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

Előadások honlapja

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

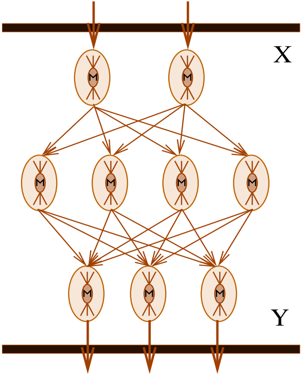
Vizsga

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%**

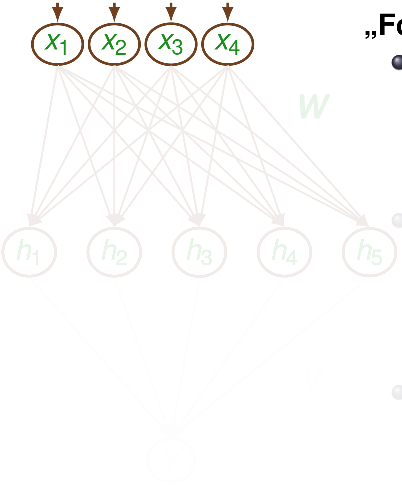
- Az információ – számok – a bemeneti X rétegből a kimeneti Y réteg fele halad.
- Egy mintára kiszámítjuk a kimeneti értéket, minden neuron értékét kiszámítva.
- Feltételezzük, hogy minden – adat és súly – folytonos.



MLP NEM Bayes-háló

A neurális hálók kapcsolatai kommunikációt, a Bayes-hálók kapcsolatai függőséget jelentenek.

Többrétegű Hálók II



„Formálisan”:

- A neuronháló csúcsai az információt feldolgozzák.

$$y_{\text{ki}} \stackrel{\text{def}}{=} f_{\text{akt}}(\boldsymbol{x}_{\text{be}})$$

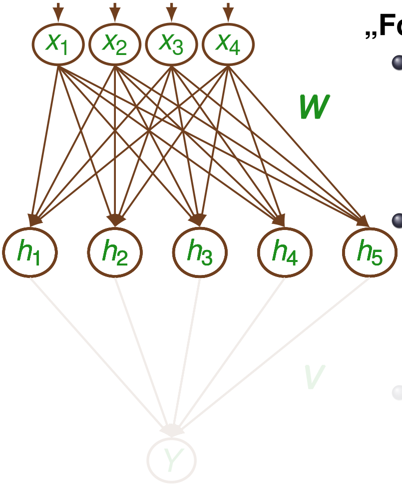
- Az élek: a csúcsok közötti kapcsolatok.

$$x_{\text{be}}^{(i)} \stackrel{\text{def}}{=} \sum_j w_{ij} x_{\text{ki}}^{(j)}$$

- A fenti lépések ismétlődnek.

MI az információ?

Többrétegű Hálók II



„Formálisan”:

- A neuronháló csúcsai az információt feldolgozzák.

$$y_{\mathrm{ki}} \stackrel{\mathrm{def}}{=} f_{\mathrm{akt}}(\boldsymbol{x}_{\mathrm{be}})$$

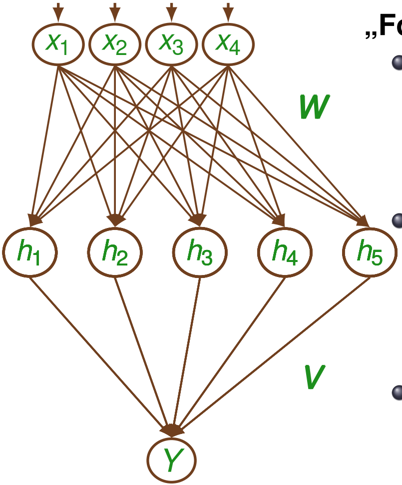
- Az élek: a csúcsok közötti kapcsolatok.

$$x_{\mathrm{be}}^{(i)\,\prime} \stackrel{\mathrm{def}}{=} \sum_j w_{ij} x_{\mathrm{ki}}^{(j)}$$

- A fenti lépések ismétlődnek.

MI az információ?

Többrétegű Hálók II



„Formálisan”:

- A neuronháló csúcsai az információt feldolgozzák.

$$y_{\text{ki}} \stackrel{\text{def}}{=} f_{\text{akt}}(\mathbf{x}_{\text{be}})$$

- Az élek: a csúcsok közötti kapcsolatok.

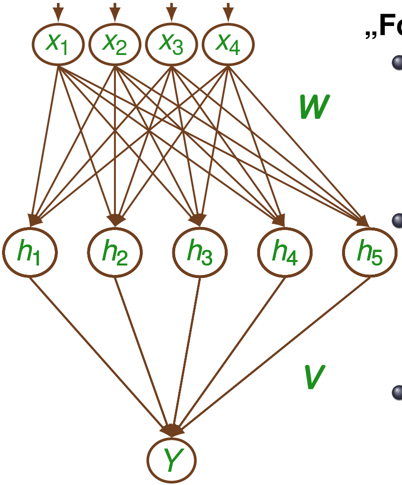
$$x_{\text{be}}^{(i)\,\prime} \stackrel{\text{def}}{=} \sum_j w_{ij} x_{\text{ki}}^{(j)}$$

- A fenti lépések ismétlődnek.

