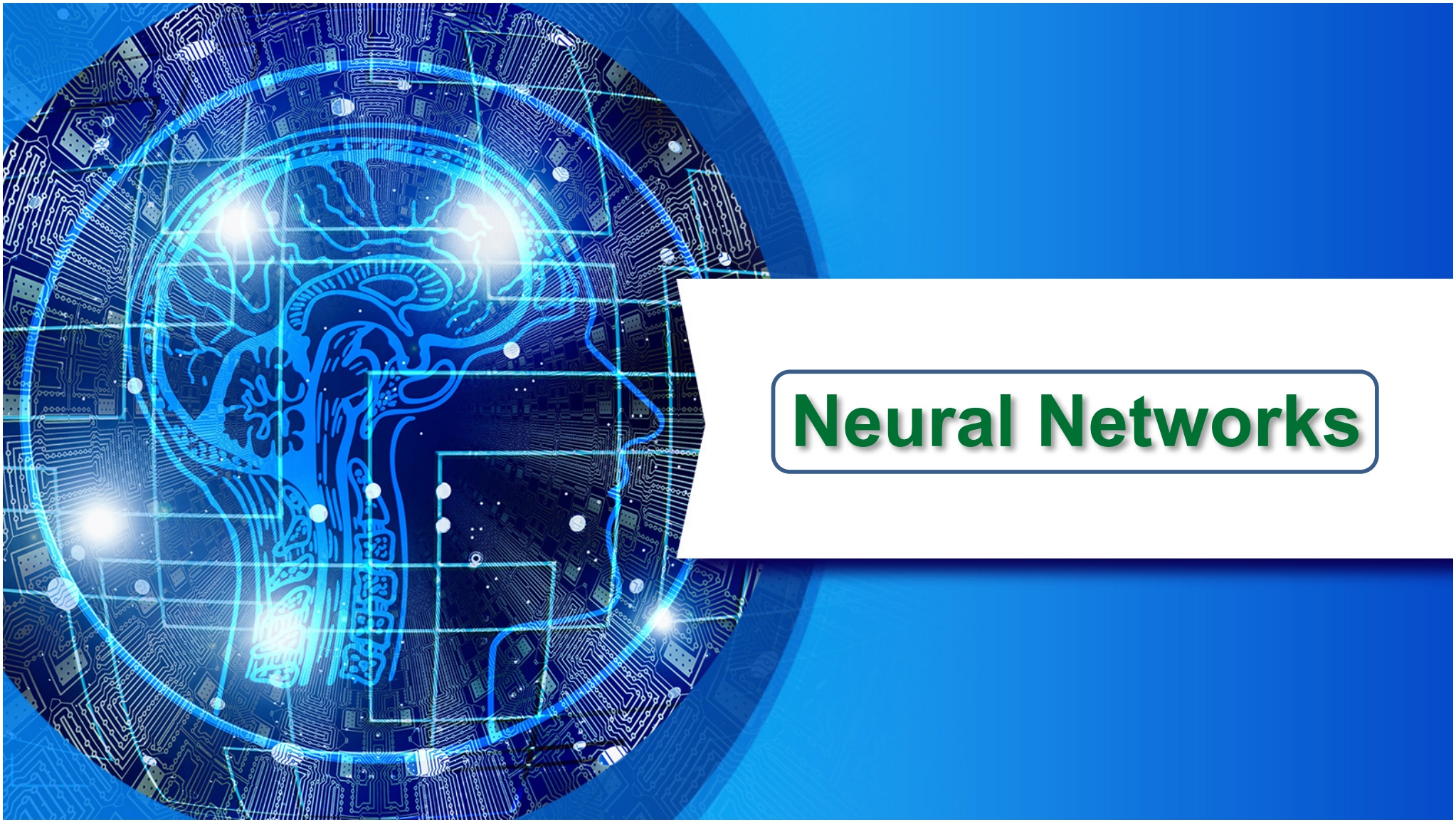
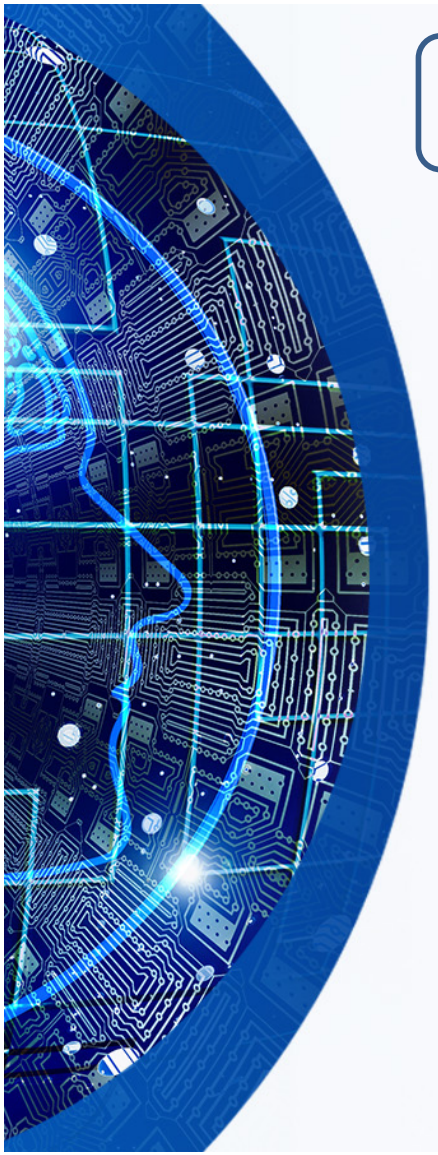


Neural Networks

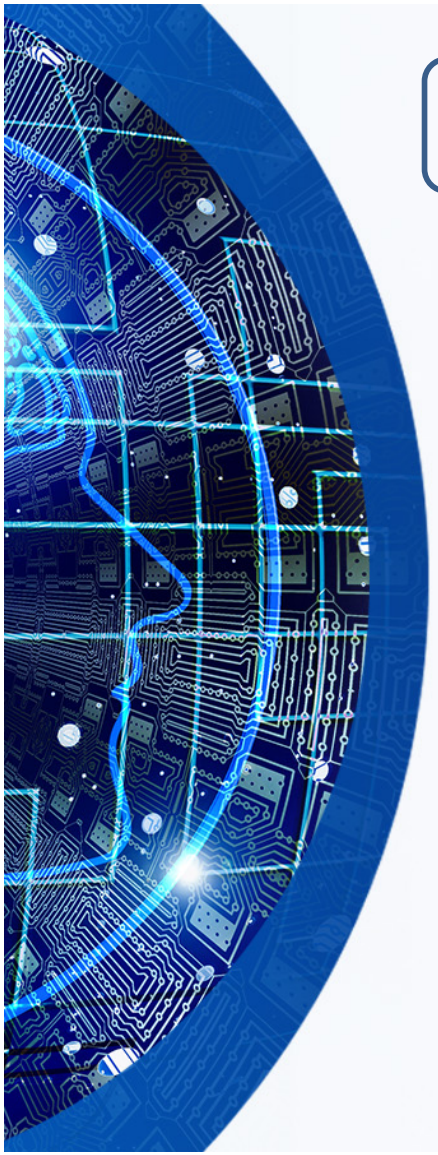
Neural Networks

- What are Neural Networks
- Biological Neural Networks
- ANN - Basic



Neural Network (NN)

