# Mesterséges Intelligencia

Csató Lehel

Matematika-Informatika Tanszék
Babeş–Bolyai Tudományegyetem, Kolozsvár

2006/2007

# Az Előadások Témái



- Bevezető: mi a *mesterséges* intelligencia ...
- „Tudás”–reprezentáció
- Gráfkeresési stratégiák
- Szemantikus hálók / Keretrendszerek
- Játékok modellezése
- Bizonytalanság kezelése
- Grafikus modellek
- Tanuló rendszerek
- Szimulált kifűtés, Genetikus algoritmusok
- Neurális hálók
- Gépi tanulás
- Nemparametrikus módszerek

# Adminisztra ... ... trívia

## Előadások honlapja

**http://www.cs.ubbcluj.ro/~csatol/mestint**

## Vizsga

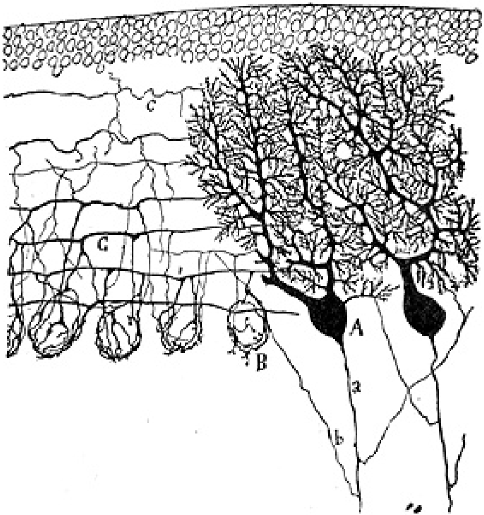
Szóbeli (60%) + Gyakorlat (40%)
(v) Előadás (60%)

## Laborgyakorlatok:

1. Clean vagy Prolog - dedukciós algoritmus **30%**
2. C / C++ / C# / · · · - genetikus algoritmus **10% vagy**
3. Matlab - Neurális hálózatok vagy SVM **10%**

**Ramon y Cajal**

- 1894 — —1900 időszakban;
- idegrendszer vizsgálata;
- építőegységek azonosítása: neuron;
- idegsejt – mint az idegrendszer építőeleme



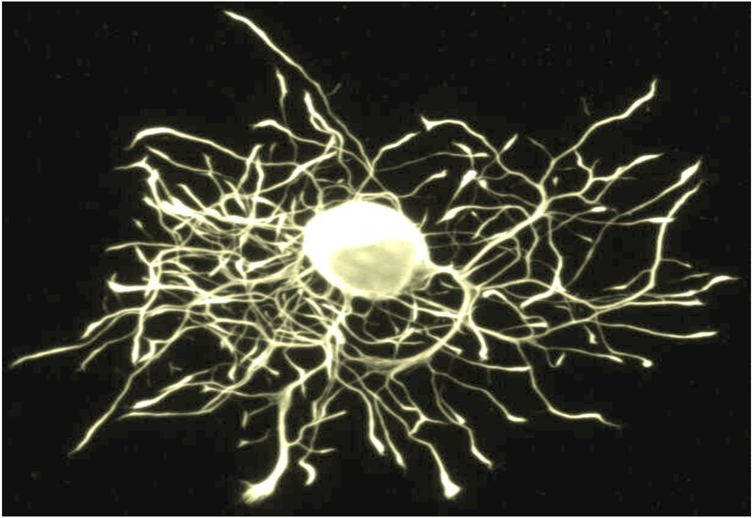

**Ramon y Cajal**
*Nobel díj 1906*

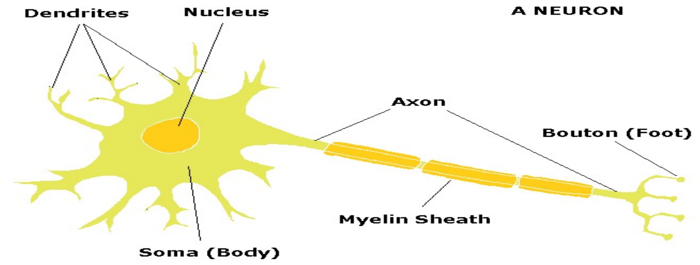
*The cerebellar cortex (a kitten cerebellum).*

