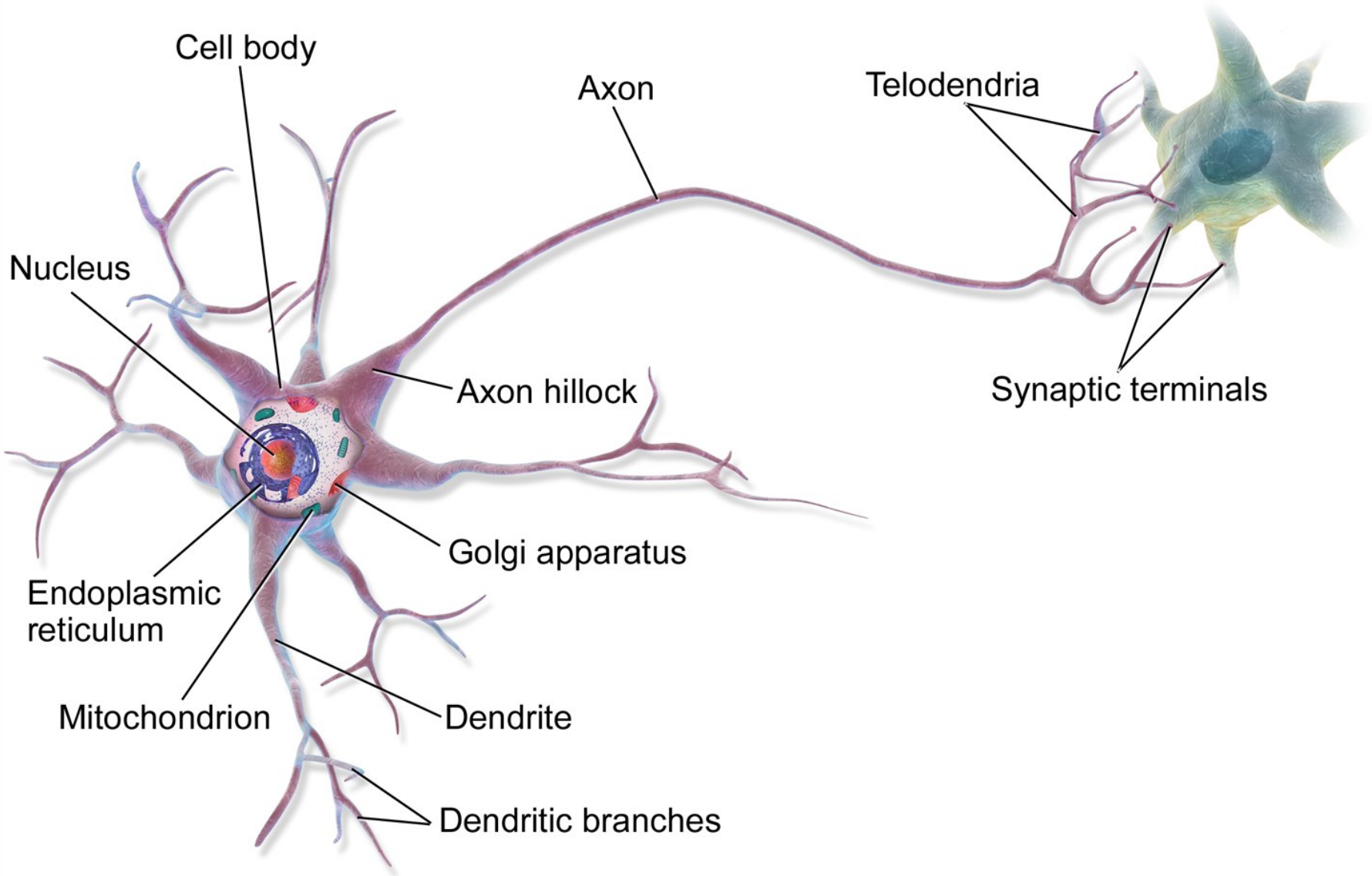
Невронни мрежи - въведение

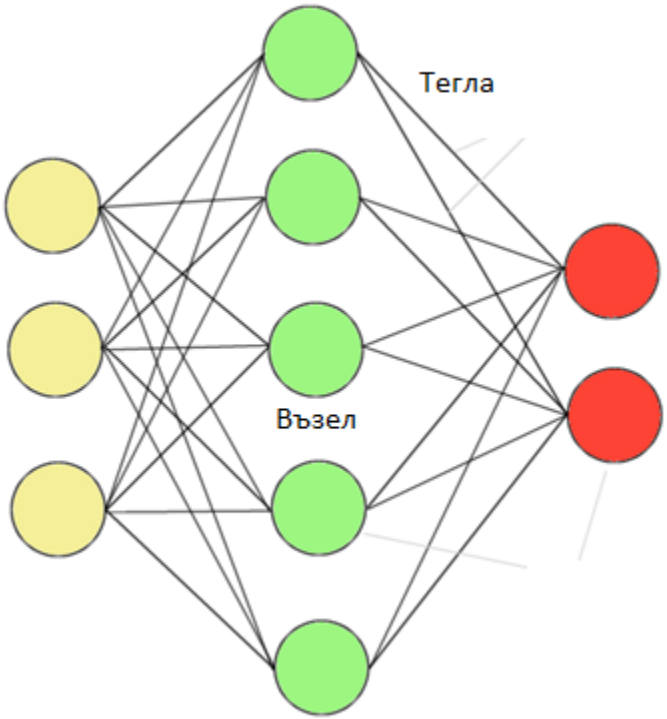
Станимир Кабаиванов

WT*…


Cell body
Axon
Telodendria
Nucleus
Axon hillock
Synaptic terminals
Golgi apparatus
Endoplasmic reticulum
Mitochondrion
Dendrite
Dendritic branches

