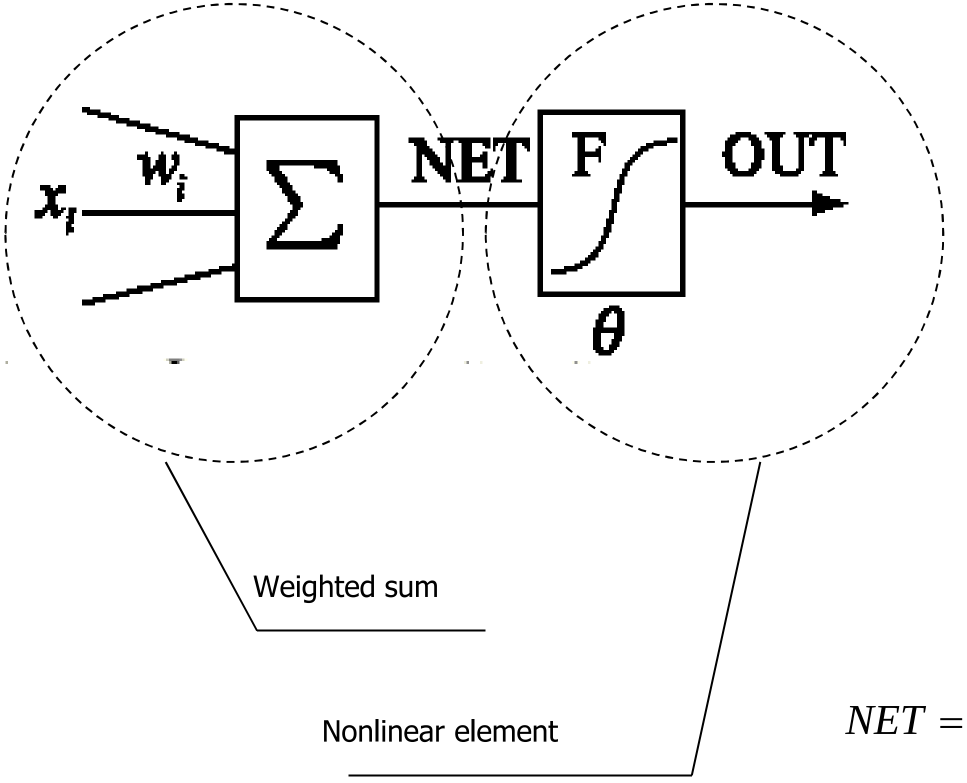
1918
TALLINNA
TEHNIKAÜLIKOOL

Modelleerimine ja Juhtimine Tehisnärvivõrgudega

Identification and Control with artificial neural networks

Eduard Petlenkov,

Artificial neuron



Weighted sum

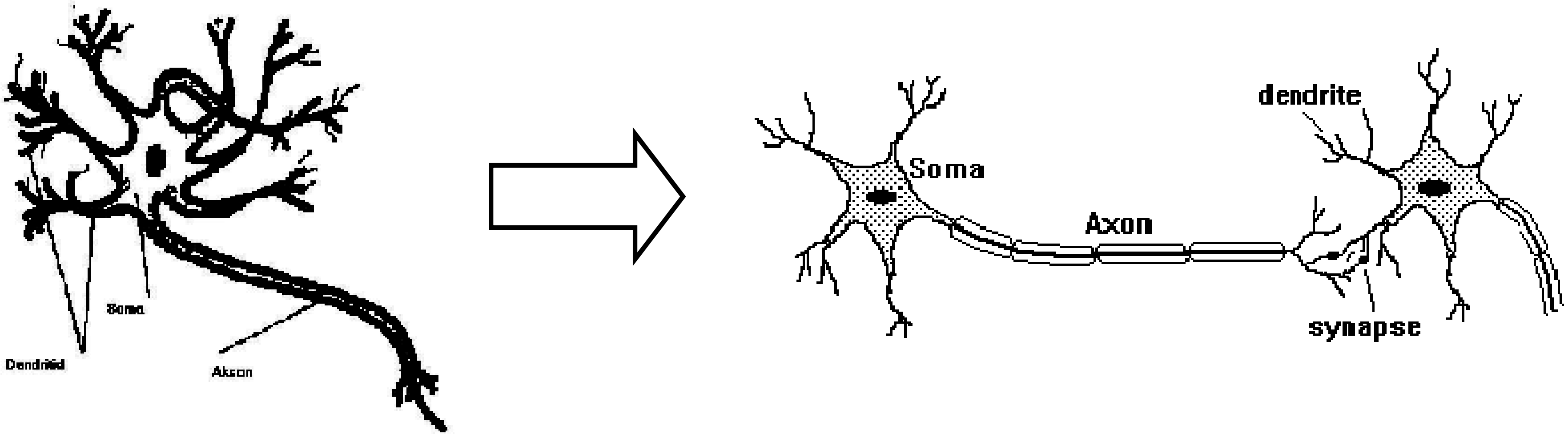
Nonlinear element

Input vector:
$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

Vector of weight coeffitients:
$$W = [w_1 \ldots w_n]$$

Weighted sum:
$$NET = W \cdot X = [w_1 \ldots w_n] \cdot \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = w_1 x_1 + \ldots + w_n x_n$$