- Neural Networks were inspired by the neural architecture of a human brain.
- Massively parallel, distributed system, made up of simple processing units (neurons).
- It provided an output by applying the function on the inputs provided.
- NN composed of neurons, contains activation functions that makes it possible to predict non-linear outputs

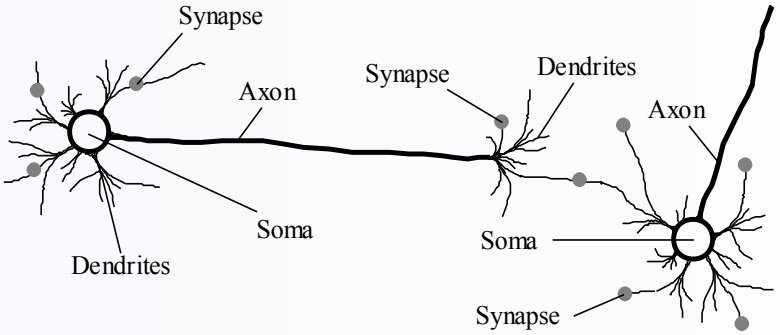


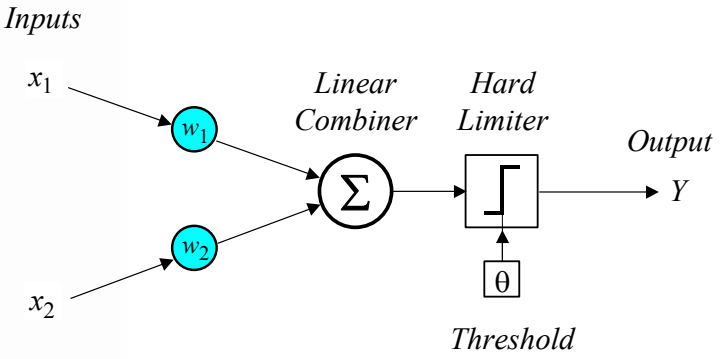
History

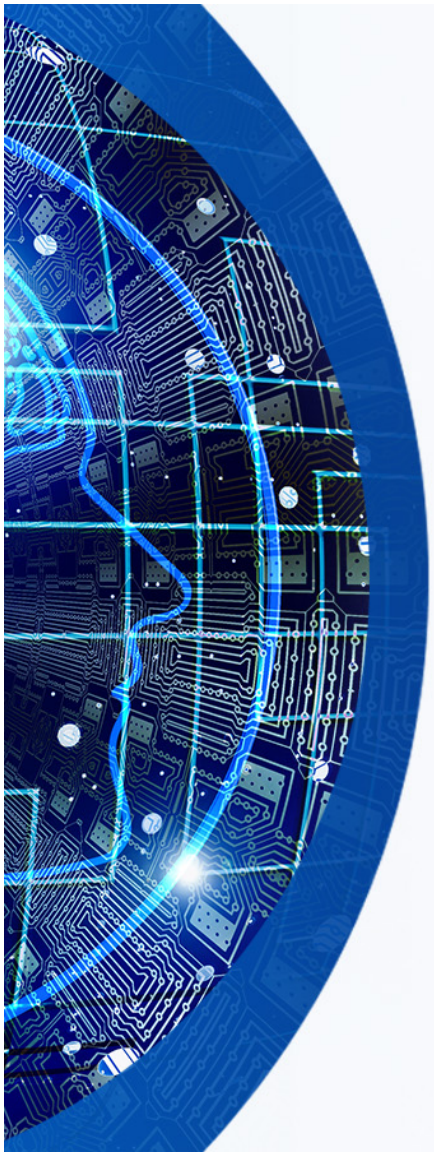
- late-1800's - Neural Networks appear as an analogy to biological systems
- 1960's and 70's – Simple neural networks appear
  - Fall out of favor because the perceptron is not effective by itself, and there were no good algorithms for multilayer nets
- 1986 – Backpropagation algorithm appears
  - Neural Networks have a resurgence in popularity
  - More computationally expensive