Невронни мрежи

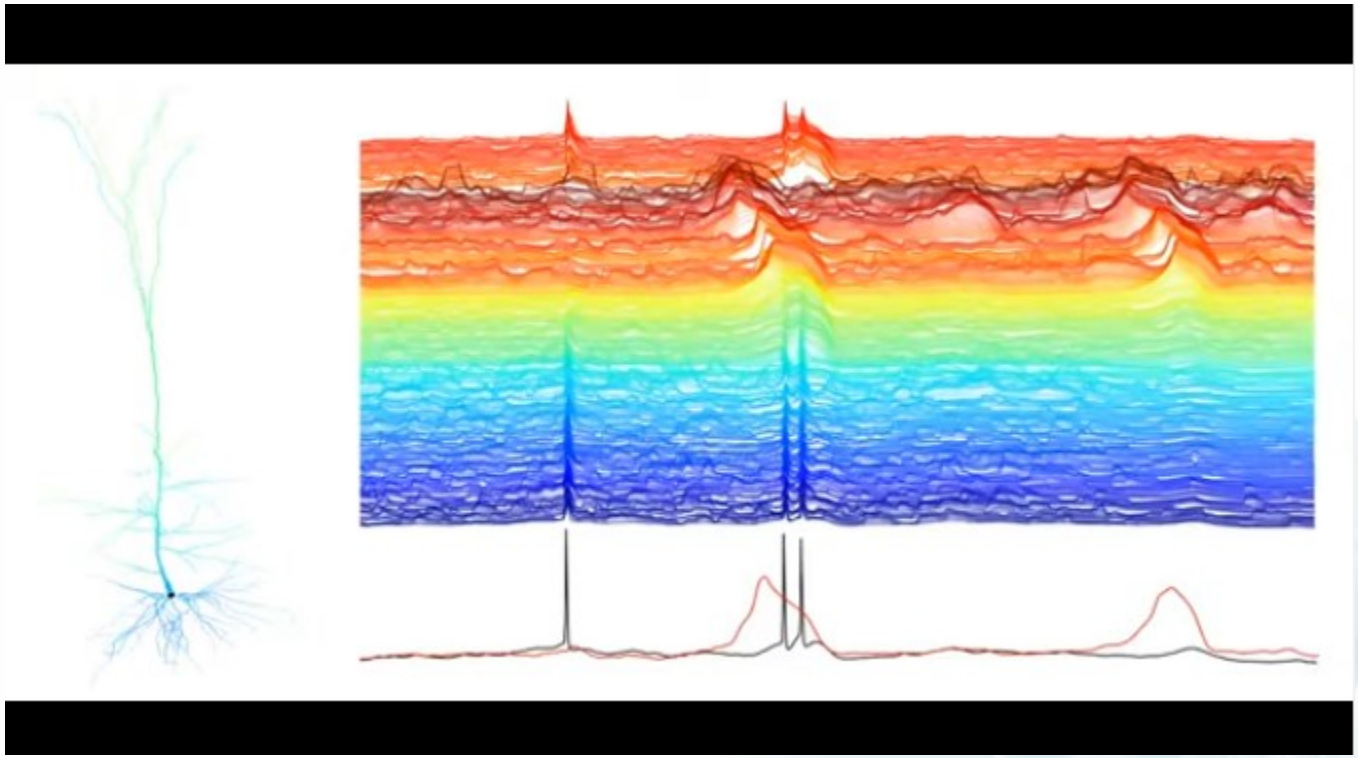
- Изкуствените НМ включват два основни компонента, които са заимствани от естествените



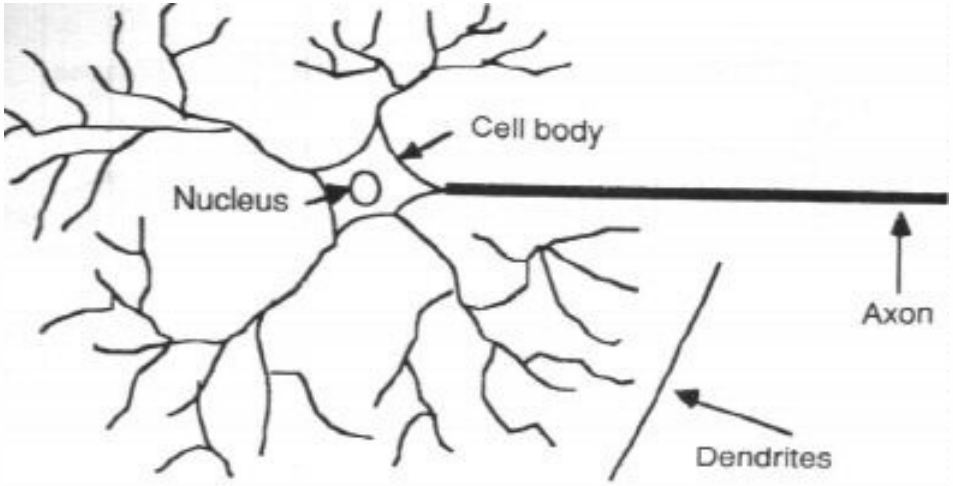
1. Неврони (възли)
2. Синапси (тегла)

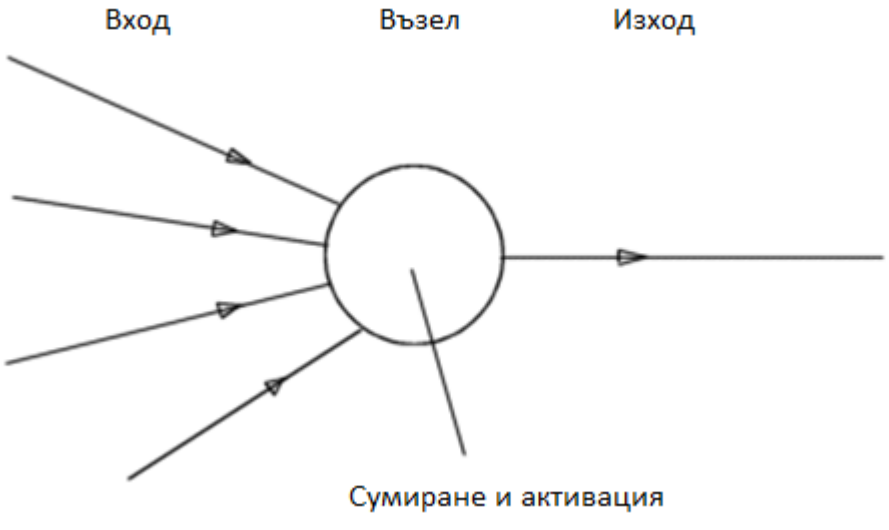
Къде сме ние?

