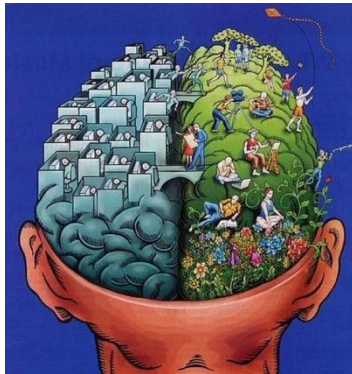




# Budapesti Műszaki és Gazdaságtudományi Egyetem

## Méréstechnika és Információs rendszerek Tanszék



## Neurális hálók

Előadó:

Hullám Gábor

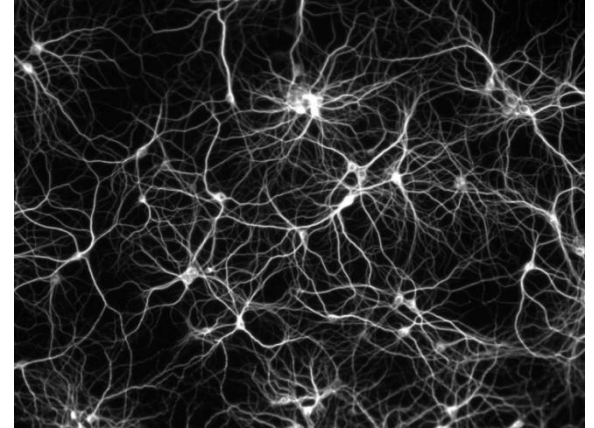
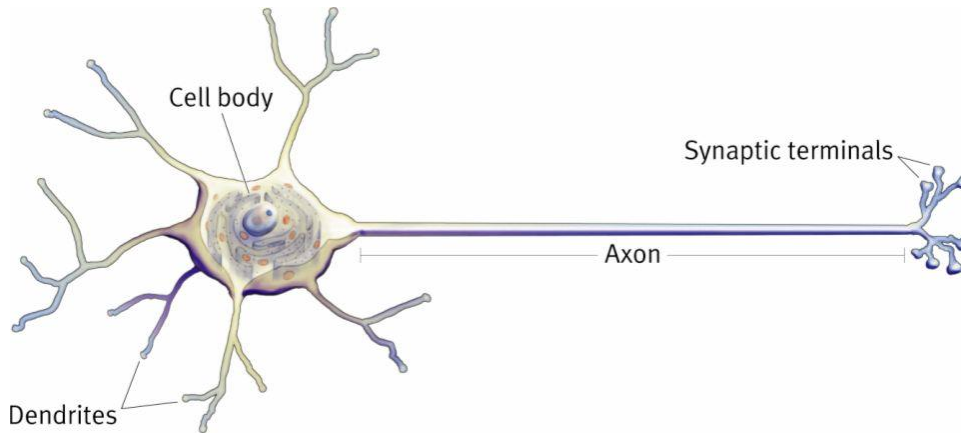
Pataki Béla

Előadás anyaga:

Dobrowiecki Tadeusz

# Neurális hálózat - neuronok

---



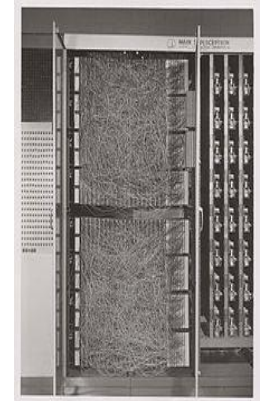
- Neuron doktrina: S. Ramón y Cajal (1852-1934)
- Mesterséges neuron: W. McCulloch and W. Pitts, 1943
- Tanulás: D. Hebb, 1949
- SNARC (Stochastic Neural Analog Reinforcement Calculator): M. Minsky, 1951
- Perceptron: F. Rosenblatt, 1957
  
- Neuronok becsült száma az emberi agyban: F. Azevedo et al., 2009  
86.1 +/- 8.1 milliárd



# Perceptron - történelem

---

- ▶ 1957 - Frank Rosenblatt: Perceptron
  - A perceptron algoritmus első implementációja a Mark I Perceptron gép
- ▶ 1969: Minsky és Papert: Perceptrons c. könyve
  - A perceptronnak súlyos korlátai vannak a képességeit tekintve
  - Hatására kb 10 évig nem történt kutatás a témakörben (AI Winter)



# Klasszikus neurális hálózatok

---

- ▶ 1980 - 1986 - David E. Rumelhart, Geoffrey E. Hinton and Ronald J. Williams: "Learning representations by back-propagating errors"
  - Hibavisszaterjesztés algoritmus
  - Újra beindította a kutatást a témakörben
- ▶ 1995 - Vapnik: Support-vector networks
  - Számos problémára sokkal jobb eredményt ad, kevésbé nehézkes a tanítása
  - Hatásásra a neurális hálózatok kutatása (ismét) alábbhagyott



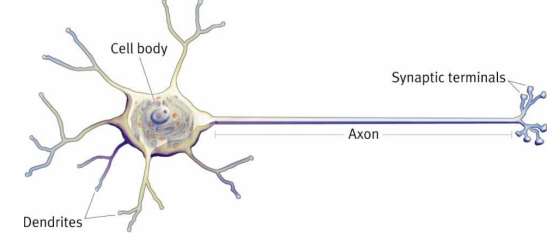
# Mély neurális hálózatok

---

- ▶ 2000-es évek közepén a neurális hálózatok témakörben alig lehetett cikket elfogadtatni
- ▶ 2004 – G. Hinton - CIFAR (Canadian Institute for Advanced Research)
  - ▶ Új csomagolás a neurális hálózatoknak: **deep learning**
- ▶ 2006 - Hinton, Osindero, Yee-Whye Teh: A fast learning algorithm for deep belief nets
  - Új áttörés: mély hiedelem hálók rétegenkénti tanítása,
  - Újra beindítja a kutatást

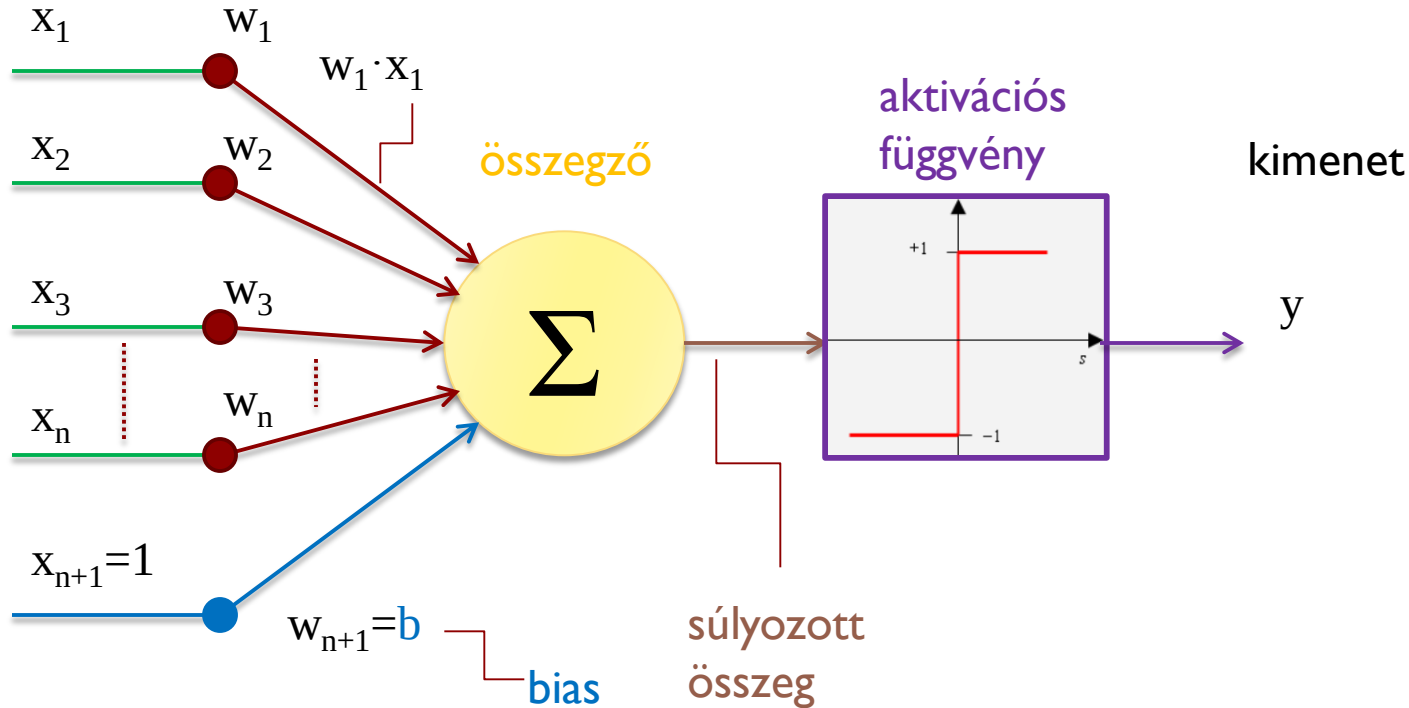


# Perceptron



bemenet

súlyok



$$y = \text{sgn}\left(\sum_{i=1}^{n+1} w_i x_i\right) = \text{sgn}\left(\sum_{i=1}^n w_i x_i + w_{n+1}\right) = \text{sgn}(\mathbf{x}^T \mathbf{w} + b)$$