Perceptron and Neural Nets

- **Biological neuron to artificial neuron**

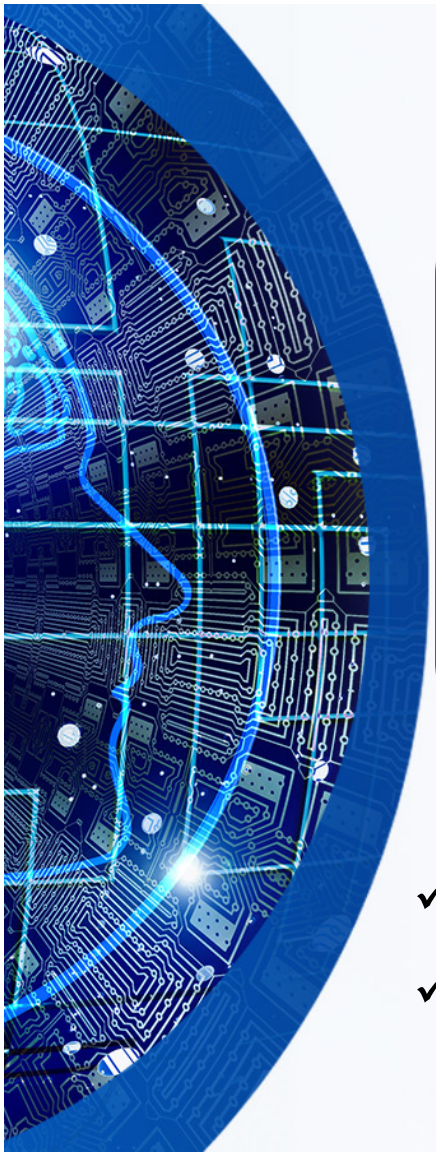






Neural Network Architecture

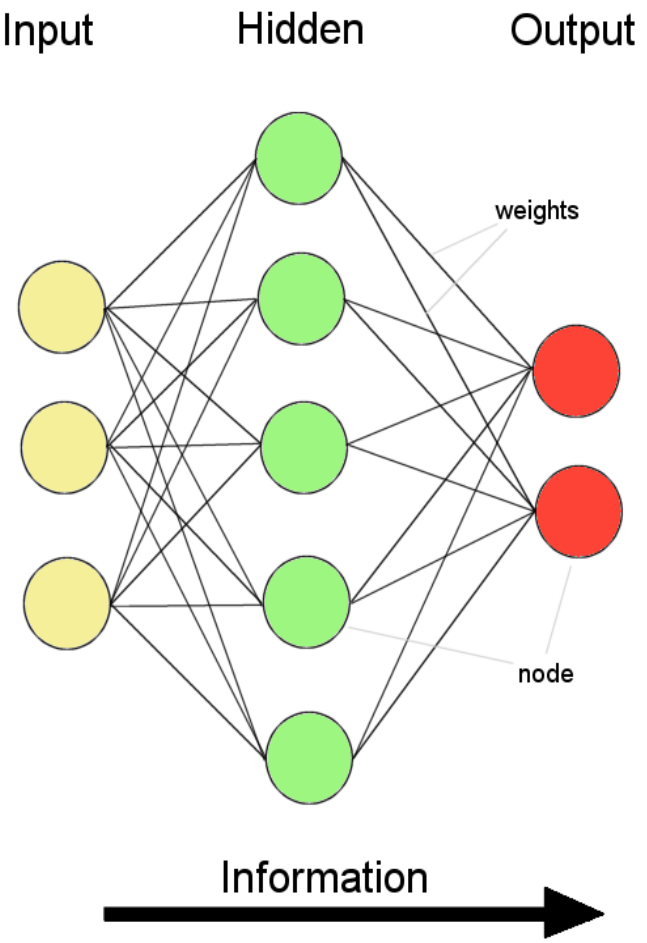
1. # of hidden layers (depth)
2. # of units per hidden layer (width)
3. Type of activation function (nonlinearity)
4. Form of objective function

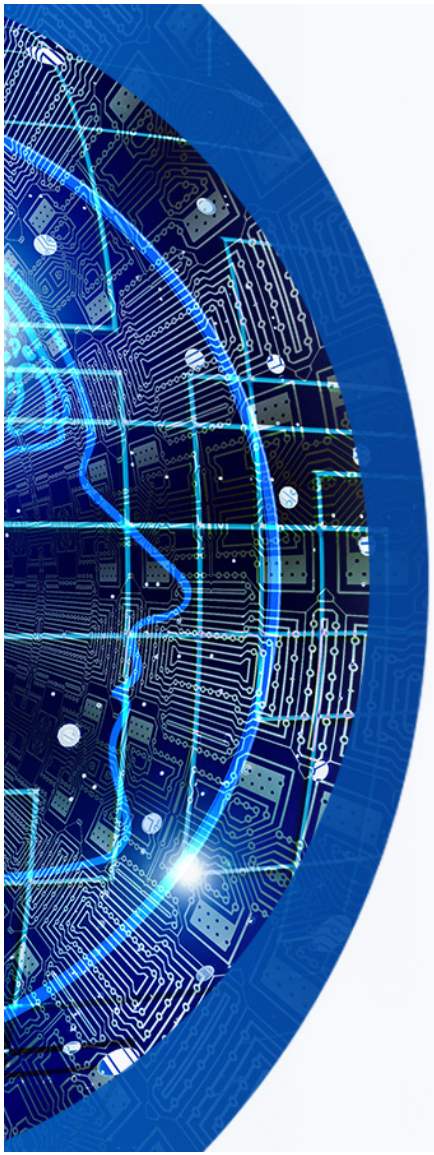


Feed-forward Neural Nets

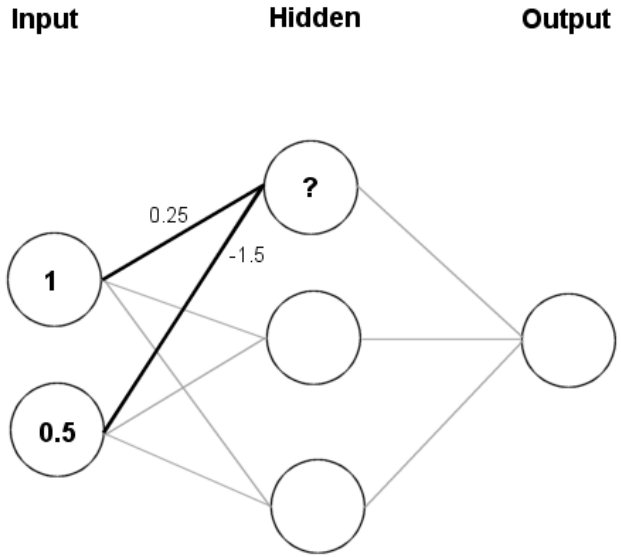
- ❑ **Information flow is unidirectional**
  - ➢ **Data is presented to *Input layer***
  - ➢ **Passed on to *Hidden Layer***
  - ➢ **Passed on to *Output layer***
- ❑ **Information is distributed**
- ❑ **Information processing is parallel**