https://youtu.be/3LQLCqHT5Ws



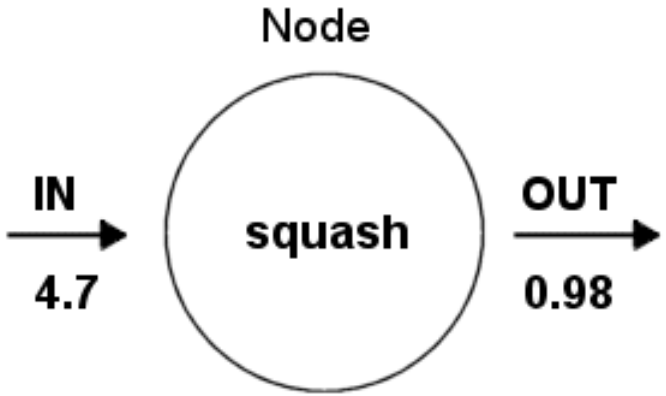
Неврони vs Възли



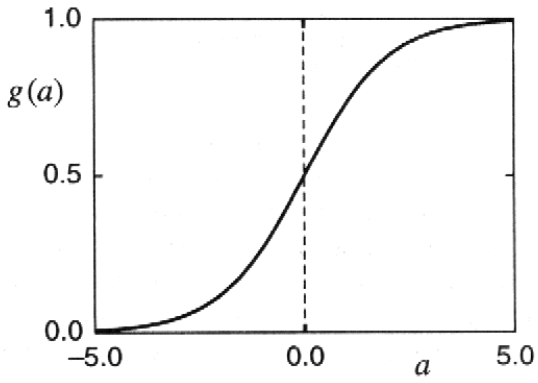


Сумиране и активация

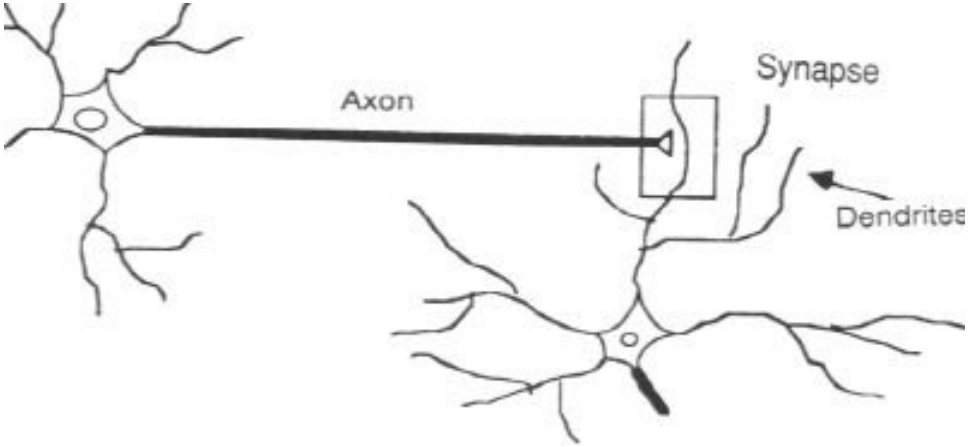
Структура на невроните

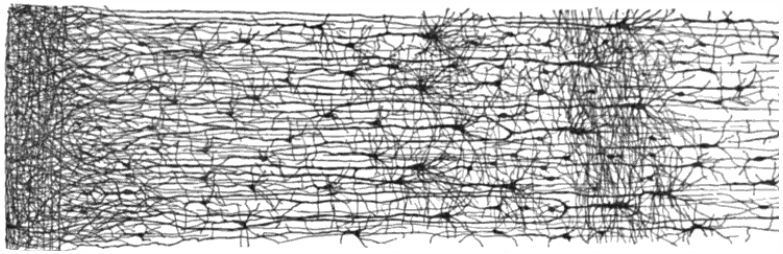


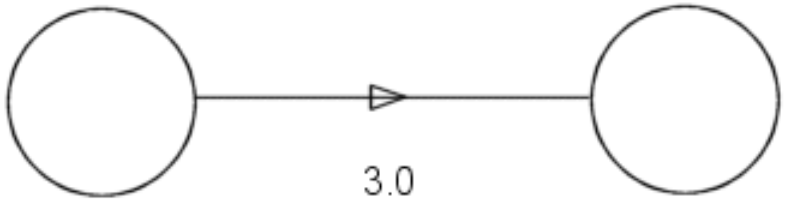
Активираща функция (ограничена, непрекъсната):



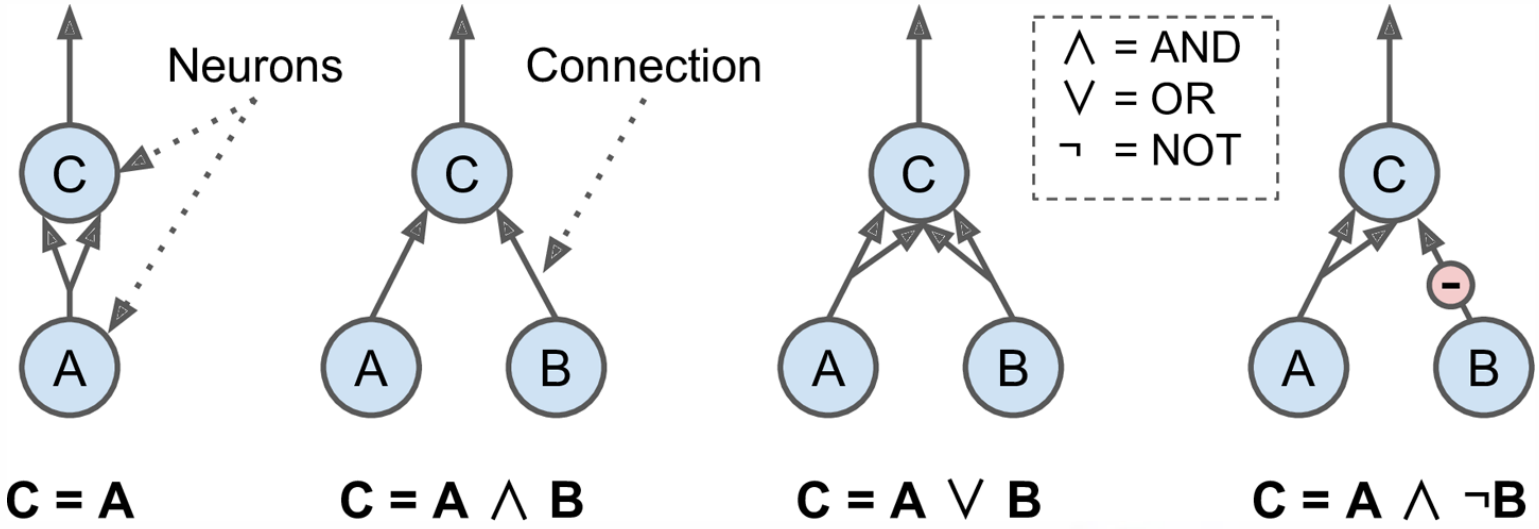
От неврони към мрежи…







Логически операции



Мрежа на McCulloch и Pitts

За да се активира неврон, то трябва поне два от „входовете“ да са активни.

logical_ops DEMO…..