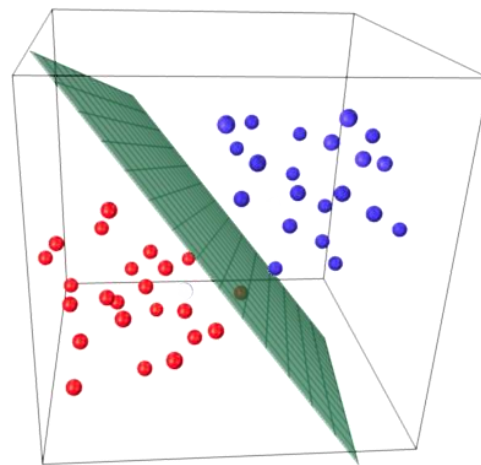
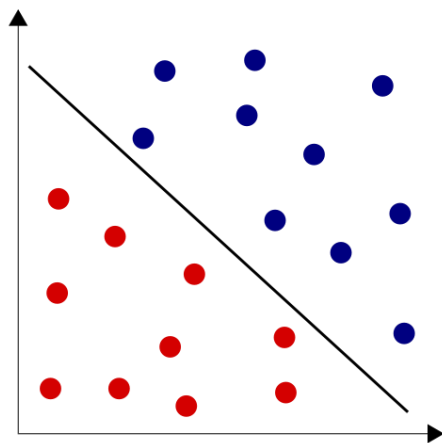
# A perceptron képessége

---

$$y = \text{sgn}\left(\sum_{i=1}^{n+1} w_i x_i\right) = \text{sgn}\left(\sum_{i=1}^n w_i x_i + w_{n+1}\right) = \text{sgn}(\mathbf{x}^T \mathbf{w} + b)$$

$$\mathbf{x}^T \mathbf{w} + b > 0$$

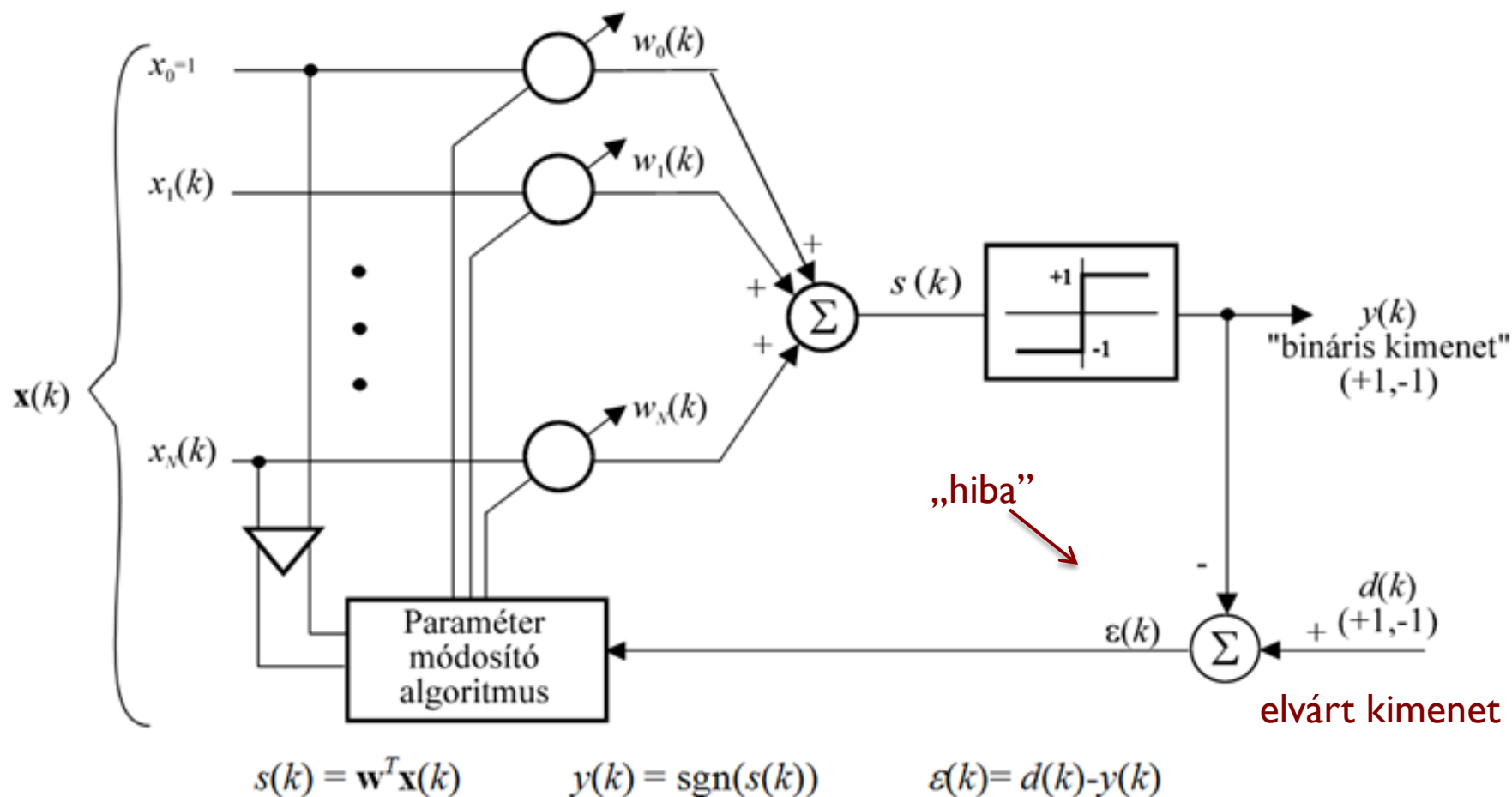
$$\mathbf{x}^T \mathbf{w} + b < 0$$



**lineárisan (hipersíkkal) szeparálható** függvényt képes reprezentálni



# Perceptron tanítása 1.



$$\mathbf{w}(k) = \mathbf{w}(k-1) + \alpha \varepsilon(k) \mathbf{x}(k)$$

$\alpha$  - **bátorsági faktor**,  
tanulási tényező



# Perceptron tanítása 2.

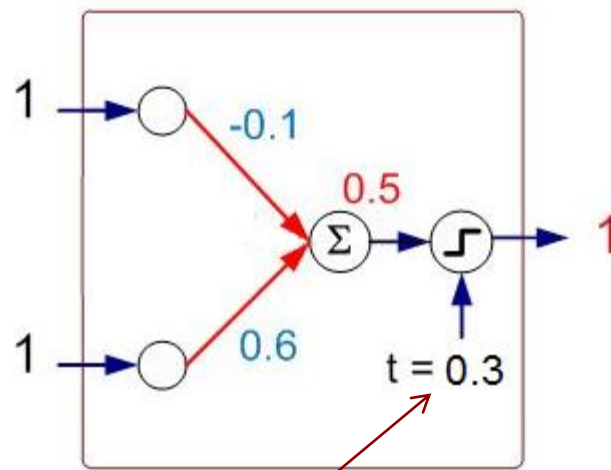
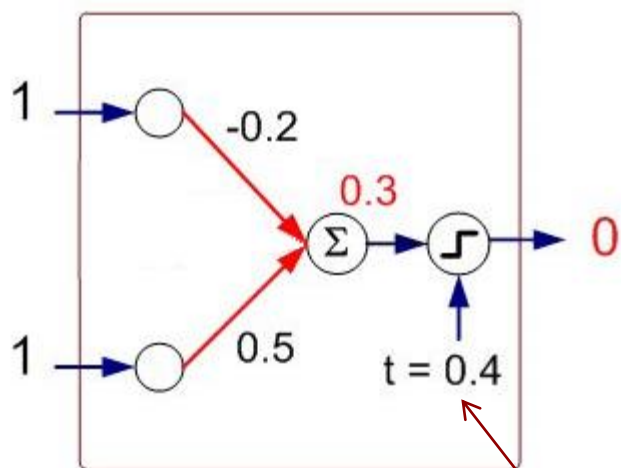
Tanítás konvergens, ha

- A tanító példák lineárisan szeparálhatók
- Bátorsági tényező elegendően kicsi

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \alpha \varepsilon(k) \mathbf{x}(k)$$