- ✓ **Neurones (nodes)**
- ✓ **Synapses (weights)**



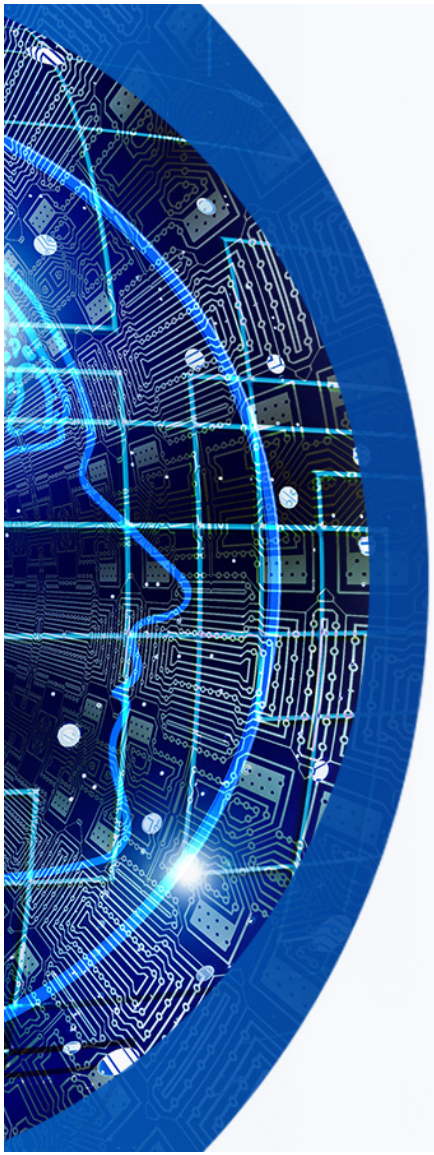


Feeding data through the net



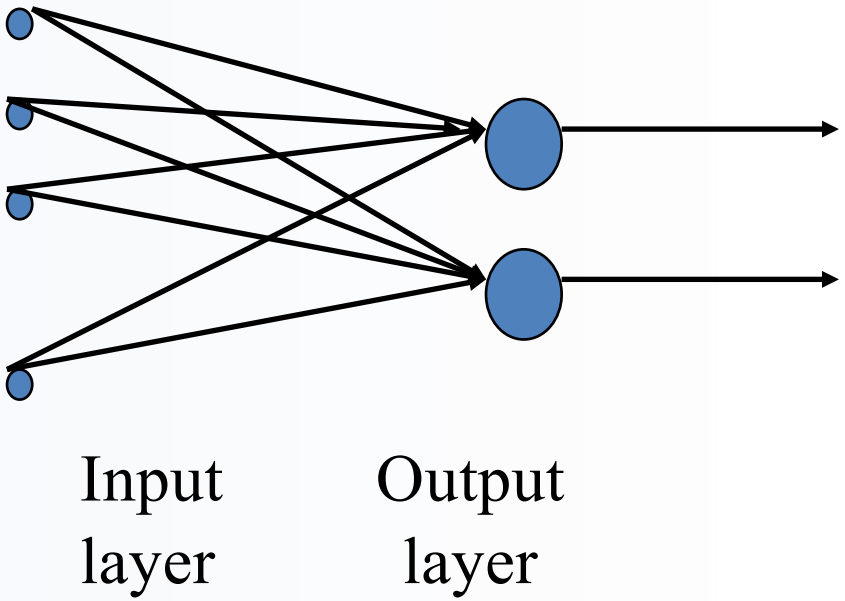
$(1 \times 0.25) + (0.5 \times (-1.5)) = 0.25 + (-0.75) = -\mathbf{0.5}$

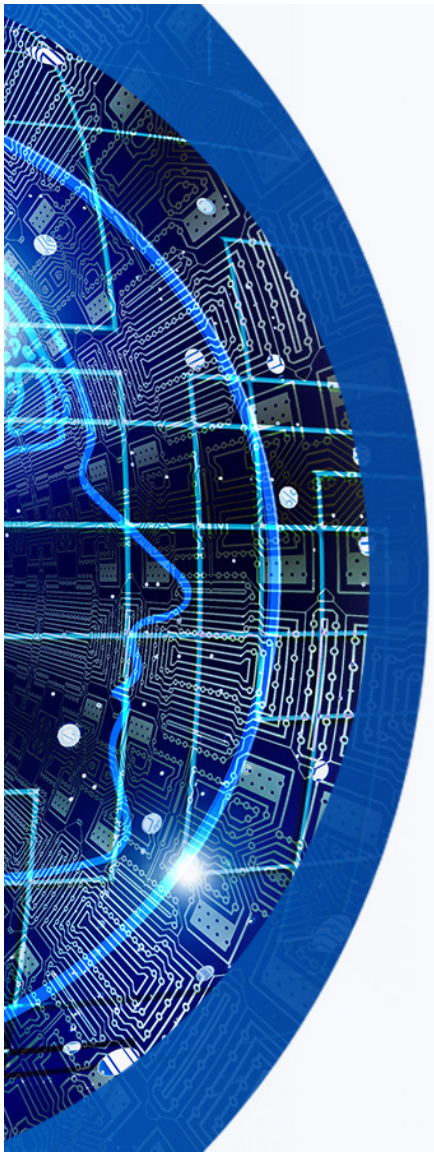
Squashing: $\dfrac{1}{1+e^{0.5}} = 0.3775$



Different Network Topologies

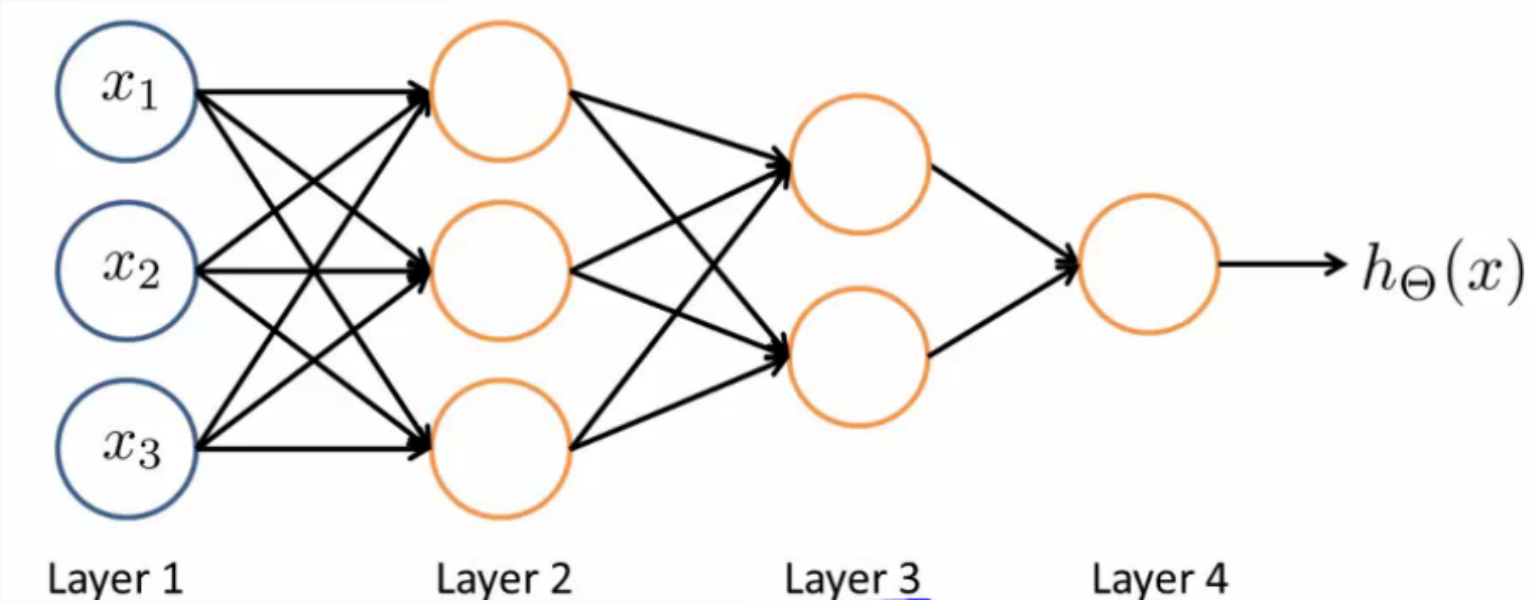
- **Single layer feed-forward network**
  - **Input layer projecting into the output layer**