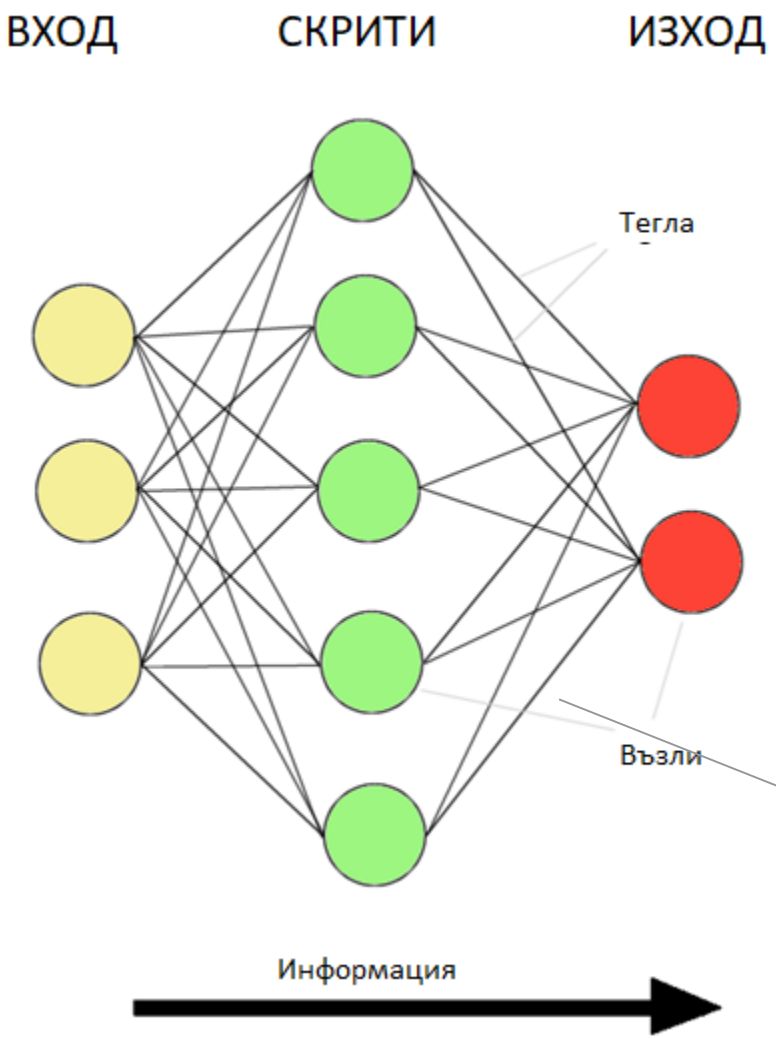
Ориентири в мрака…

Топология

Алгоритми

Защо го правим?

Напред и само напред!



Движение само в посока напред

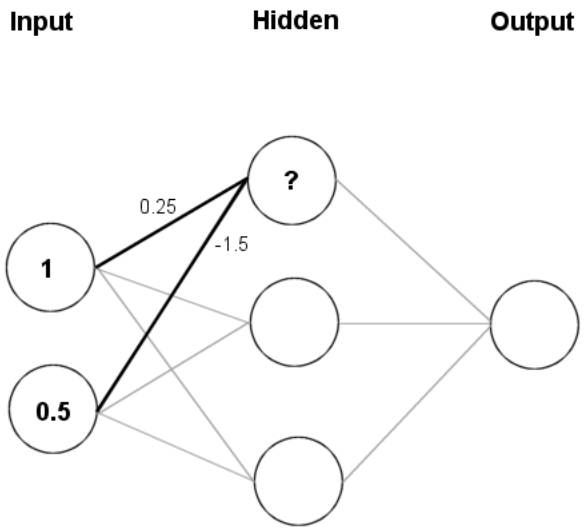
1) *Входно ниво*
2) *Скрито ниво (или нива)*
3) *Изходно ниво*

„Знанието“ се разпределя.

Обработката е (псевдо)паралелна

Вътрешно представяне на информацията

Пример…



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

$= \mathbf{-0.5}$

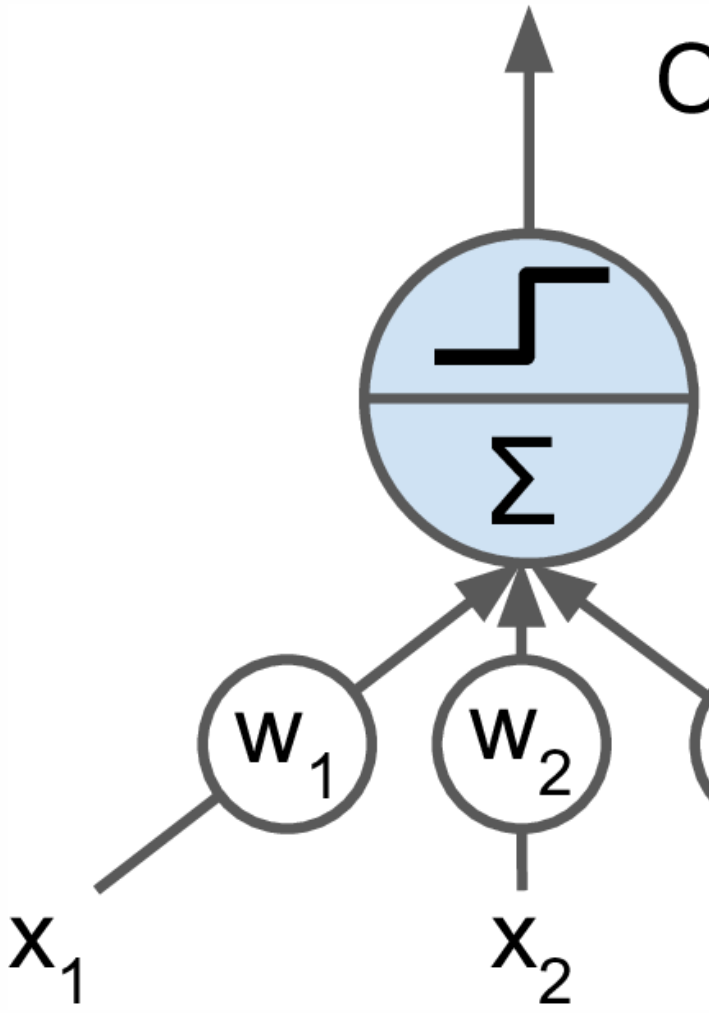
Активация: $\frac{1}{1+e^{0.5}} = 0.3775$

Къде е най-важното?