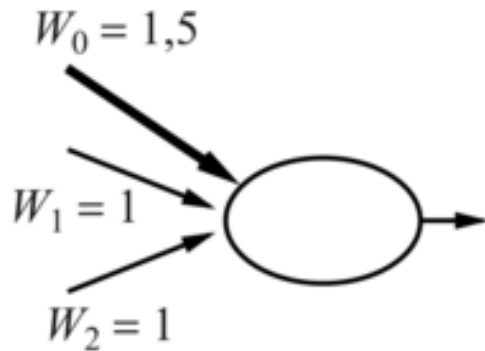
$$\eta = 0.1$$

| $X_1$ | $X_2$ | $Y$ |
|-------|-------|-----|
| 1     | 1     | 1   |

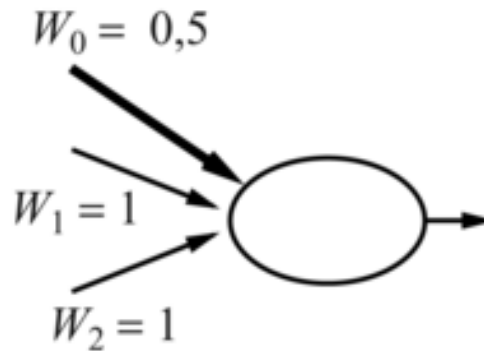


Küszöb érték

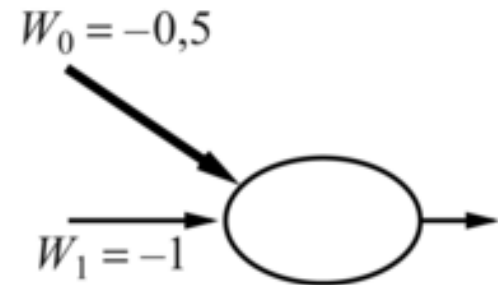
# Perceptron és a logikai függvények



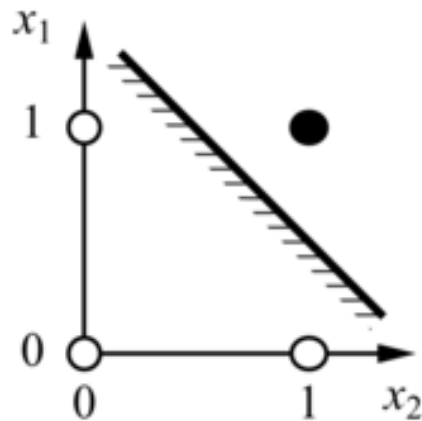
ÉS



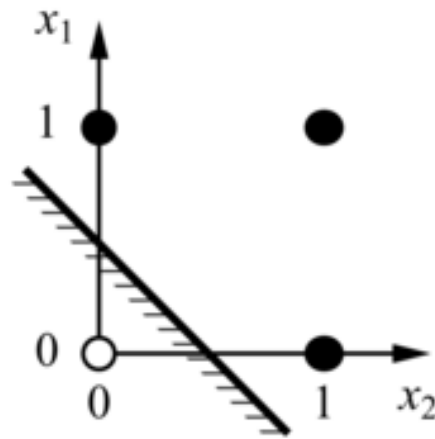
VAGY



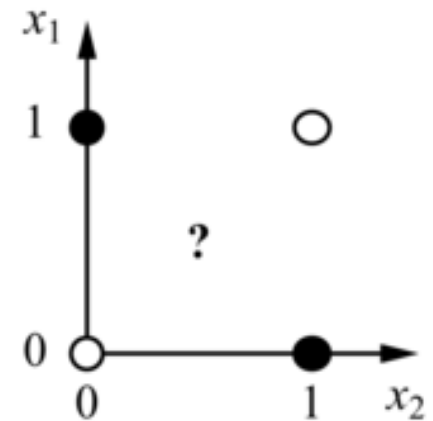
NEM



$x_1$  és  $x_2$



$x_1$  vagy  $x_2$



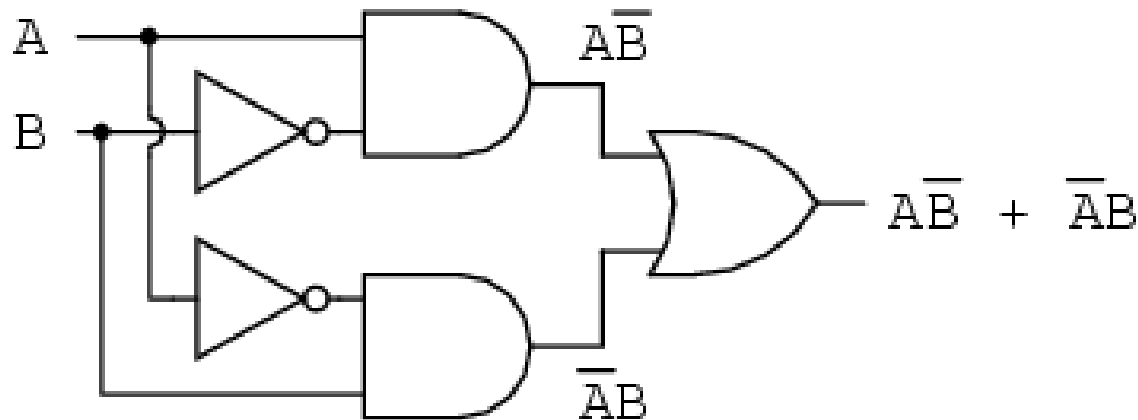
$x_1$  xor  $x_2$



# XOR (és hasonlóan lineárisan nem szeparálhatók) problémája

---

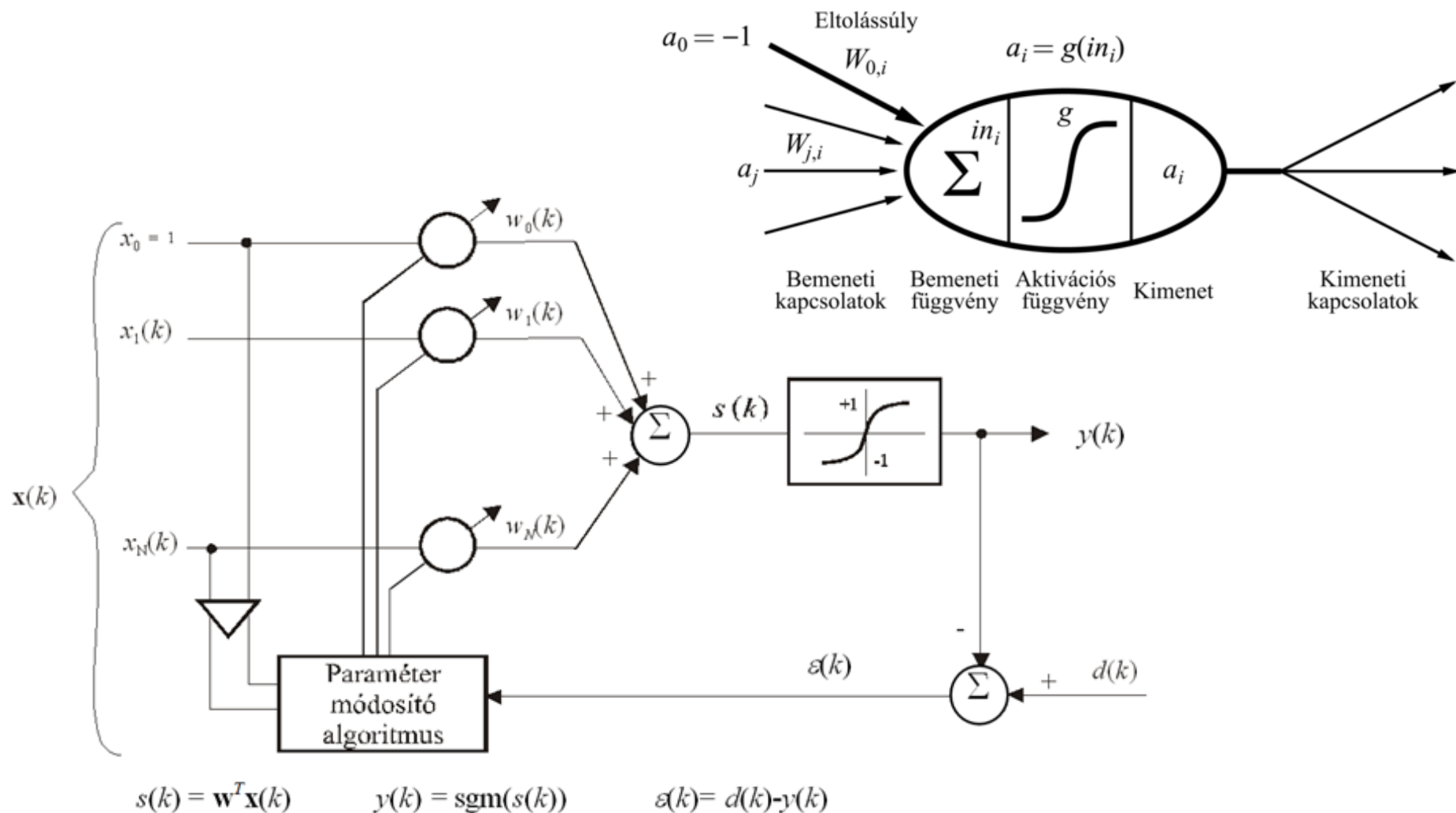
- több réteg kellene
- tanulás (preceptronnál) csak 1 rétegre
  - hiba értelmezése a kérdés