Different Network Topologies

Multi-layer feed-forward networks

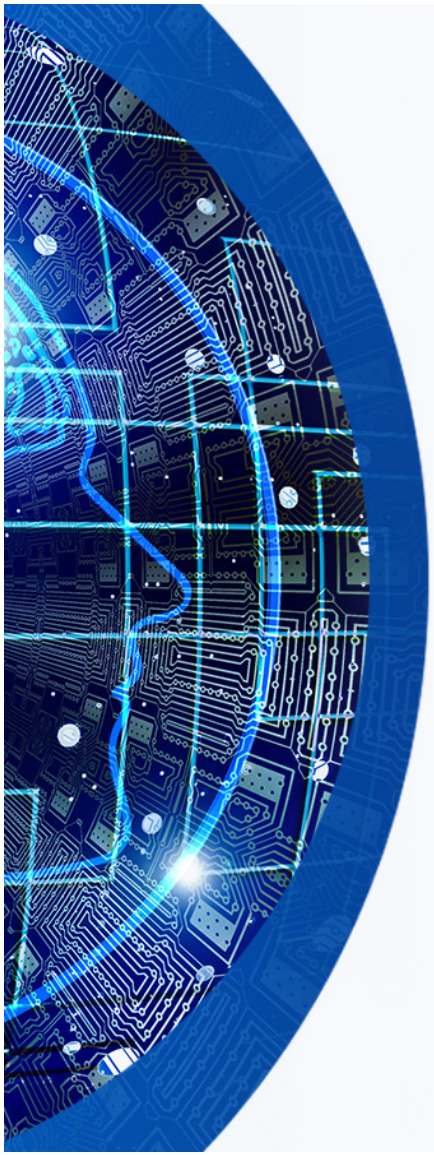


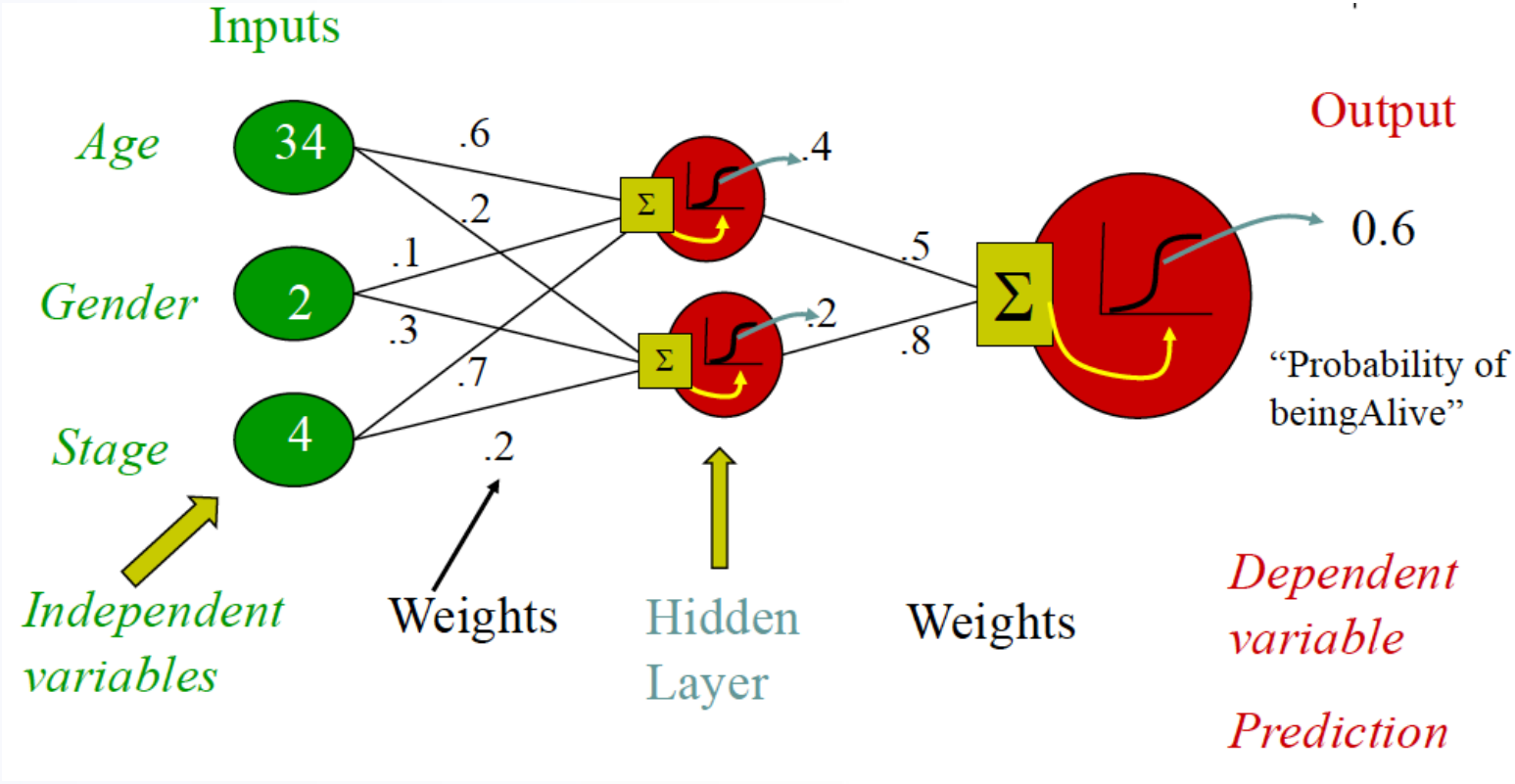
Input layer

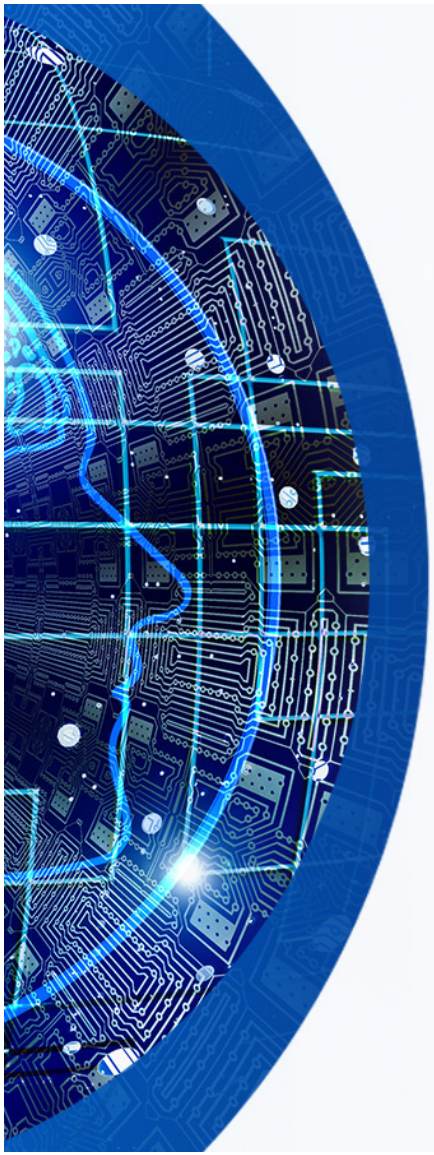
Hidden layers

Output layer

Neural Network Model



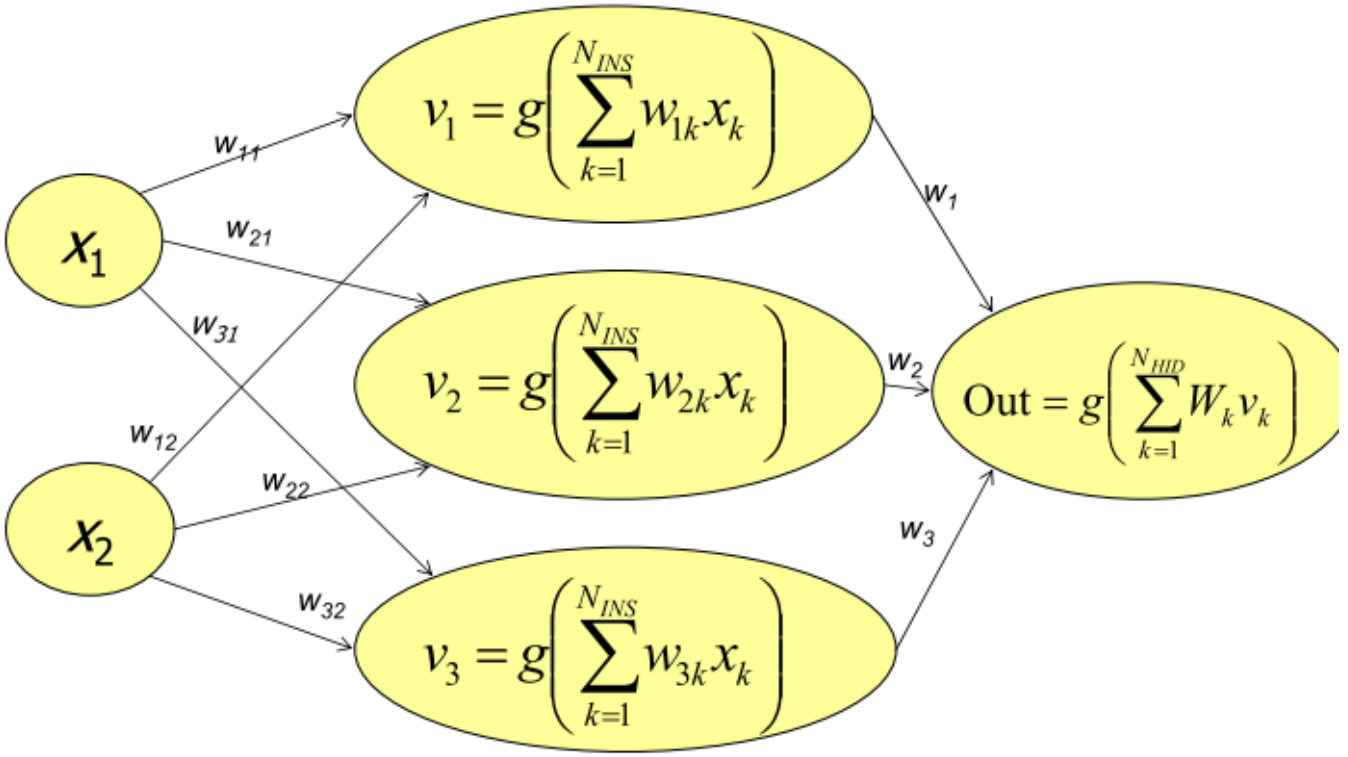


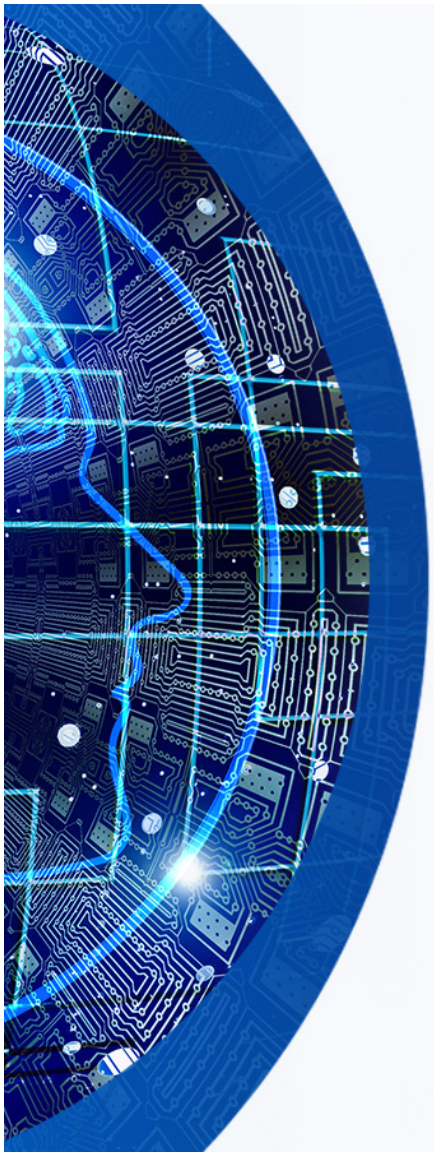