MI az információ?

definíció szerint $\mathbf{x}_{\text{ki}} \in \mathbb{R}^k$

Többrétegű Hálók II



„Formálisan”:

- A neuronháló csúcsai az információt feldolgozzák.

$$y_{\mathrm{ki}} \stackrel{\mathrm{def}}{=} f_{\mathrm{akt}}\left(\boldsymbol{x}_{\mathrm{be}}\right)$$

- Az élek: a csúcsok közötti kapcsolatok.

$$x_{\mathrm{be}}^{(i)\,\prime} \stackrel{\mathrm{def}}{=} \sum_j w_{ij} x_{\mathrm{ki}}^{(j)}$$

- A fenti lépések ismétlődnek.

MI az információ?

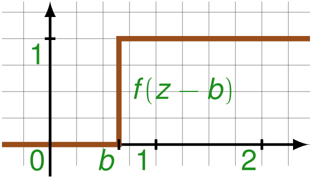
definíció szerint $\boldsymbol{x}_{\mathrm{ki}} \in \mathbb{R}^k$

Mesterséges Intelligencia
12
Csató Lehel
Multilayer
B.P.
Hebb
Self. Org.
Hebb-hálók
S.O.M.

A bemenő aktivitás:
$y_j = \sum_i w_{ij} x_i + b_j$.

A perceptron tanulási algoritmushoz hasonlóan:

- előbb: $\boldsymbol{x}_i' = [x_1, \ldots, x_d, 1]^T$
- $\Rightarrow \quad \boldsymbol{w}_{\cdot j}' = [w_1, \ldots, w_d, b]^T$



$f(z) = \mathrm{sgn}(z)$

Univerzális approximáció

A következő rendszer:

$$f_M(z|\boldsymbol{w}, \boldsymbol{b}) = \sum_{m=1}^{M} w_m f(z - b_m)$$

univerzális approximátor, azaz $\forall \delta > 0 \; \forall g \in L^2(\mathbb{R}^d)$
$\exists M, \boldsymbol{w}, \boldsymbol{b}$ ú.h. $\|g - f_M(z|\boldsymbol{w}, \boldsymbol{b})\|_{L_2} \leq \delta$

Univerzális approximátor: bizonyítás a szint-halmazokon alapszik (lásd Lebesgue integrálás).

Vizualizálás:
Mintavételezés a ***w*** és ***b*** értékekből

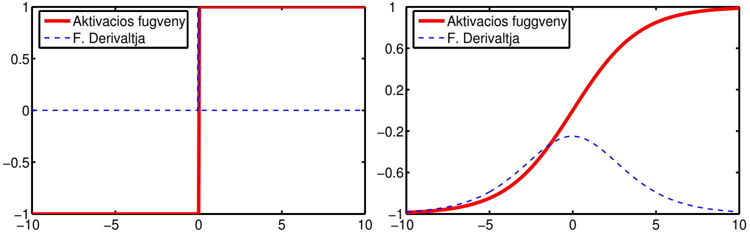
MATLAB kód:

```
%Generatorfuggvenyek
f = inline('((x>0)-0.5).*2');
g = inline('2./(1+exp(-2*x))-1');
% komponensek szama
mComp = 10;
% fv.-nyek szama
nF    = 6;
% X tengely
xx=-10:0.05:10;
xd=repmat(xx,[mComp,1]);
% plot init
clf; hf = subplot(1,2,1);
set(gca, ...
 'Position',[0.05 0.1 0.42 0.85],...
 'YTick',[],'XTick',[]);
hold on; box on; ylim([-25,25]);
set(gca,...
  'Position',[0.55 0.1 0.42 0.85],...
  'YTick',[],'XTick',[]);
hold on; box on; ylim([-25,25]);
```

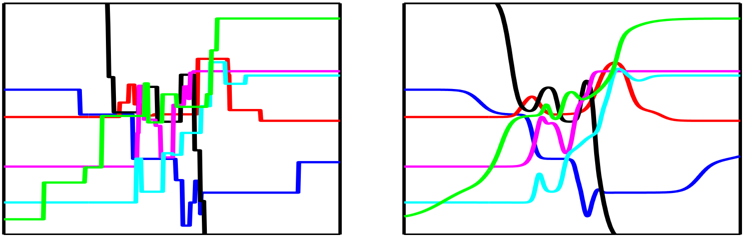
```
hg = subplot(1,2,2);
% megjelenitesi stilus
style = {'b','r','k','m','c','g'};
% F.v.-nyek generalasa
for ii=1:nF;
  % egyutthatok
  ss=(2*rand(mComp,1) -1)*6;
  ww=(2*rand(mComp,1) -1)*6;
  bb=(2*rand(mComp,1) -1)*10;

  bd=diag(bb)*ones(size(xd));
  y_f = sum(diag(ww) * f(diag(ss)*xd + bd)
  y_g = sum(diag(ww) * g(diag(ss)*xd + bd)
  % megjelenites
  plot(hf,xx,y_f,style{ii},'LineWidth',3);
  plot(hg,xx,y_g,style{ii},'LineWidth',3);
end;
% nyomtatas
print -depsc2 rand_fn.eps
```