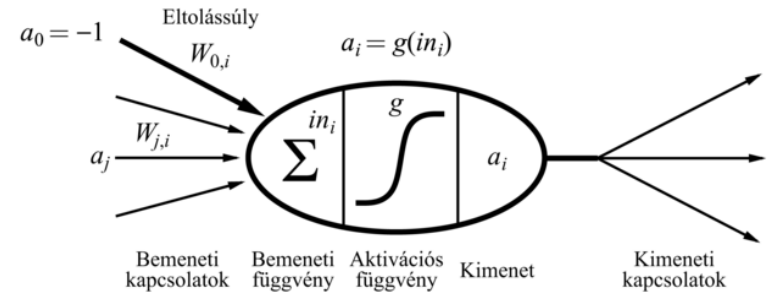
$$\mathbf{A \oplus B = A\bar{B} + \bar{A}B}$$



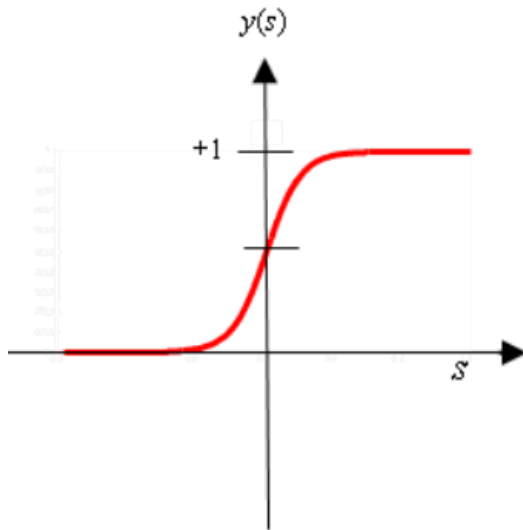
# Mesterséges neuron



# Mesterséges neuron – aktivációs függvények

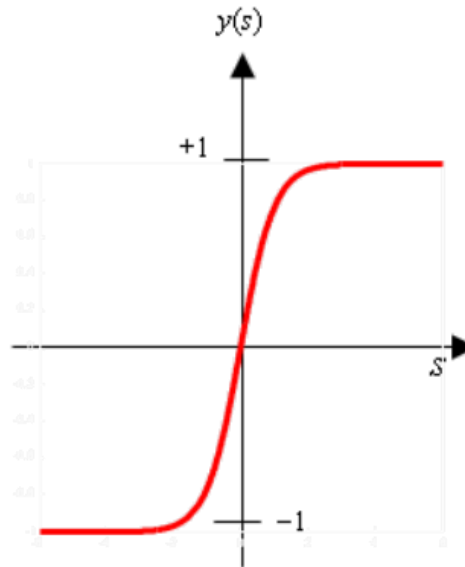


Sigmoid

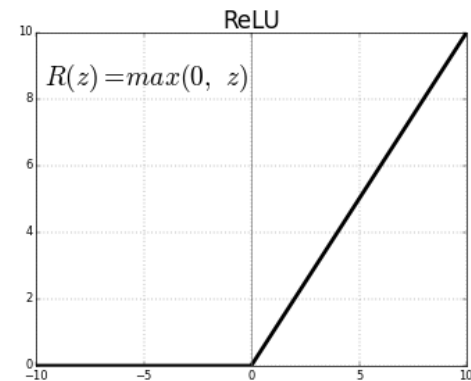


$$y = \frac{1}{1 + e^{-Ks}}; \quad K > 0$$

Tanh

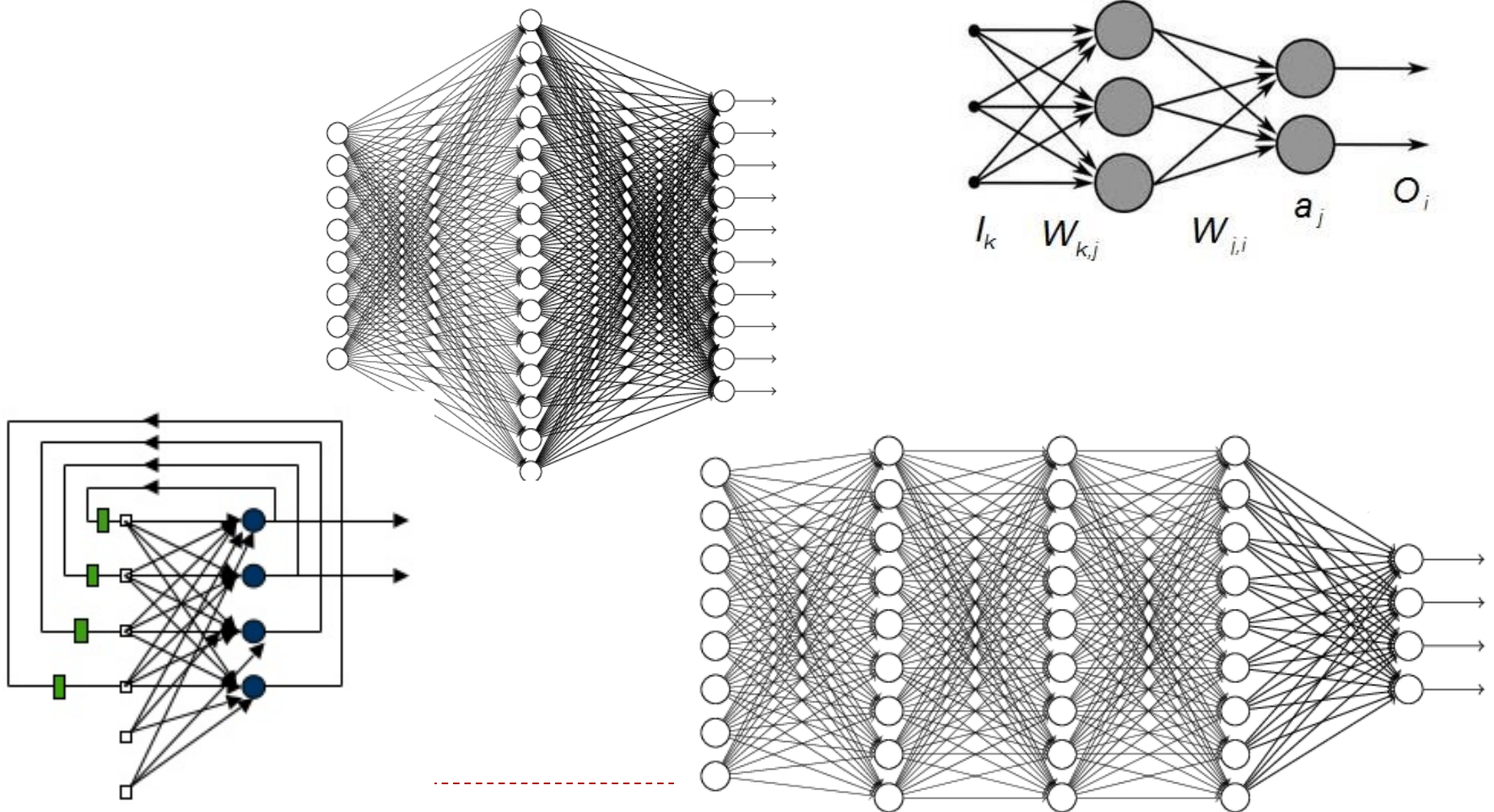


$$y = \frac{1 - e^{-Ks}}{1 + e^{-Ks}}; \quad K > 0$$



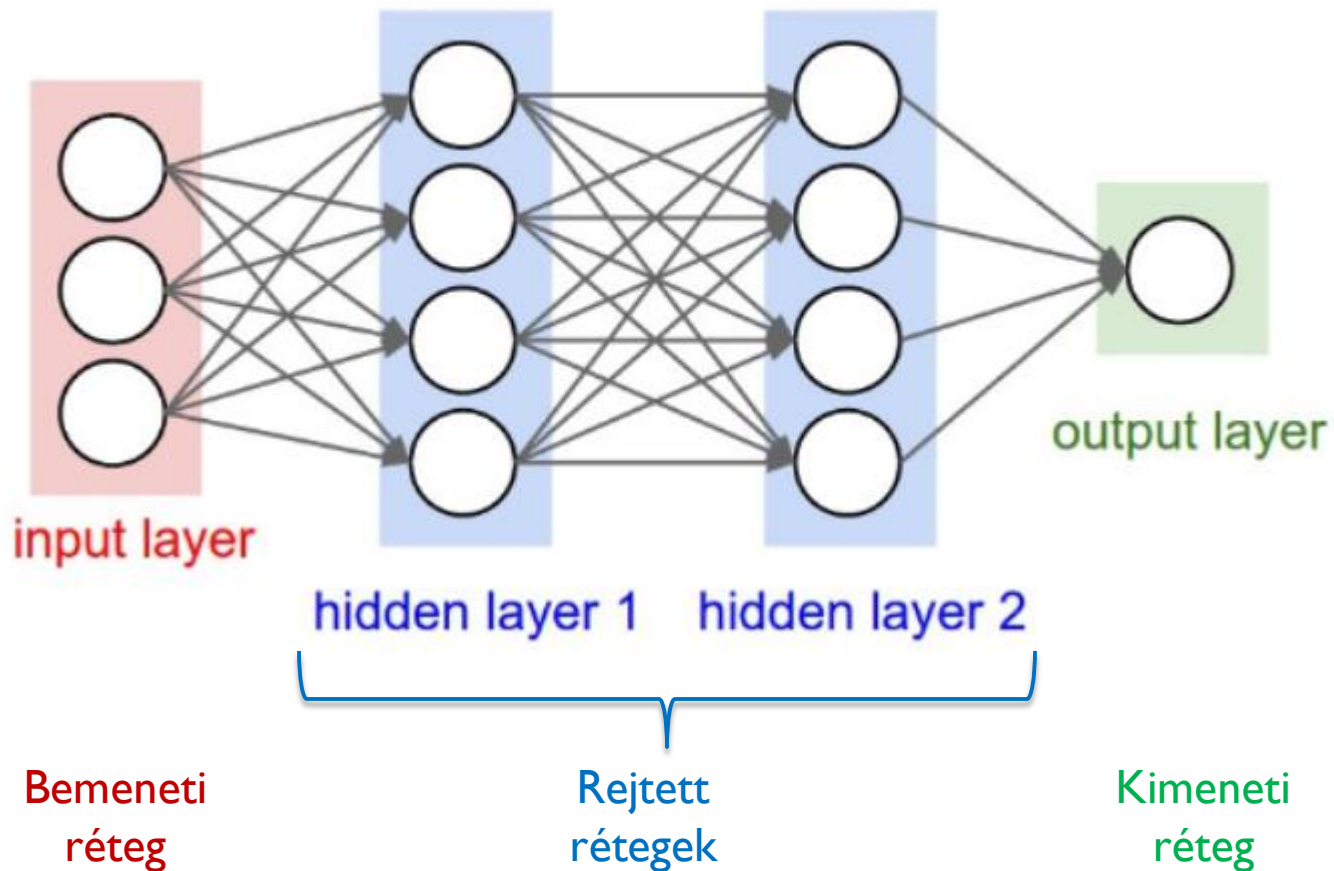
# Mesterséges Neurális Háló (Artificial Neural Network)

nemlineáris approximációt megvalósító, induktív tanulási algoritmussal tanítható matematikai struktúra.



# Mesterséges neurális háló felépítés

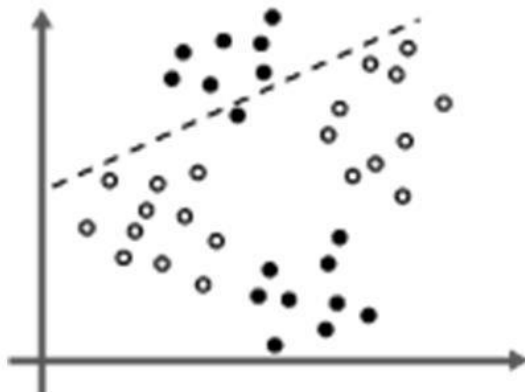
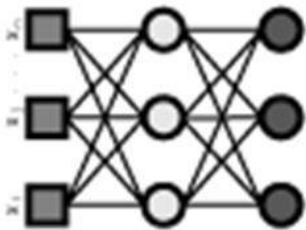
---



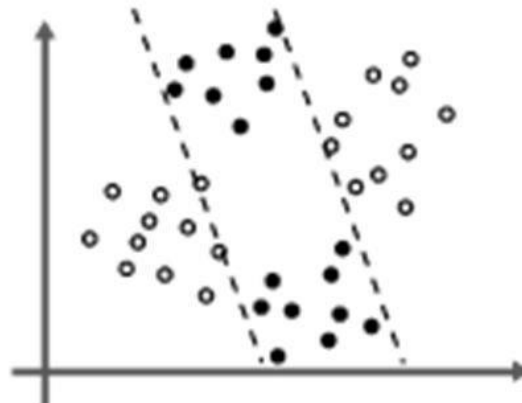