Hidden Layer Net

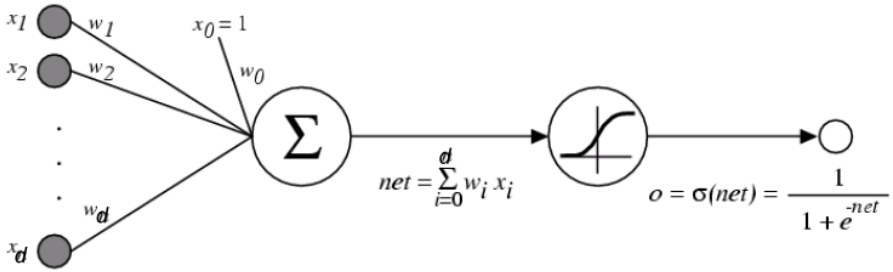
$N_{INPUTS} = 2$

$N_{HIDDEN} = 3$



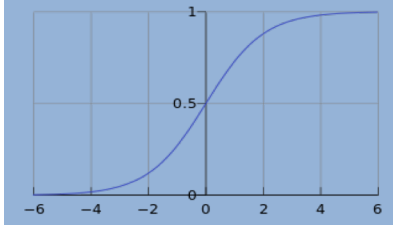


Sigmoid unit



Sigmoid / Logistic Function

$$\text{logistic}(u) \equiv \frac{1}{1+e^{-u}}$$



$\sigma(x)$ is the sigmoid function/activation function (also linear, threshold)