*The letter A marks the Purkinje cells with dendritic ramifications.*

http://nobelprize.org/medicine/articles/cajal/

**Idegsejt fényképe:**





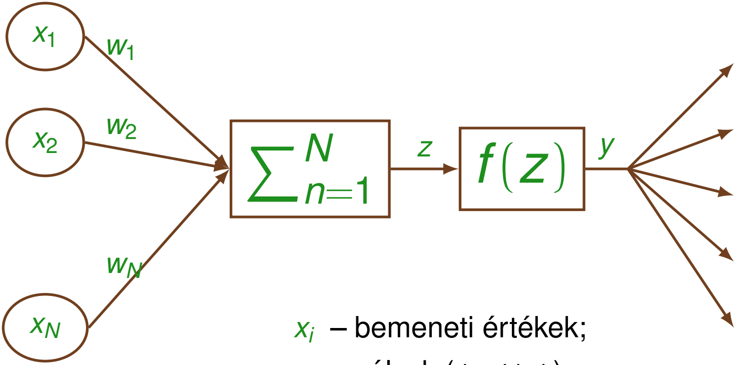

## Neuron alkotóelemei

- Szóma (sejtmag) – az információ feldolgozása;
- Dendrit – az információ összegyűjtése;
- Axon – az információ terjesztése;

*mi az információ?*

http://www.ship.edu/~cgboeree http://en.wikipedia.org/wiki/Neuron

## „Mesterséges” Neuron




Ahol:

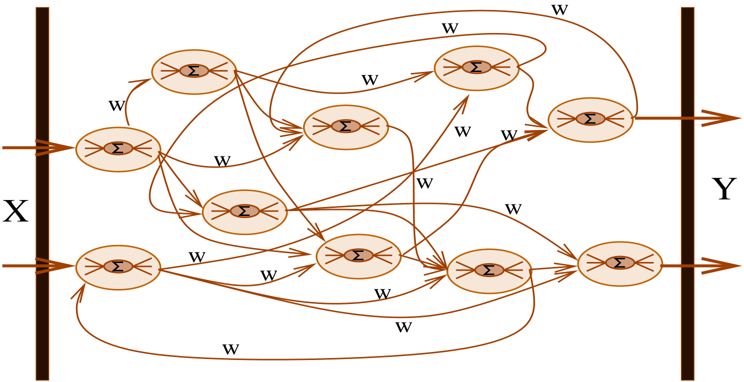
$x_i$ – bemeneti értékek;

$w_i$ – súlyok (dendritek);

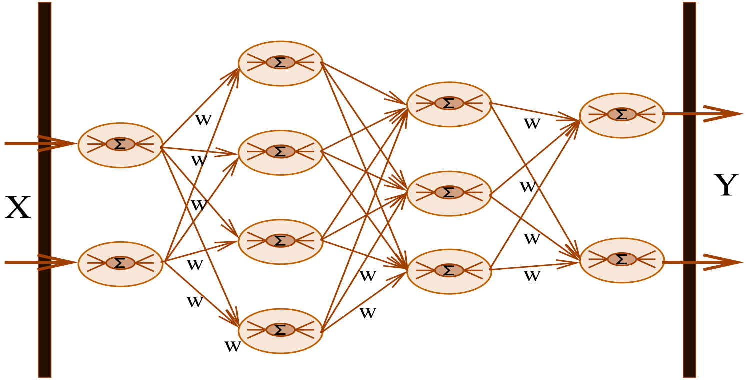
$\sum \cdots$ – összegző (szóma);

$f(z)$ – átalakító;

$\longrightarrow$ – kapcsolatok (axonok);




- aszinkron működés;
- nagyon sok kapcsolat;
- ?hogyan? kreáljunk/töröljünk kapcsolatokat;




- szinkron – ciklusos – működés;
- rétegek $\Rightarrow$ korlátolt számú kapcsolat;

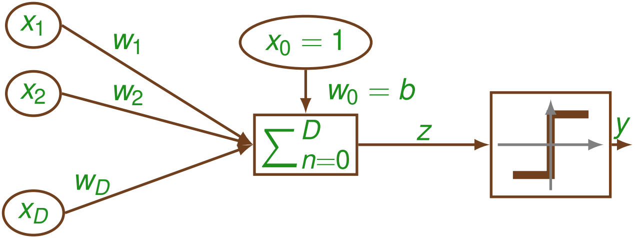
- Idegrendszer = információ-feldolgozó egység. Kérdések:
  - Mi az *információ*,
  - Hogyan történik a feldolgozás.

- biológiai neuronok aszinkron-működésűek: miért jó a mesterséges neuron szinkronizált jellege.

- mesterséges neuronok egymáshoz kapcsolódása csekély: a biológiai rendszerek nagyságrendekkel több kapcsolatot kezelnek.