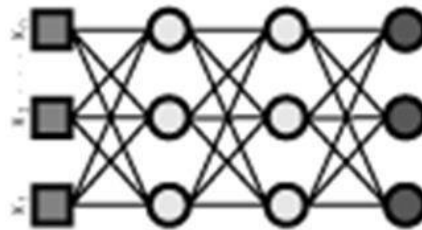
# Mesterséges neurális háló

---

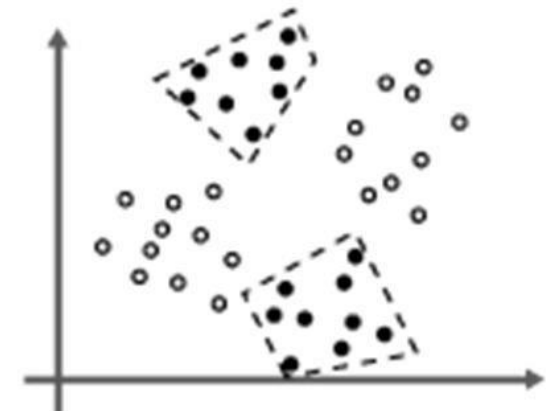
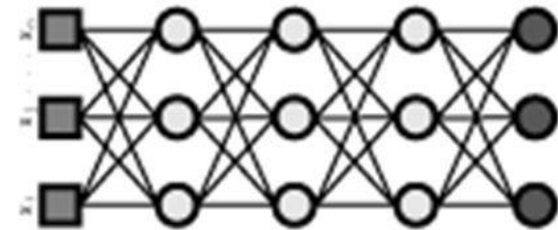
single layer



two layer



three layer





# Mesterséges neurális háló reprezentációs képessége

D. Hilbert (1900) 23 matematikai problémája/sejtése:

13. probléma: "Bizonyítsuk be, hogy az  $x^7+ax^3+bx^2+cx+1=0$  hetedfokú egyenlet nem oldható meg pusztán kétváltozós függvények segítségével!,,

...

"Mutassuk meg, hogy van olyan háromváltozós folytonos függvény, mely nem írható fel véges számú kétváltozós folytonos függvény segítségével!,,

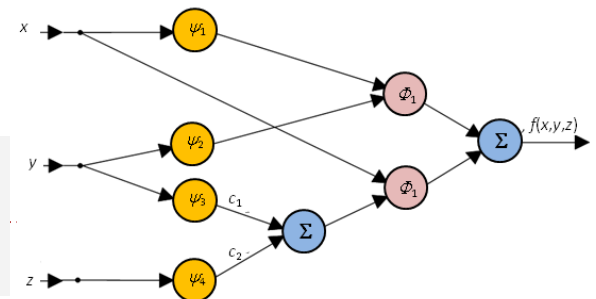
...

A. Kolmogorov (1957): nem csupán minden háromváltozós függvény, hanem tetszőleges  $N$ -változós folytonos függvény felírható csupán **egy** változós függvények és az összeadás segítségével.

...