$$\frac{1}{1+e^{-x}}$$

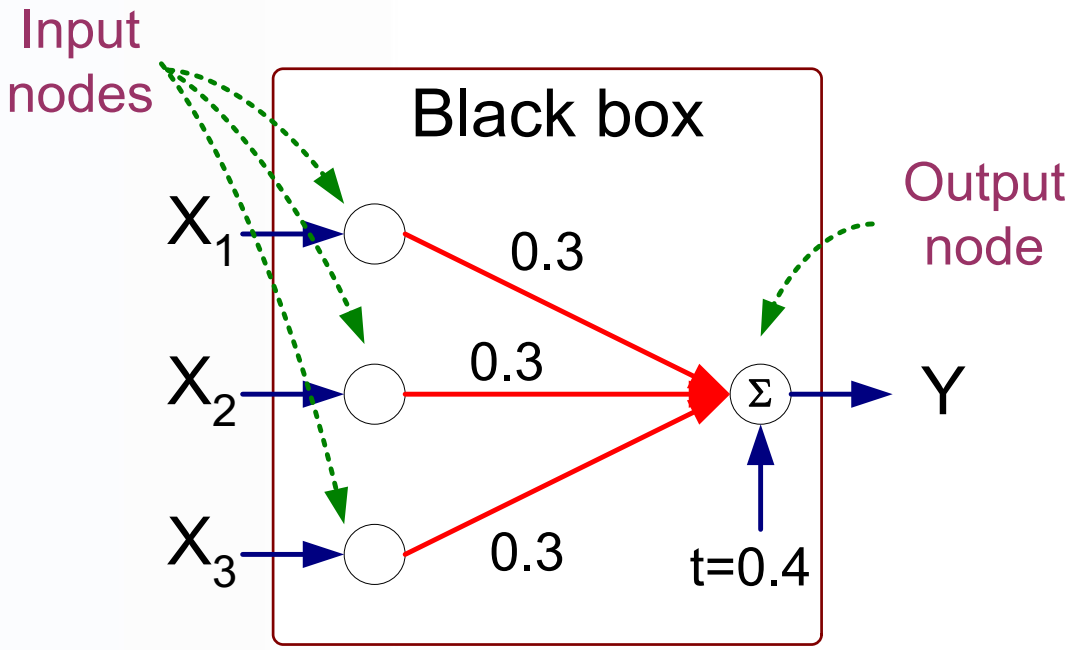
Nice property: $\frac{d\sigma(x)}{dx} = \sigma(x)(1-\sigma(x))$ Differentiable

We can derive gradient decent rules to train

- One sigmoid unit
- *Multilayer networks* of sigmoid units → Backpropagation

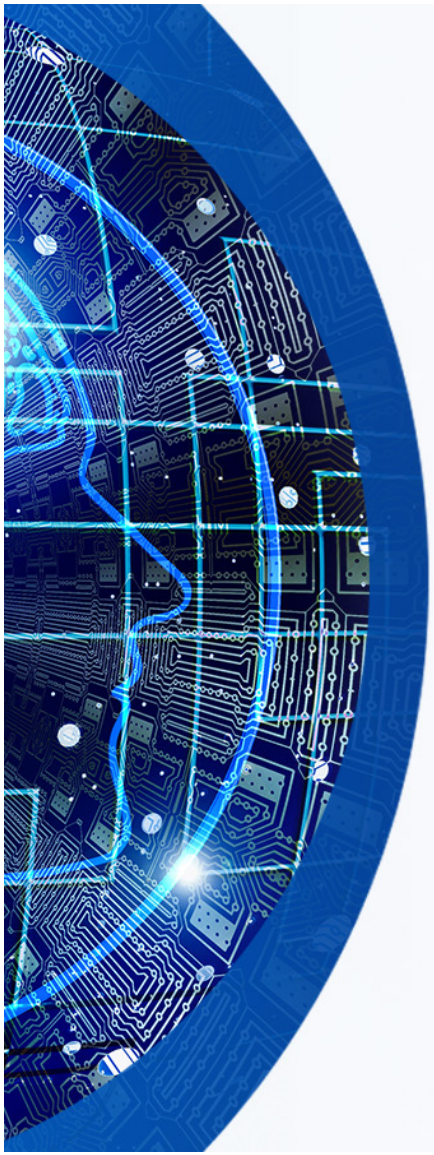
Artificial Neural Networks (ANN)

X_1	X_2	X_3	Y
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1
0	0	1	0
0	1	0	0
0	1	1	1
0	0	0	0

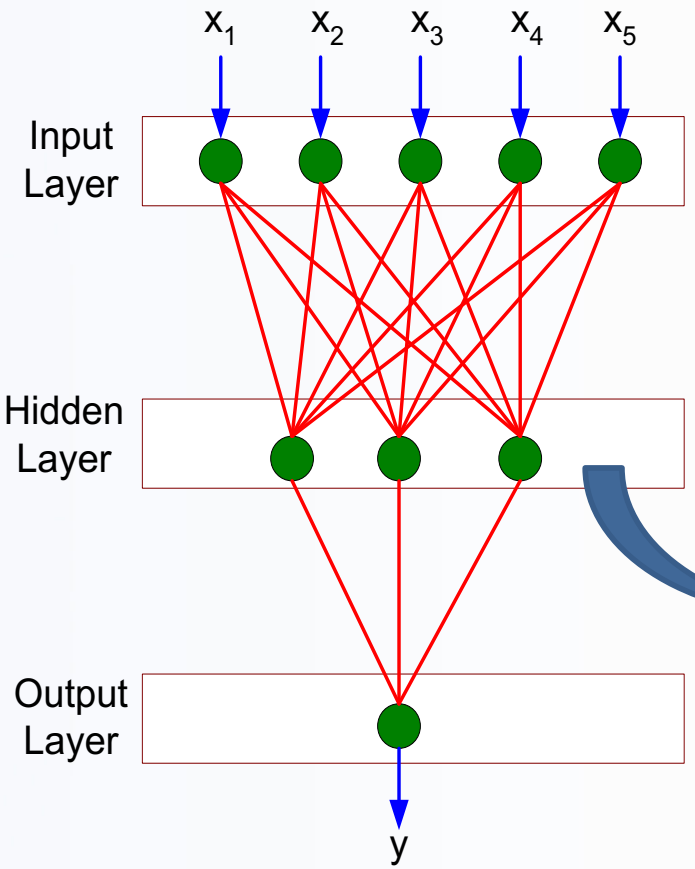


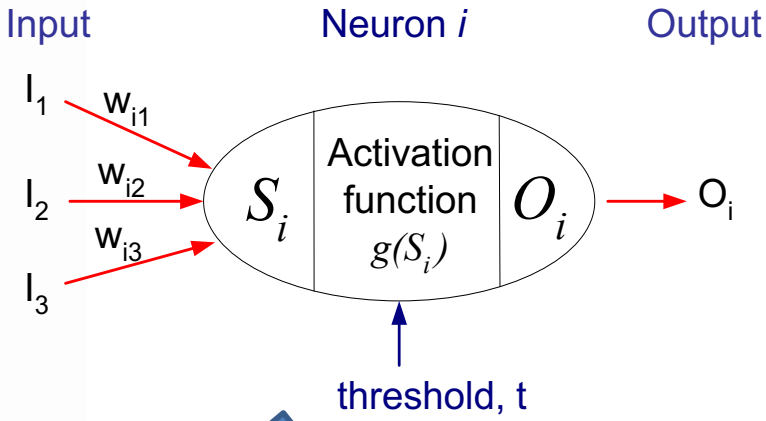
$$Y = I(0.3X_1 + 0.3X_2 + 0.3X_3 - 0.4 > 0)$$

$$\text{where } I(z) = \begin{cases} 1 & \text{if } z \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