# Perceptron

## Egyszerűsített természetes neuron-modell:




- Korai neuron-modell: McCullogh-Pitts $\sim$ 1958;
- $x_i$ és $y$ értékek binárisak;
- Aktivációs függvény a lépcsőfüggvény:

$$H(z) = \begin{cases} -1 & \text{ha } z < 0 \\ +1 & \text{ha } z \geq 0 \end{cases}$$

## Perceptron

- ON/OFF neuron állapotok
  *A neuronok vagy aktívak vagy nem, mint a bináris logikában*
  *rezolució-szerű működés*

- Tanulási szabály: ha $(\mathbf{x}^{(n)}, \mathbf{y}^{(n)})$-en hiba történt, akkor

$$w_i(t+1) \Leftarrow w_i(t) + x_i^{(n)} y^{(n)}$$

  ahol $w_{ij}$ súly, $\mathbf{x}^{(n)}$ bemeneti minta, $y^{(n)}$ a minta osztálya.

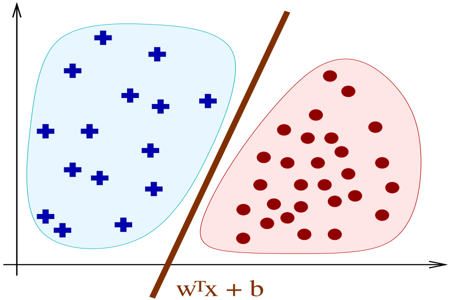
- Konvergencia-tétel: Ha a
  - tanulási minták $\{(\mathbf{x}^{(n)}, \mathbf{y}^{(n)})\}_{n=1}^{N}$ (osztályozási feladat)
  - elválaszthatóak, def

  akkor a perceptron konvergál.

# Elválaszthatóság I

kék osztály kódja legyen $+1$;

piros osztály kódja legyen $-1$;

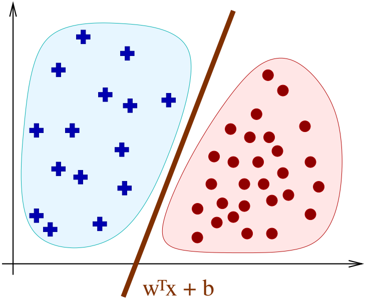



**Szeparáló** hipersík

a $\{\boldsymbol{w}, b\}$ vektorral illetve valós szám által meghatározott felület, ha

$$\begin{aligned} \boldsymbol{w}^T\boldsymbol{x}_i + b &> 0 \qquad \forall i \quad ; \quad y_i > 0. \\ \boldsymbol{w}^T\boldsymbol{x}_i + b &< 0 \qquad \forall i \quad ; \quad y_i < 0. \end{aligned}$$

# Elválaszthatóság II




**Szeparáló** hipersík:

$$y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) > 0 \qquad \forall i = 1, N.$$

ahol $\mathbf{x}_i$ - $i$-edik mintavektor és $y_i$ az ahhoz tartozó osztály kódja.

vissza

**Konvergenciatétel**

A tanulási szabály alkalmazásával a perceptron **véges** idő alatt konvergál.

**Biz:**
*Módszer:* találjunk – felső és alsó – korlátot a lépések számának.

1. Újradefiniáljuk a bemeneti adatokat:

$$\hat{\mathbf{x}}_i \stackrel{\text{def}}{=} [\mathbf{x}_i, 1]^T \in \mathbb{R}^{d+1} \qquad \Rightarrow \qquad \mathbf{w} \stackrel{\text{def}}{=} [\mathbf{w}, b]^T \in \mathbb{R}^{d+1}$$

$$\mathbf{x}_i \stackrel{\text{def}}{=} y_i \hat{\mathbf{x}}_i$$

# Konvergenciatétel II

2 **Újradefiniált feladat**
Találjunk **egy (1)** $\boldsymbol{w} \in \mathbb{R}^{d+1}$ értéket úgy, hogy

$$\forall i \;\; ; \;\; \boldsymbol{w}^T \boldsymbol{x}_i > 0$$

minden $\boldsymbol{x}_i$-re (már tartalmazzák a címkéket is).

3 Tanítási szabály
Ha hiba történt, akkor módosítjuk a perceptron súlyait:

$$w_i(t+1) \Leftarrow w_i(t) + x_i^{(n)}$$

(rekurzívan visszafejtve: $\boldsymbol{w}(t+1) = \boldsymbol{x}^1 + \cdots + \boldsymbol{x}^t$)

4 Hiba $\quad \Leftrightarrow \quad \boldsymbol{w}^T(t)\boldsymbol{x}^{(n)} < 0$.

(1) Amennyiben van egy érték, akkor nagyon sok elfogadható érték létezik.