<http://mialmanach.mit.bme.hu/neuralis/ch01s04>

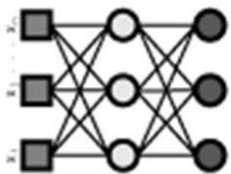
$$f(x_1, x_2, \dots, x_N) = \sum_{q=1}^{2N+1} \phi_q \left( \sum_{p=1}^N \psi_{pq}(x_p) \right)$$



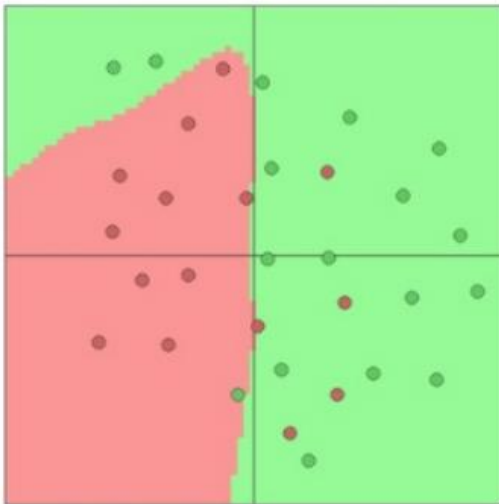
# Mesterséges neurális háló

- ▶ Már egy rejtett réteggel is univerzális approximátor (bármely függvényt képes reprezentálni)

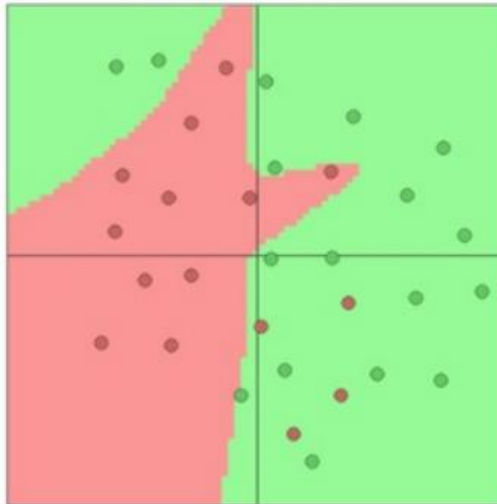
single layer



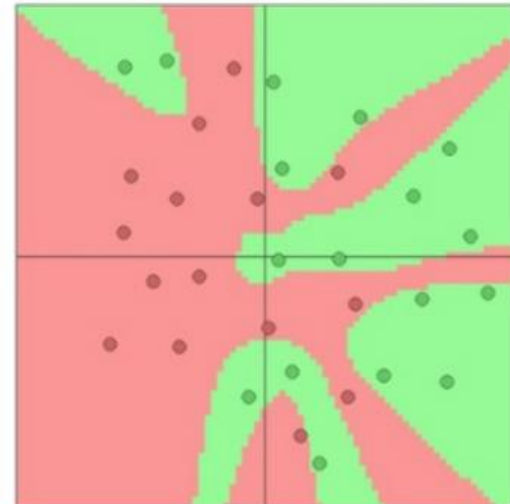
3 hidden neurons



6 hidden neurons



20 hidden neurons



# Mesterséges neurális háló tanítása

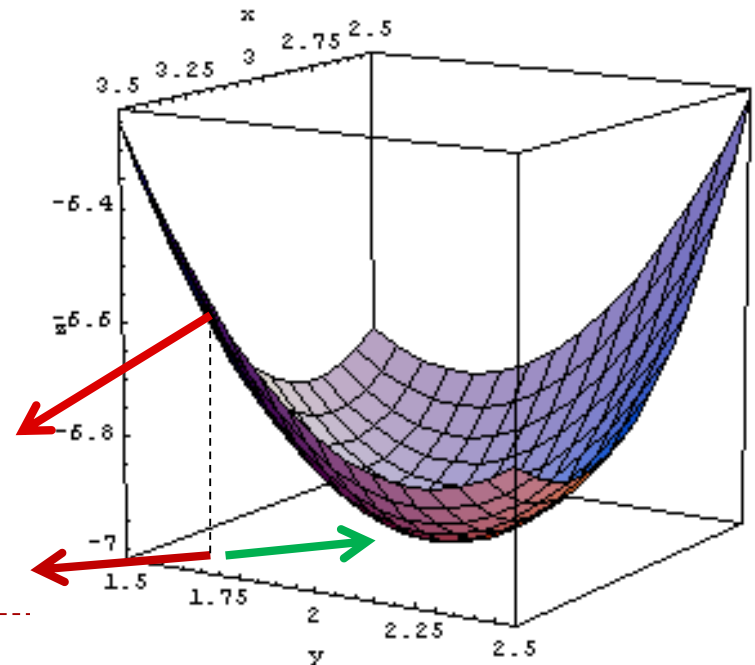
Többrétegű előrecsatolt háló tanítása (elemi alapok)

**Hibavisszaterjesztés gradiens módszerrel**

- példa bemeneteket mutatunk a hálónak,
- ha hiba lép fel (a kimenet és a kívánt érték eltér), a súlyokat úgy módosítjuk, hogy a hiba csökkenjen.

**A trükk a hiba megállapítása és a hibának a hibát létrehozó súlyok közti szétosztása.**

$$W \leftarrow W - \alpha \frac{\partial E}{\partial W}$$



# Hiba- / költség- / veszteség-függvény választás

---

- ▶ Négyzetes hiba – regresszió

$$E(\underline{d}, \underline{y}) = \frac{1}{2} \sum_i (d_i - y_i)^2$$

- ▶ Keresztentrópia – osztályozás

$$E(\underline{d}, \underline{y}) = -\sum_i d_i \log y_i$$

- ▶ A cél azon súlyok megtalálása, melyek mellett minimális a hiba

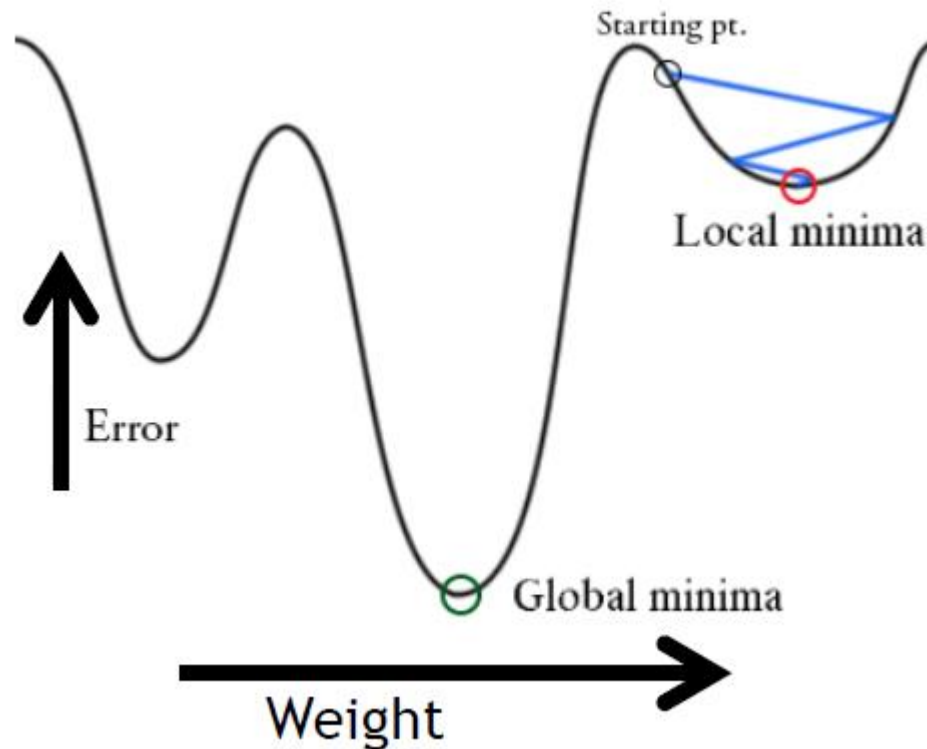
$$w^* = \arg \min_w \sum_{i=1}^N E(d_i, y_i) \quad , \text{ ahol } \underline{y} = f(\underline{x}, \underline{w})$$



# A tanítás nehézségei

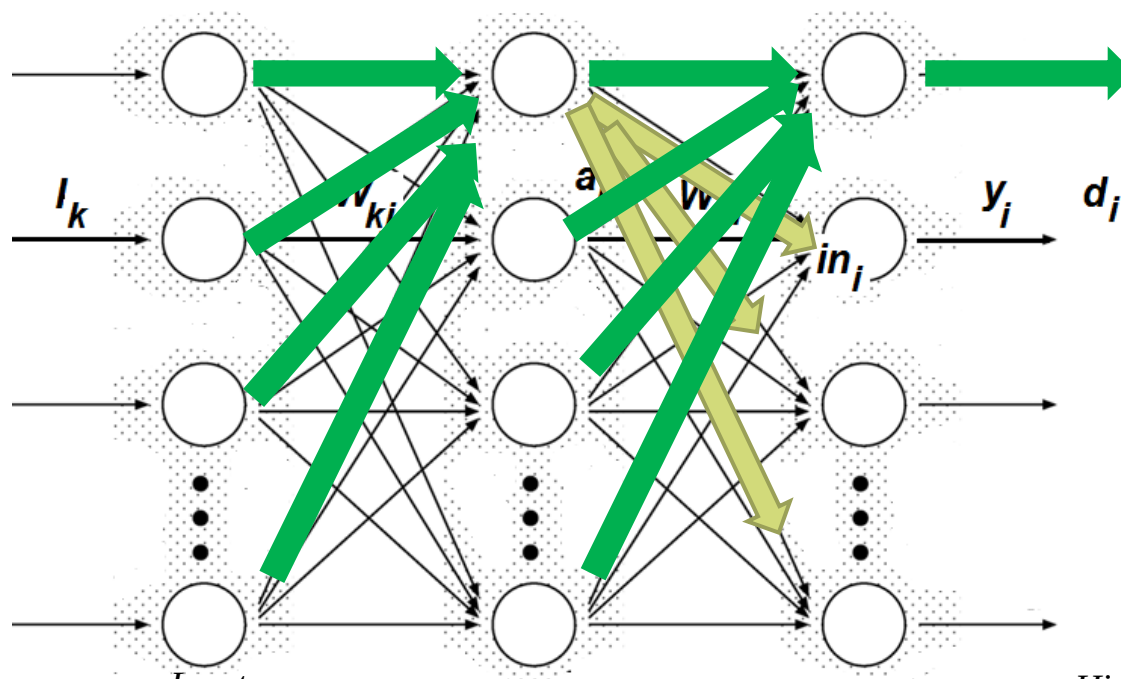
---

- Kihívás: a neurális hálók tanítása egy nem konvex optimalizálási feladat



# A tanítás lépései

- ▶ Kimenet számítása (forward pass)
- ▶ Hiba visszaterjesztése (backpropagation)