- Теглата определят поведението на мрежата!

→ Как да намерим „точните“ тегла?

Перцептрони - LTU



Output: $h_{\mathbf{w}}(\mathbf{x}) = \text{step}(\mathbf{w}^T . \mathbf{x})$

Step function: step(z)

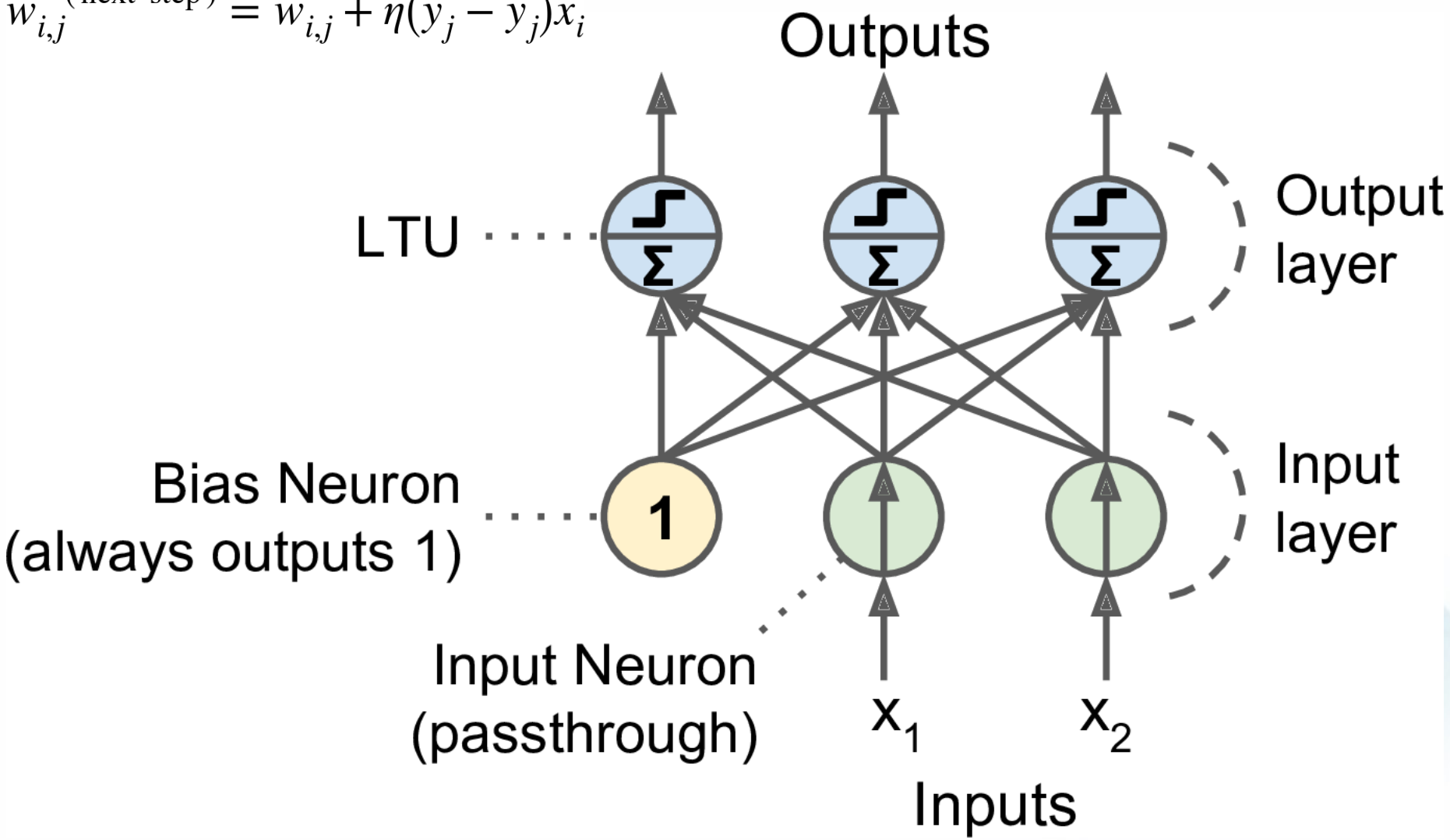
Weighted sum: $z = \mathbf{w}^T . \mathbf{x}$

Weights

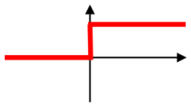
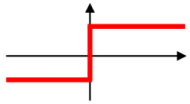
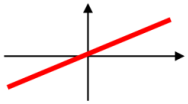
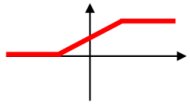
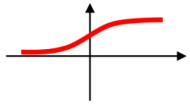
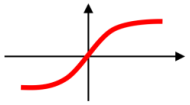
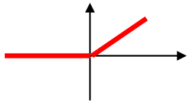
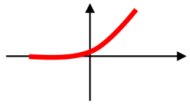
Inputs

Перцептрони

$$w_{i,j}^{(\text{next step})} = w_{i,j} + \eta(y_j - \hat{y}_j)x_i$$



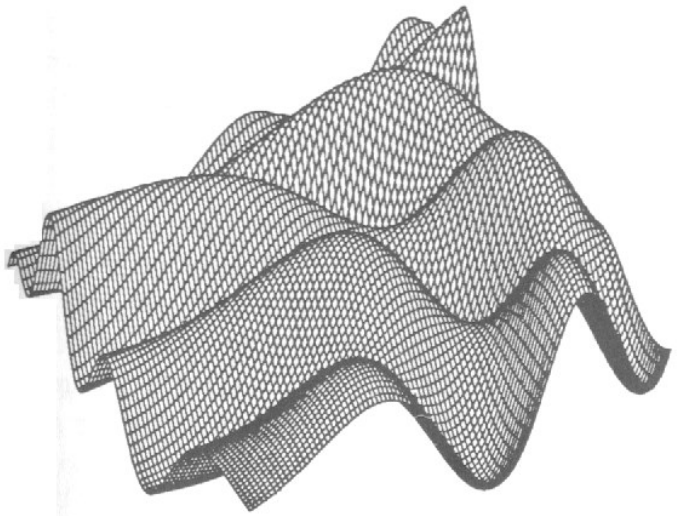
Перцептрони

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$	Multi-layer Neural Networks	
Rectifier, ReLU (Rectified Linear Unit)	$\phi(z) = max(0, z)$	Multi-layer Neural Networks	
Rectifier, softplus	$\phi(z) = \ln(1 + e^{z})$	Multi-layer Neural Networks	