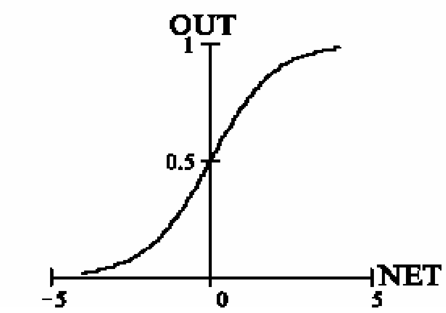
Biological neuron and biological neural networks



Activation functions (1)

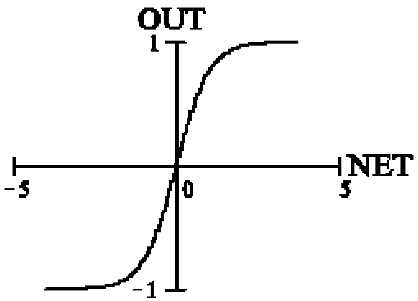
OUT=f(NET)

Sigmoid functions are having an "S" shape (**sigmoid curve**)



1 Logistic function

$$OUT = \frac{1}{1+e^{-NET}}$$



2 Hyperbolic tangent

$$OUT = \frac{e^{NET} - e^{-NET}}{e^{NET} + e^{-NET}}$$

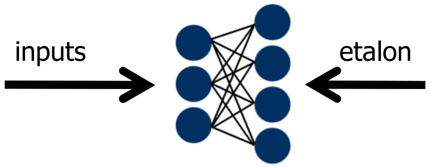
Types of artificial neural networks

Structure:

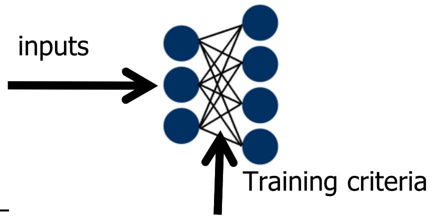
- Feedforward
- Feedback
  - internal feedback
  - external feedback

Training/ learning:

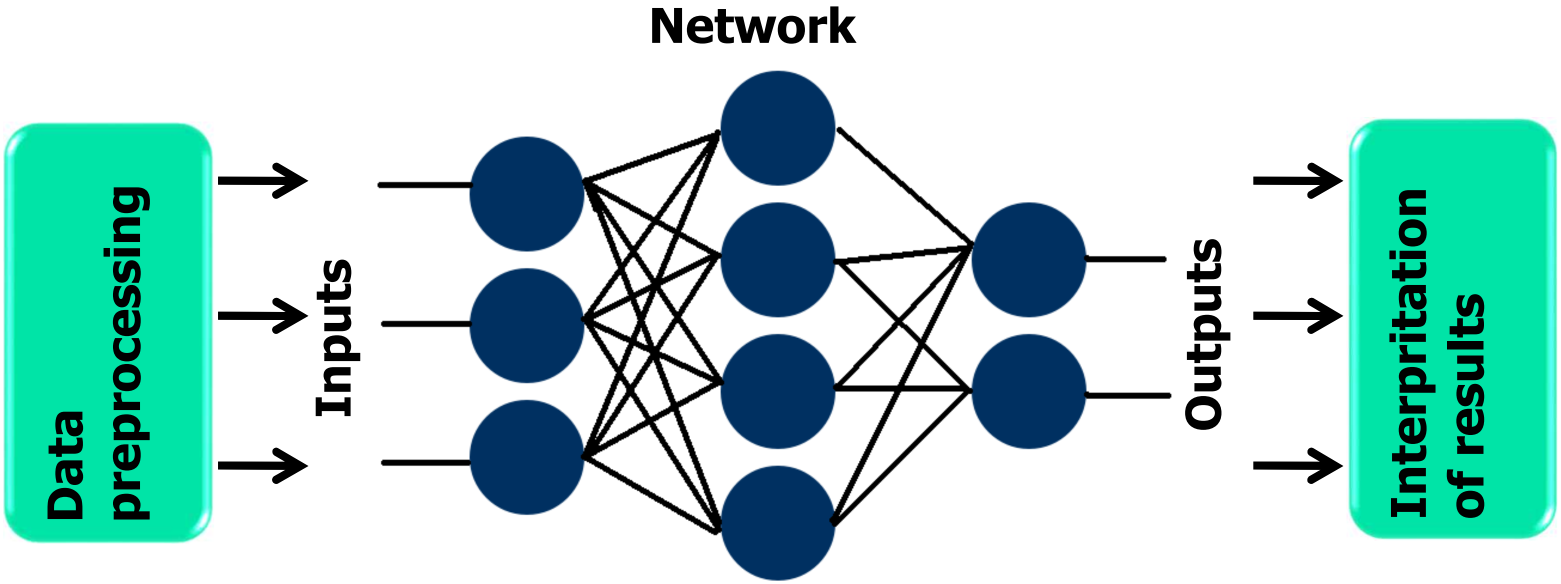
- Supervised learning



- Selflearning (unsupervised)

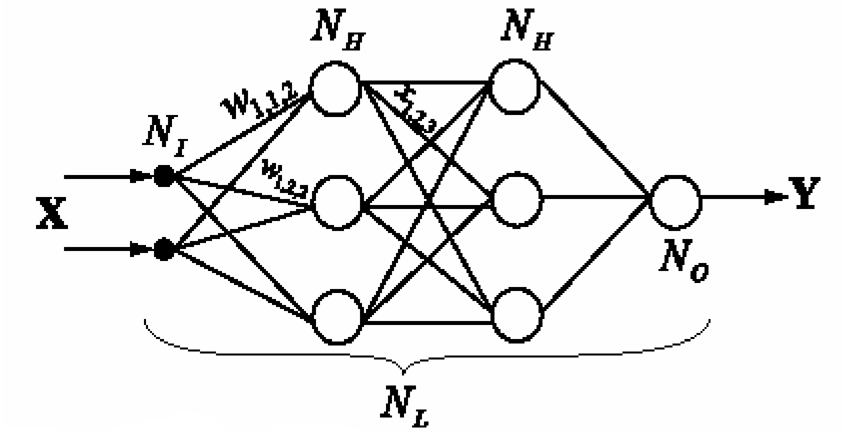


How to use artificial neural networks



Feedforward neural networks and multilayer perceptron

Feedforward networks are networks in which an output of a neuron can be connected only with an input of a next layer neuron.



"from each to each"

N_I - Input layer

N_O - Output layer

N_H - Hidden layer

w_{ijl} - Weighting coefficients, where

i is the number of the neuron's input

j - neuron's number in the layer

l - number of the layer

Mathematical function of a two-layer perceptron

$$X=\begin{bmatrix}x_1\\ \vdots\\ x_n\end{bmatrix};\quad Y=\begin{bmatrix}y_1\\ \vdots\\ y_m\end{bmatrix};\quad W_1=\begin{bmatrix}w_{111} & \cdots & w_{n11}\\ \vdots & \ddots & \vdots\\ w_{1k1} & \cdots & w_{nk1}\end{bmatrix};\quad \Theta_1=\begin{bmatrix}\theta_{11}\\ \vdots\\ \theta_{k1}\end{bmatrix};\quad W_2=\begin{bmatrix}w_{112} & \cdots & w_{k12}\\ \vdots & \ddots & \vdots\\ w_{1m2} & \cdots & w_{km2}\end{bmatrix};\quad \Theta_2=\begin{bmatrix}\theta_{12}\\ \vdots\\ \theta_{m2}\end{bmatrix};$$

F_1 - Activation function of the hiddent layer neurons;

F_2 - Activation function of the output layer neurons.

$$Y=\underbrace{F_2(W_2(\underbrace{F_1(W_1X-\Theta_1))}_{output\quad of\quad the\quad first\quad layer}-\Theta_2)}_{output\quad of\quad the\quad network}$$