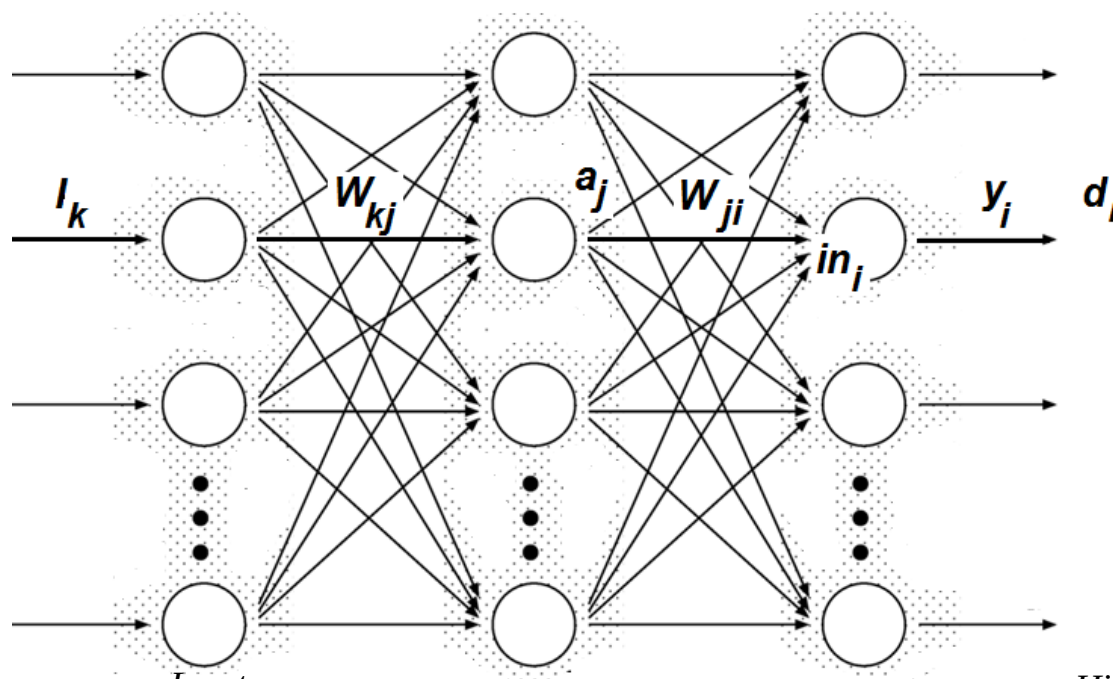


$$a_j(\underline{x}) = g(w_{k0} + \sum_{k=1}^{Input} x_k \cdot w_{kj})$$

$$y_i(\underline{x}) = g^*(w_{j0} + \sum_{j=1}^{Hidden} a_j(\underline{x}) \cdot w_{ji})$$

# A tanítás lépései

- ▶ Kimenet számítása (forward pass)
- ▶ Hiba visszaterjesztése (backpropagation)

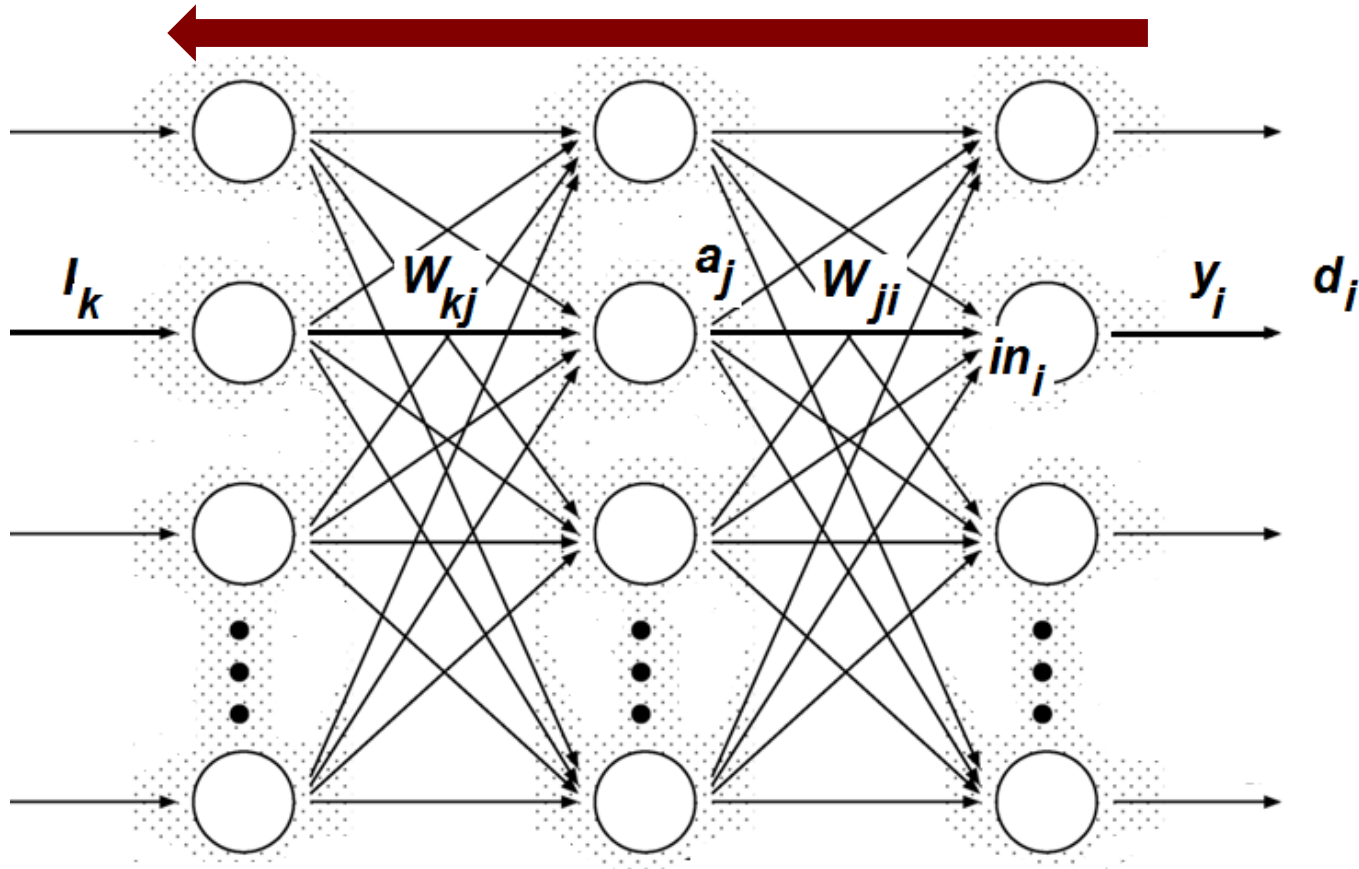


$$a_j(\underline{x}) = g(w_{k0} + \sum_{k=1}^{Input} x_k \cdot w_{kj})$$

$$y_i(\underline{x}) = g^*(w_{j0} + \sum_{j=1}^{Hidden} a_j(\underline{x}) \cdot w_{ji})$$



# Backpropagation



$$W_{k,j} \leftarrow W_{k,j} - \alpha \frac{\partial E}{\partial W_{k,j}} \quad W_{j,i} \leftarrow W_{j,i} - \alpha \frac{\partial E}{\partial W_{j,i}} \quad E = \frac{1}{2} \sum_i (d_i - y_i)^2$$