Обучение на НМ

- Обучение с обратно разпространение на грешката
  - Нужно е обучаващо множество (вход / изход)
  - Започва със случайни стойности на теглата
  - Променят се според грешките (контролирано обучение)

  → Градиентно спускане по „картата на грешките“

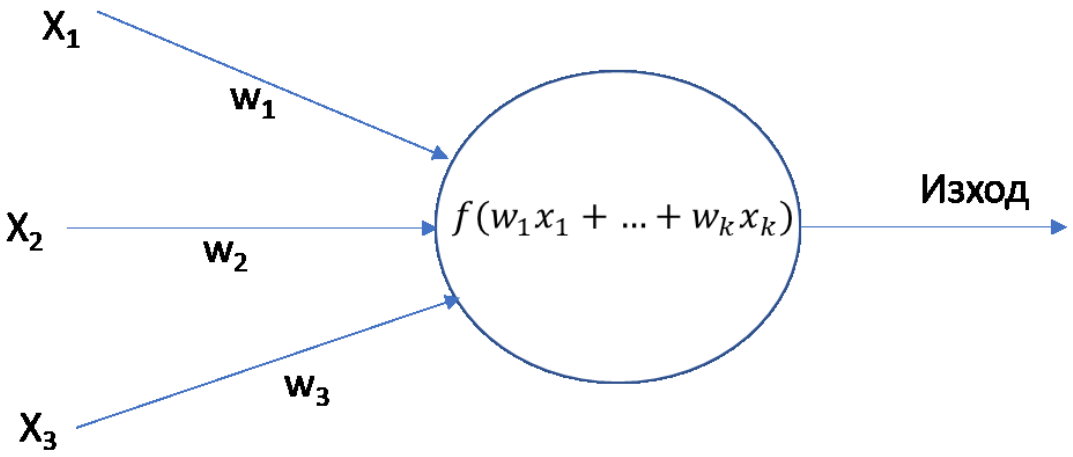


Обучение на НМ

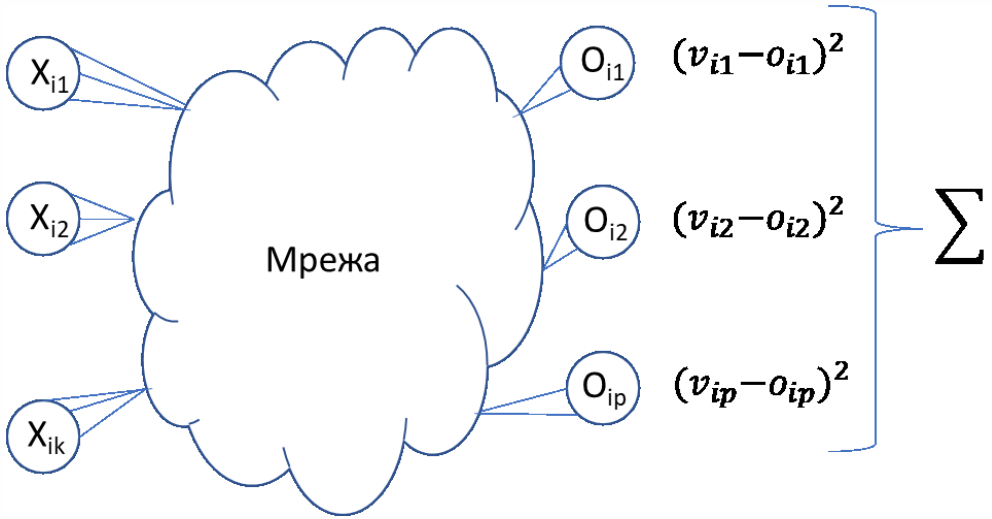
$$f(x) = \frac{1}{1+e^{-(x)}}, \quad (1)$$

$$\cdot f(x)) = \frac{e^{-x}}{(1+e^{-x})^2}, \text{откъдето}$$

$$'_k x_k) = \frac{1}{1-e^{-(w_1x_1+\ldots+w_kx_k)}} \quad (2)$$



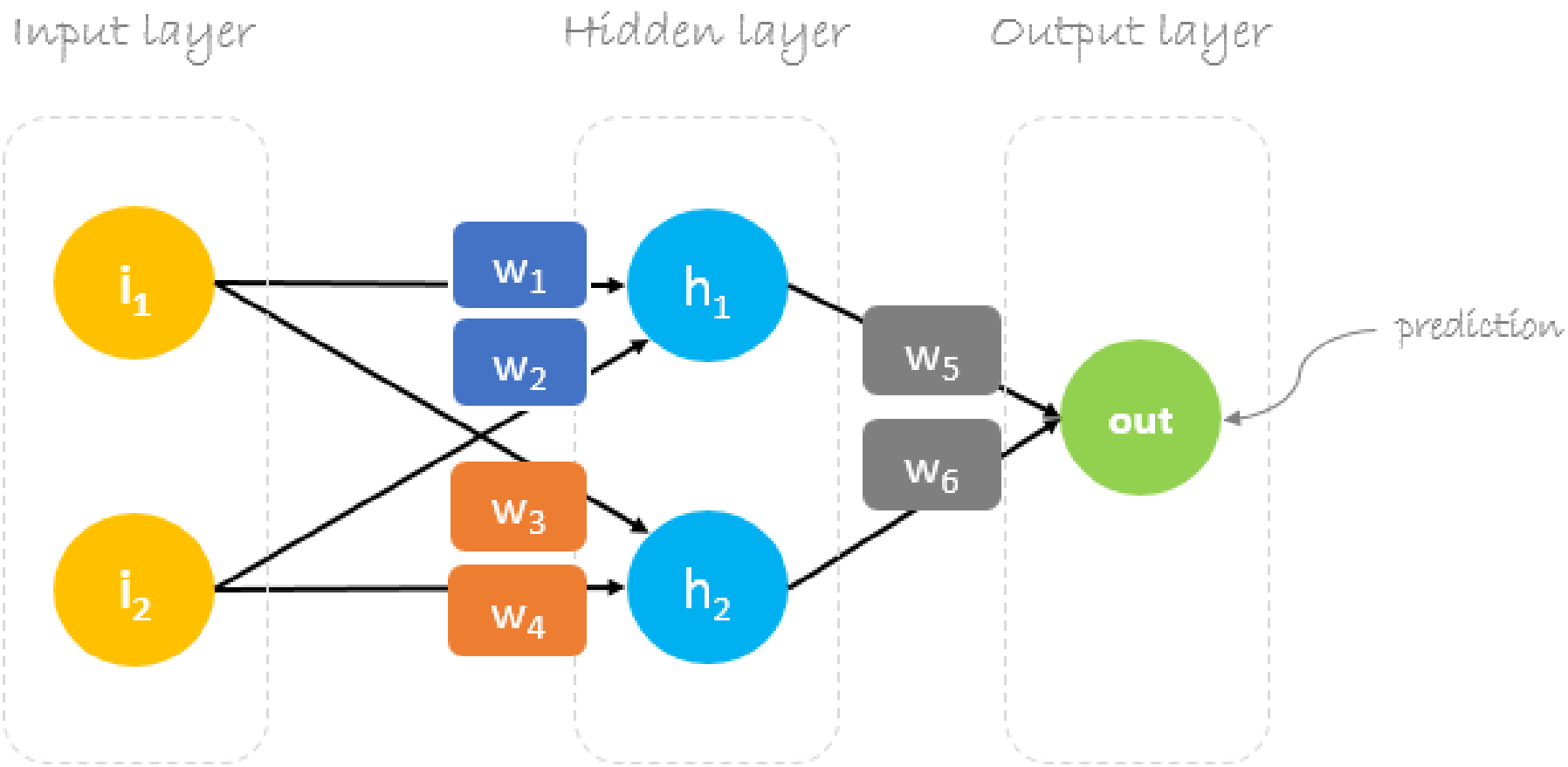
$$\mathrm{E} = \frac{1}{2}\sum_{j=1}^{p}\left\|v_{ij} - o_{ij}\right\|^2$$