5 Feltételezzük, hogy létezik szeparáló hipersík:

$$\exists\, \mathbf{w}_0 \in \mathbb{R}^{d+1} \forall i \quad ; \quad \mathbf{w}_0^T \mathbf{x}^{(n)} \geq \alpha > 0$$

6 Vizsgáljuk a $\mathbf{w}_0^T \mathbf{w}(t+1)$ skaláris szorzatot:

$$\mathbf{w}_0^T \mathbf{w}(t+1) = \mathbf{w}_0^T \mathbf{x}^1 + \cdots + \mathbf{w}_0^T \mathbf{x}^t \geq t\alpha$$

7 Cauchy-Schwartz egyenlőtlenség $\|a\|^2 \|b\|^2 \geq \left(a^T b\right)^2$ alkalmazzuk:

$$\|\mathbf{w}_0\|^2 \|\mathbf{w}(t+1)\|^2 \geq \left(\mathbf{w}_0^T \mathbf{w}(t)\right)^2 \geq (t\alpha)^2$$

vagyis

$$\|\mathbf{w}(t+1)\|^2 \geq \frac{(t\alpha)^2}{\|\mathbf{w}_0\|^2}$$

Másrészt - minden tanuló lépésnél:

8 $\boldsymbol{w}(t+1) = \boldsymbol{w}(t) + \boldsymbol{x}^{(t)}$, tehát:

$$\begin{aligned}
\|\boldsymbol{w}(t+1)\|^2 &= \|\boldsymbol{w}(t) + \boldsymbol{x}^{(t)}\|^2 \\
&= \|\boldsymbol{w}(t)\|^2 + 2\boldsymbol{w}(t)\boldsymbol{x}^{(t)} + \|\boldsymbol{x}^{(t)}\|^2 \\
&\leq \|\boldsymbol{w}(t)\|^2 + \|\boldsymbol{x}^{(t)}\|^2 = t\beta^2
\end{aligned}$$

Ahol $\beta^2 = \max\left\{\|\boldsymbol{x}^{(n)}\|^2 \;\; ; \;\; n = 1..N\right\}$

9 Van tehát két egyenlőtlenség, melyből kiszámítható a $t$ maximális értéke:

$$t_{max} x \leq \frac{\|\boldsymbol{w}_0\|^2 \beta^2}{\alpha^2}$$

**Tehát a perceptron konvergál** $\square$

**Kódoljuk** a perceptron algoritmust:

- generáljuk az adatokat. Specifikáljuk:
  - az adatok dimenzióját, számát;
  - az elválasztó hipersíkot;
  - generáljuk a bemeneti mintákat illetve az azokhoz tartozó kimeneti osztálykódokat.
- Transzformálunk a konvergenciatétel szerint.
- Inicializáljuk a perceptront az üres értékekkel.
- Futtatjuk az algoritmust:
  - mérjük a hibajavító lépéseket;
  - mérjük a ciklusok számát.

# Matlab Demó II

```
% ADATOK SPECIFIKALASA
dim = 9;
N   = 200;
% SZEPARALO HIPERSIK
w_0 = [1;-3; 4;-1; 2;-6;-5; 5; 3];
b   =  2;

% ADATOK GENERALASA
X = sign(rand(N,dim) - 0.5);
% CIMKEK
Y = sign(X*w_0 + b);
% NULLA CIMKEK KISZURESE
idx = find(~Y);
idx = setdiff([1:N],idx);
X = X(idx,:); Y=Y(idx);
N = length(Y);
```

# Matlab Demó III

```
%%%%%%  Perceptron tanitas %%%%%%
% 1 adatok transzformalasa
X = [ X ones(N,1)];
% 2 adatok szorzasa a cimkekkel
XX= diag(Y) * X;

%% tanitas
w = zeros(dim+1,1);
Term = 0;
lepes= 0; ciklus= 0;
```

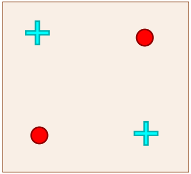
```
while ~Term;
  Term = 1;

  for ii=1:N;
    if XX(ii,:)*w <= 0
      w = w + XX(ii,:)';
      lepes = lepes + 1;
      fprintf('%d:',lepes);
      fprintf('%4d',w);fprintf('\n');
      Term = 0;
    end;
  end;
  ciklus = ciklus +1;
end;
```

Kiíratjuk a statisztikákat:

```
fprintf('\nCiklusok szama: %d\nEredmény:',ciklus);
fprintf('%4d',w); fprintf('\n');
fprintf('Eredeti :');
fprintf('%4d',[w_0;b]); fprintf('\n');
```

# Összefoglaló



- ’60-as évek tanulási paradigmája;
- logikai függvényeket keres – bemenet és kimenet binárisak;
- cél a gondolkodás modellezése
- Nem tudja szeparálni az *XOR* művelet kimeneteit (Minsky és Papert ’62)

**Feladat:**
találjunk konfigurációt, mely az *XOR*-t helyesen osztályozza.