# Mesterséges Neurális Háló – hibavisszaterjesztés, alapok

$$E = \frac{1}{2} \sum_i (d_i - y_i)^2 = \frac{1}{2} \sum_i \text{Err}_i$$

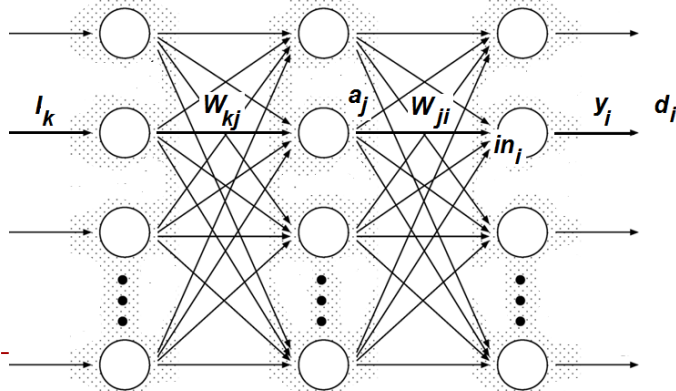
$$W_{j,i} \leftarrow W_{j,i} - \alpha \frac{\partial E}{\partial W_{j,i}}$$

$$E(\mathbf{W}) = \frac{1}{2} \sum_i (d_i - g(\sum_j W_{j,i} a_j))^2 = \frac{1}{2} \sum_i (d_i - g(\sum_j W_{j,i} g(\sum_k W_{k,j} I_k)))^2$$

$$\frac{\partial E}{\partial W_{j,i}} = -a_j (d_i - y_i) g'(\sum_j W_{j,i} a_j) = -a_j (d_i - y_i) g'(in_i) = -a_j \Delta_i$$

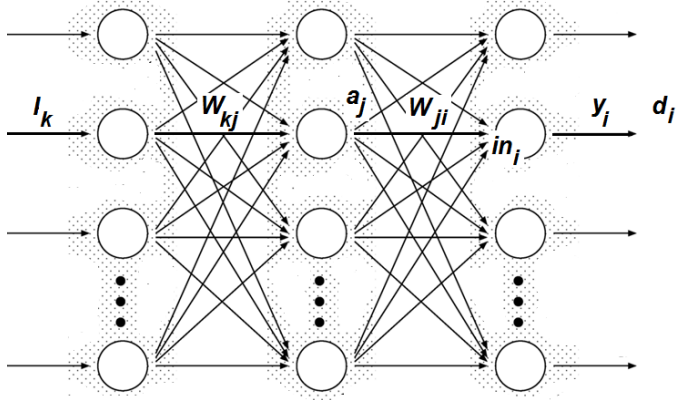
$$W_{j,i} \leftarrow W_{j,i} - \alpha \frac{\partial E}{\partial W_{j,i}} = W_{j,i} + \alpha a_j \text{Err}_i g'(in_i)$$

$$\Delta_i = \text{Err}_i g'(in_i)$$



$$W_{j,i} \leftarrow W_{j,i} + \alpha a_j \Delta_i$$

# Mesterséges Neurális Háló – hibavisszaterjesztés, alapok



$$\frac{\partial E}{\partial W_{j,i}} = -a_j (d_i - y_i) g'(\sum_j W_{j,i} a_j) = -a_j (d_i - y_i) g'(in_i) = -a_j \Delta_i$$

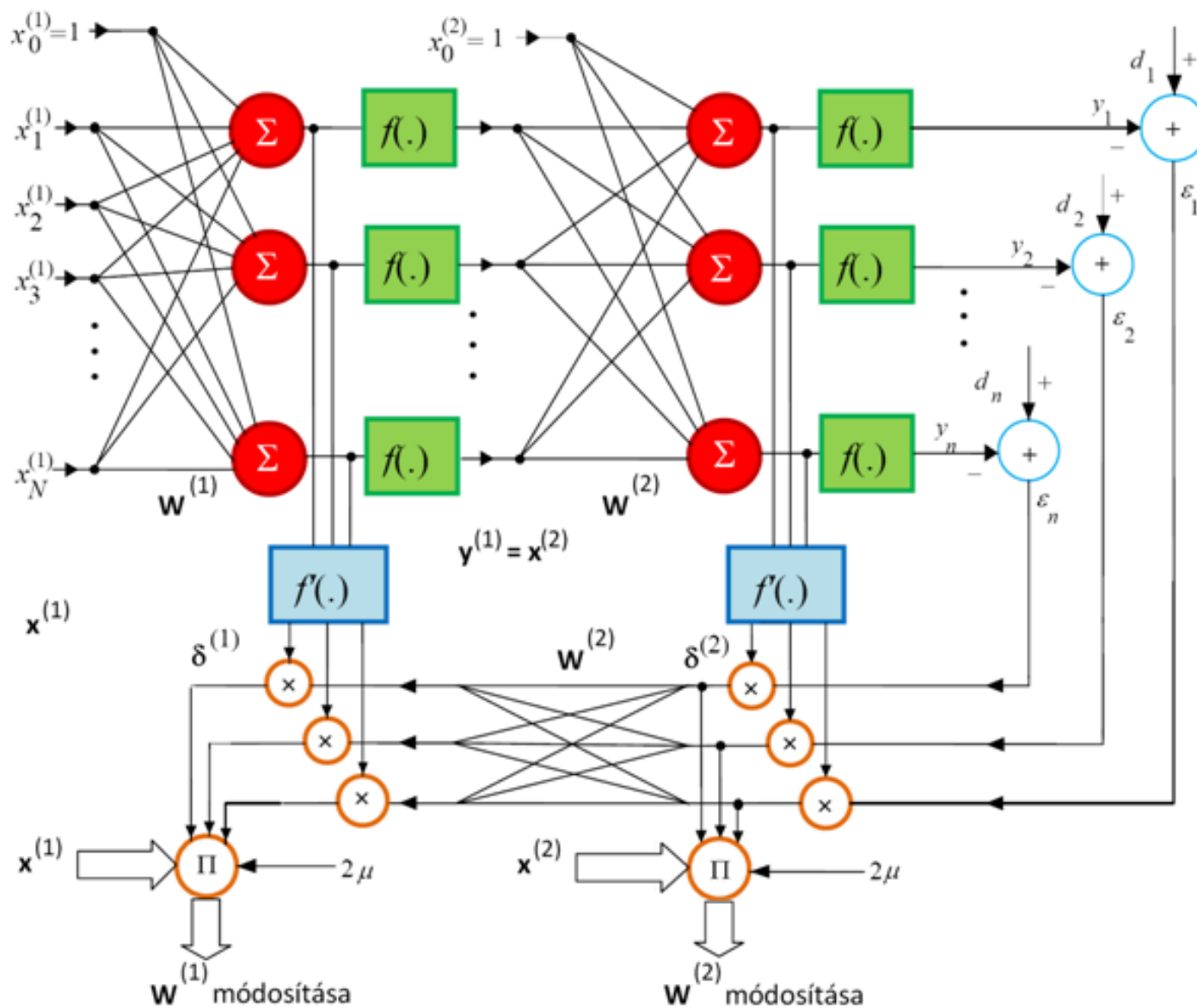
$$\frac{\partial E}{\partial W_{k,j}} = -I_k \Delta_j$$

$$W_{k,j} \leftarrow W_{k,j} - \alpha \frac{\partial E}{\partial W_{k,j}}$$

$$W_{k,j} \leftarrow W_{k,j} + \alpha I_k \Delta_j \quad \Delta_j = g'(in_j) \sum_i W_{j,i} \Delta_i$$



# Mesterséges Neurális Háló



# Neurális háló - tulajdonságok

---

## Aktivációs függvény

- A hibavisszaterjesztést alkalmazó (sekély) hálókbán általában a szigmoid függvényt vagy annak valamelyik változatát használjuk.
- A szigmoidnak megvan az a tulajdonsága, hogy deriváltja

$$g' = g(1 - g)$$

$$g(x) = \frac{1}{1 + e^{-x}}$$



# Neurális háló - tulajdonságok

---

## Kifejezőképesség

- Nem rendelkeznek az általános logikai reprezentációk kifejezőképességével.
- A többretegű hálók osztálya egészében, mint osztály az attribútumok bármely függvényének reprezentációjára képes.

## Számítási hatékonyság

- Legrosszabb esetben a szükséges epochok száma a bemenetek számának,  $n$ -nek, még exponenciális függvénye is lehet.
- A hibafelület lokális minimumai problémát jelenthetnek.



# Neurális háló - tulajdonságok

---

**Általánosító képesség:** jó általánosításra képesek.

**Zajérzékenység:** alapvetően nemlineáris regresszió, nagyon jól tolerálják a bemeneti adatok zajosságát.

**Átláthatóság:** alapvetően fekete doboz.

**A priori ismeret felhasználása:** nem könnyű  
(még aktívan kutatott terület)