$$\nabla E = \left(\frac{dE}{dw_1}, \frac{dE}{dw_2}, \ldots, \frac{dE}{dw_l}\right)$$

$$\Delta w_i = -\gamma \frac{dE}{dw_i}$$

Пример 1


Input layer
Hidden layer
Output layer
i_1
i_2
w_1
w_2
w_3
w_4
h_1
h_2
w_5
w_6
out
prediction

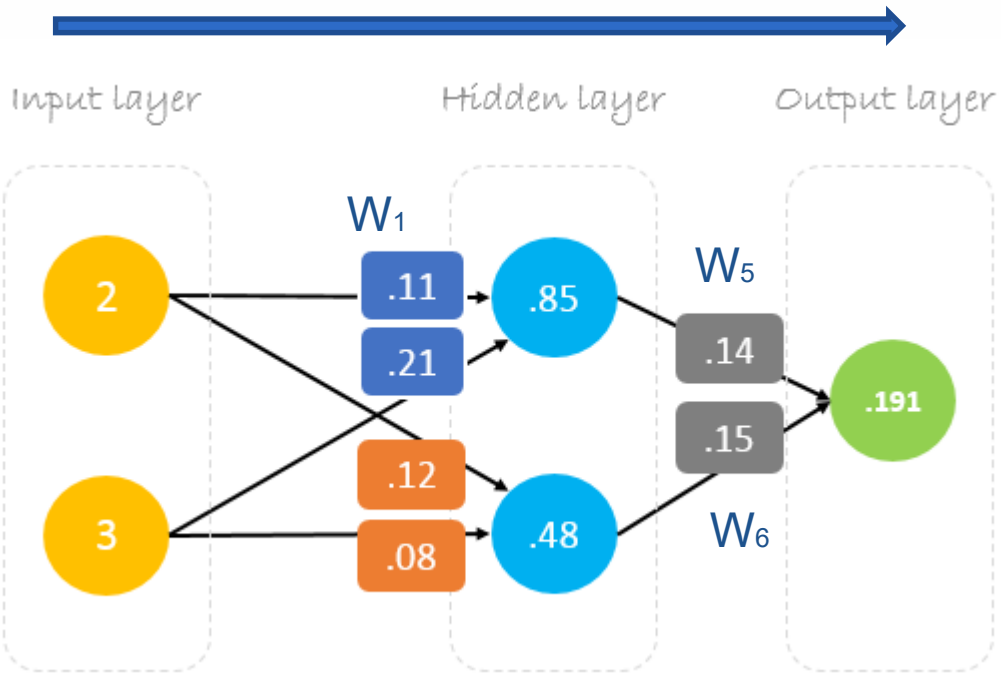
Пример 1

2*0.11+3*0.21 = 0.85
2*0.12+3*0.08 = 0.48

0.85*0.14+0.48*0.15 = 0.191

Но реалната стойност е: 1

Грешка: 0.5*(0.191-1)^2=0.327



Пример 1

Грешка: 0.5*(0.191-1)^2=0.327

2*0.11+3*0.21 = 0.85
2*0.12+3*0.08 = 0.48

0.85*0.14+0.48*0.15 = 0.191

Но реалната стойност е: 1

$$\text{prediction} = \text{out}$$

$$\text{prediction} = (h_1)\, w_5 + (h_2)\, w_6$$

$$h_1 = i_1 w_1 + i_2 w_2$$
$$h_2 = i_1 w_3 + i_2 w_4$$

$$\text{prediction} = (i_1 w_1 + i_2 w_2)\, w_5 + (i_1 w_3 + i_2 w_4)\, w_6$$

Пример 1

Грешка: 0.5*(0.191-1)^2=0.327

$$^{*}W_6 = W_6 - a\left(\frac{\partial Error}{\partial W_6}\right)$$

2*0.11+3*0.21 = 0.85
2*0.12+3*0.08 = 0.48

0.85*0.14+0.48*0.15 = 0.191

Но реалната стойност е: 1

$$\frac{\partial Error}{\partial W_6} = \frac{\partial Error}{\partial prediction} * \frac{\partial prediction}{\partial W_6}$$

chain rule

$$Error = \frac{1}{2}(prediction - actual)^2$$

$$prediction = (i_1 w_1 + i_2 w_2) w_5 + (i_1 w_3 + i_2 w_4) w_6$$

$$\frac{\partial Error}{\partial W_6} = \frac{\frac{1}{2}(predictoin - actula)^2}{\partial prediciton} * \frac{\partial\left(i_1 w_1 + i_2 w_2\right) w_5 + \left(i_1 w_3 + i_2 w_4\right) w_6}{\partial W_6}$$

$$\frac{\partial Error}{\partial W_6} = 2 * \frac{1}{2}(predictoin - actula)\frac{\partial(predictoin - actula)}{\partial prediciton} * (i_1 w_3 + i_2 w_4)$$

