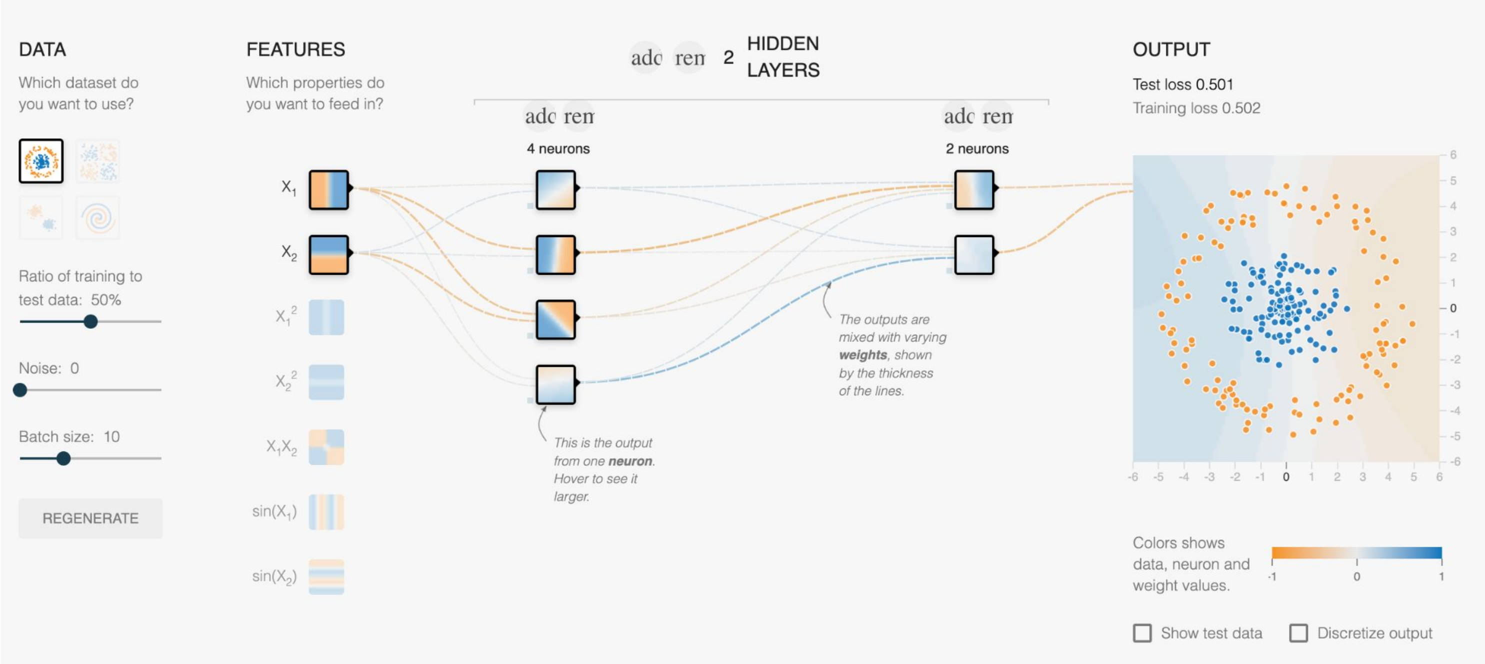


Source: OpenAI Pre-Print: https://arxiv.org/pdf/2112.09332.pdf



Source:
https://indiaai.gov.in/article/the-future-of-large-language-models-llms-strategy-opportunities-and-challenges, last visited, 18.09.2023.

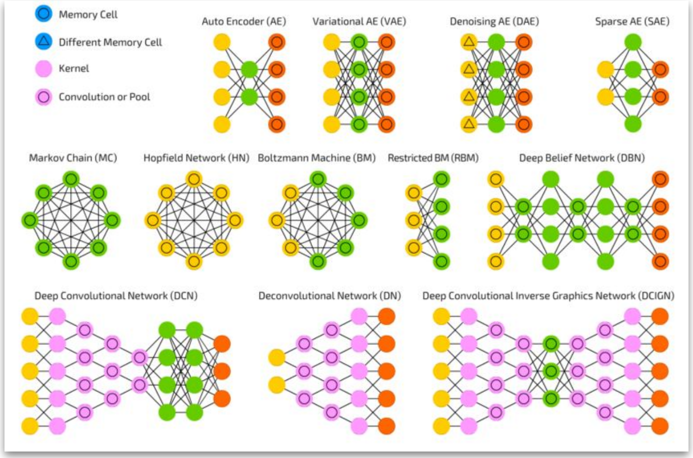
Let’s try it yourself!



Source: https://playground.tensorflow.org, 18.09.2023

Summary of Multi-Layer Perceptrons

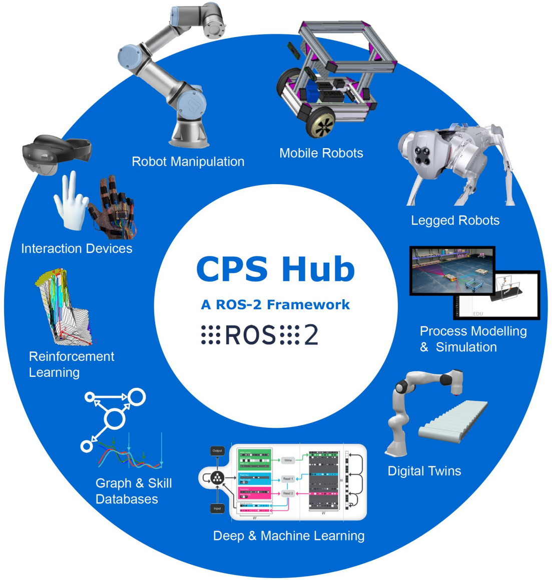
- ANNs are inspired by neuron, synapses and their interplay in the human brain.
- Perceptrons are strong abstractions of biological neurons.
- Multi-Layer Perceptrons can approximate any nonlinear function with just one hidden layer (1989 Cybenko).
- The Learning algorithm is based on a forward pass and a error backpropagation process.
- Many different network architectures exist.



Thank you for your attention!

Visit our Youtube Channel:
https://youtube.com/@CPSAustria

YouTube 117



Phone: +43 3842 402 – **1901** (Sekretariat CPS)

Email: cps@unileoben.ac.at

Web: https://cps.unileoben.ac.at