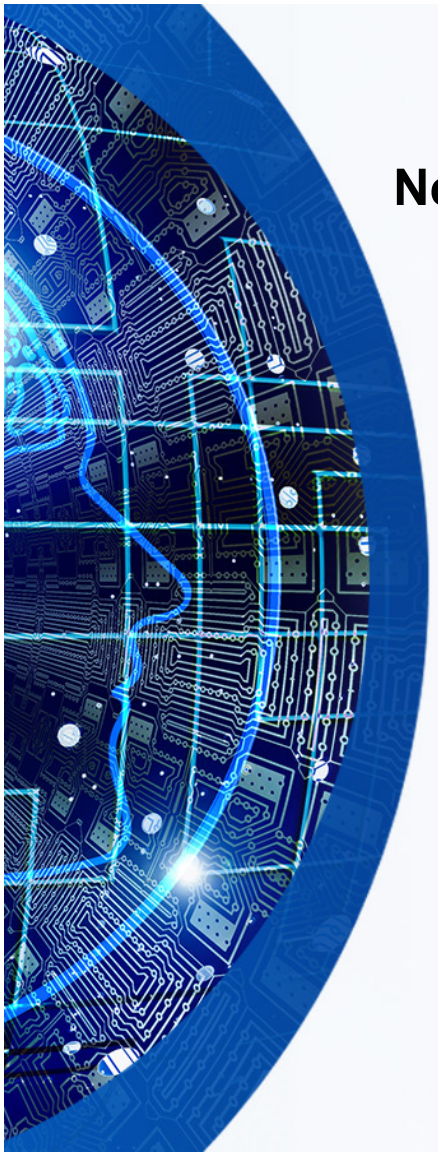


ANN Structure



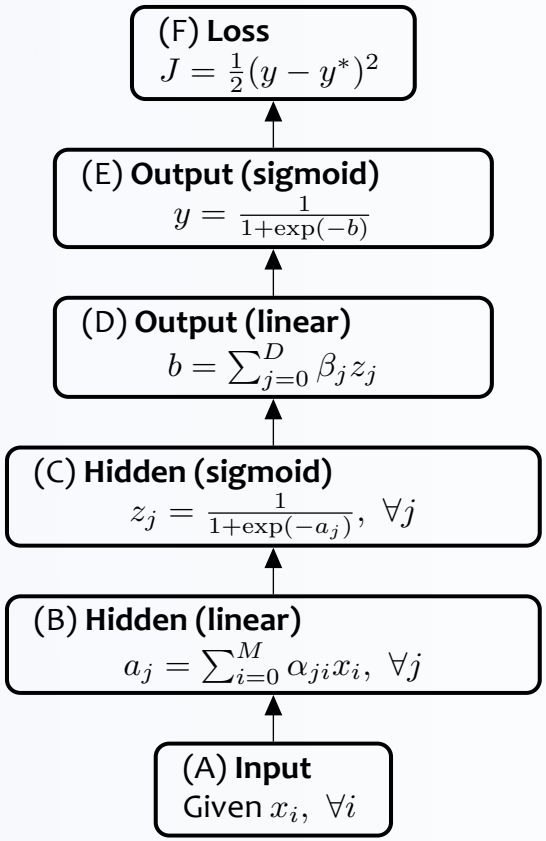


Training ANN means learning the weights of the neurons

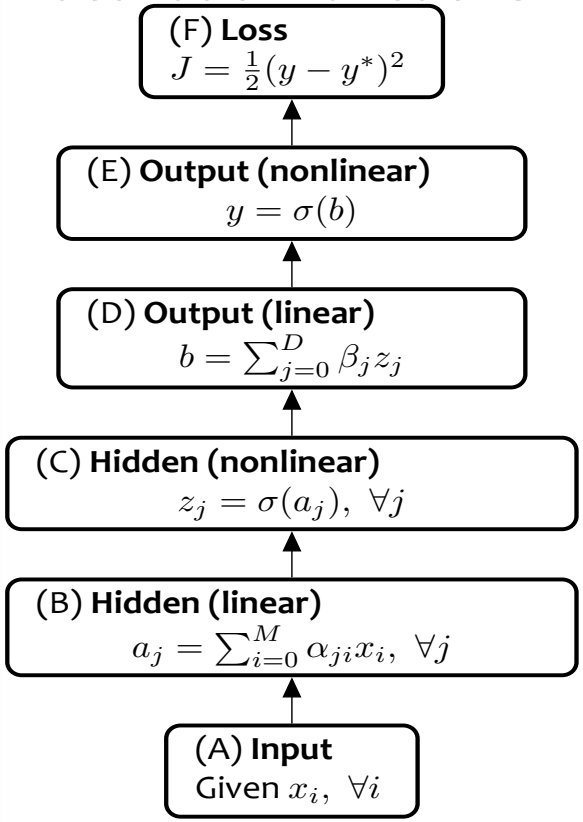


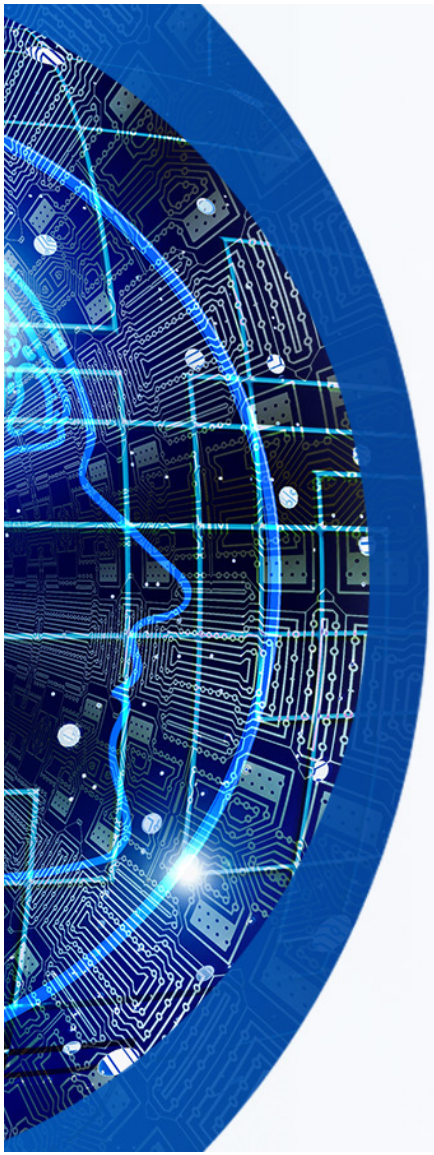
Activation Functions

Neural Network with sigmoid activation functions



Neural Network with arbitrary nonlinear activation functions





Summary

- **Neural Networks…**
  - provide a way of learning features
  - are highly nonlinear prediction functions
  - (can be) a highly parallel network of logistic regression classifiers
  - discover useful hidden representations of the input
- **Back propagation…**
  - provides an efficient way to compute gradients
  - is a special case of reverse-mode automatic differentiation