$h_2 = i_1 w_3 + i_2 w_4$

$$\frac{\partial Error}{\partial W_6} = (predictoin - actula) * (h_2)$$

Δ = prediction – actual

delta

$$\frac{\partial Error}{\partial W_6} = \Delta h_2$$

Пример 1

Грешка: 0.5*(0.191-1)^2=0.327

$$^{*}W_6 = W_6 - a\left(\frac{\partial Error}{\partial W_6}\right)$$

2*0.11+3*0.21 = 0.85
2*0.12+3*0.08 = 0.48

0.85*0.14+0.48*0.15 = 0.191

Но реалната стойност е: 1

updated weights

$$^{*}w_6 = w_6 - a\,(h_2 \,.\, \Delta)$$
$$^{*}w_5 = w_5 - a\,(h_1 \,.\, \Delta)$$
$$^{*}w_4 = w_4 - a\,(i_2 \,.\, \Delta w_6)$$
$$^{*}w_3 = w_3 - a\,(i_1 \,.\, \Delta w_6)$$
$$^{*}w_2 = w_2 - a\,(i_2 \,.\, \Delta w_5)$$
$$^{*}w_1 = w_1 - a\,(i_1 \,.\, \Delta w_5)$$

Пример 1

Реалната стойност е: 1

Грешка: 0.5*(0.191-1)^2=0.327

Делта: 0.191 – 1 = -0.809
Нека a = 0.025

$$^{*}W_6 = W_6 - a\left(\frac{\partial Error}{\partial W_6}\right)$$

$$w_6 = 0.15 - 0.025(-0.809) * 0.48 = 0.159708$$

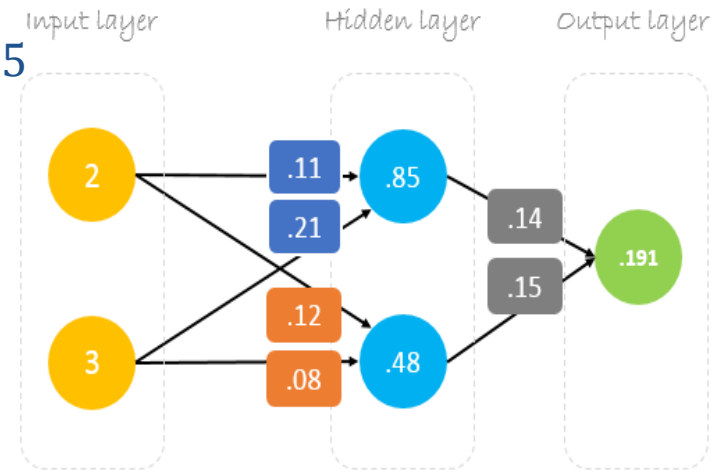
$$w_5 = 0.14 - 0.025(-0.809) * 0.85 = 0.15719125$$

$$w_4 = 0.08 - 0.025(-0.809) * 3 * 0.15 = 0.08910125$$

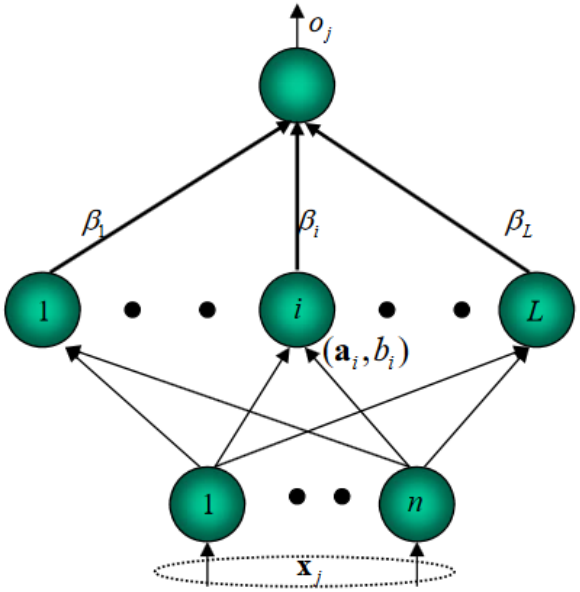
$$w_3 = 0.12 - 0.025(-0.809) * 2 * 0.15 = 0.1260675$$

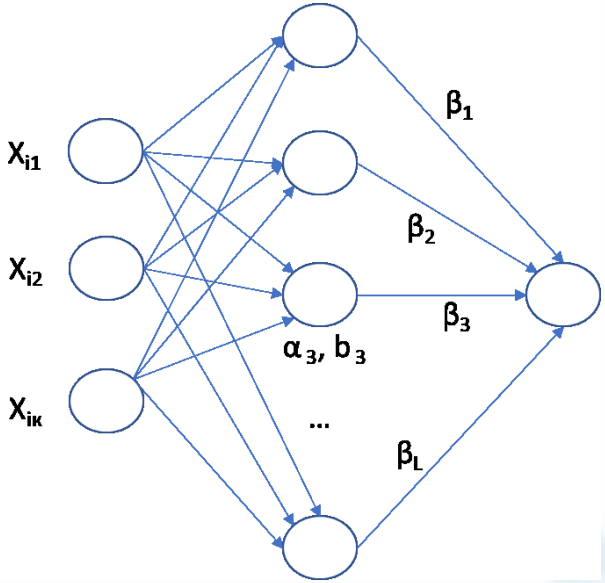
$$w_2 = 0.21 - 0.025(-0.809) * 3 * 0.14 = 0.2184945$$

$$w_1 = 0.11 - 0.025(-0.809) * 2 * 0.14 = 0.115663$$



ELM





$$o_i = \sum_{j=1}^{L} \beta_j g(x_i, \alpha_j, b_j)$$

$$\begin{bmatrix} o_1 \\ o_2 \\ \vdots \\ o_N \end{bmatrix} = \begin{bmatrix} g(x_1, \alpha_1, b_1) & \cdots & g(x_1, \alpha_L, b_L) \\ \vdots & \ddots & \vdots \\ g(x_N, \alpha_1, b_1) & \cdots & g(x_N, \alpha_L, b_L) \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_L \end{bmatrix}$$

$$O = G\beta$$

In fine…

БЛАГОДАРЯ ВИ ЗА ВНИМАНИЕТО!