Aktivációs függvények



Véletlen függvények a függvényosztályokból:



- **Nem bináris függvényt közelít.**

  A bemeneti/kimeneti értékek folytonosak.

$$f_{NN} : \mathbb{R}^d \longrightarrow \mathbb{R}$$

- Folytonos esetben használható a négyzetes hibafüggvény:

$$E(\boldsymbol{w}_{NN}) = \frac{1}{2}\sum_{n=1}^{N}\left(y_n - f_{NN}(\boldsymbol{x}_n)\right)^2$$

  ahol $\boldsymbol{w}_{NN}$ a neuronháló paraméterei.

- Differenciálható akt. függvényeknél a gradiens szabály alkalmazható $\Rightarrow$ **backpropagation szabály**.

- Error back-propagation $\Leftrightarrow$ négyzetes hiba visszaterjesztése

- Egy adott tanulási halmazhoz és egy $\boldsymbol{w}_{NN}$ neurális hálóhoz rendelt négyzetes hiba

$$E(\boldsymbol{w}_{NN}) = \frac{1}{2}\sum_{n=1}^{N}\left(y_n - f_{NN}(\boldsymbol{x}_n)\right)^2$$

  akkor zéró, ha a neurális háló minden adatot jól közelít.

- Egy adott $\boldsymbol{w}_{NN}^{(0)}$ hálónál jobban közelítő hálót kapunk ha egy kis lépést végzünk a negatív gradiens irányába.

Mesterséges Intelligencia
12
Csató Lehel
Multilayer
B.P.
Hebb
Self. Org.
Hebb-hálók
S.O.M.

- **„Képletesen”**

$$\mathbf{w}_{NN}^{(t+1)} \Leftarrow \mathbf{w}_{NN}^{(t)} - \alpha \frac{\partial E(\mathbf{w}_{NN}^{(t)})}{\partial \mathbf{w}_{NN}^{(t)}}$$

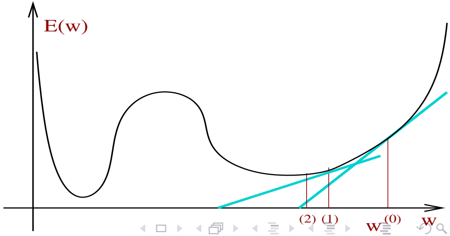
$$\Leftarrow \mathbf{w}_{NN}^{(t)} - \alpha\, \partial_{\mathbf{w}} E(\mathbf{w}_{NN}^{(t)})$$

ahol

α – tanulási együttható;

$\partial_{\mathbf{w}} E(\mathbf{w}_{NN}^{(t)})$ – a hiba gradiense (vektorok!).

Kérdés: milyen α értékek megfelelőek?
Kis lépések $\Rightarrow$ hosszú konvergencia.
Nagy lépések $\Rightarrow$ nem konvergál.

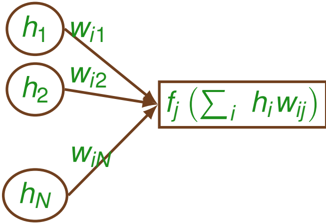


Differenciálás szabálya:

$$\frac{\partial f(g_1(w_i),\ldots,g_k(w_i))}{\partial w_i} = \sum_{g_j} \frac{\partial f(g_1,\ldots,g_k)}{\partial g_j}\frac{\partial g_j}{\partial w_i}$$

Neuronháló függvénye:

$$y = \cdots\left(\cdots, f_j\left(\sum_i h_i w_{ij}\right), \cdots\right)$$



$$\partial_{w_{ij}} E(\cdot) = \sum_j \partial_{f_j} E(\cdot)\, \partial_{w_{ij}} f_j\left(\sum_i w_{ij} h_i\right)$$

Backpropagation V

(folyt)

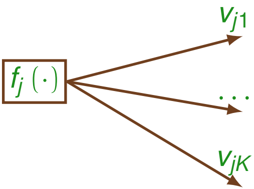
$$\partial_{w_{ij}} E(\cdot) = \partial_{f_j} E(\cdots)\ \partial_{w_{ij}} f_j \left(\sum\nolimits_{n=1}^{N} w_{ij} h_i\right)$$
$$= h_i\ \ f_j'\left(\sum\nolimits_{n=1}^{N} w_{ij} h_i\right) \partial_{f_j} E(\cdot)$$

A bemeneti érték és a kimeneti hiba szorzódnak.

És $\partial_{f_j} E(\cdot) = \partial_{f_j} E\left(\ldots, g_k\left(\sum_j v_{jk} f_j\right), \ldots\right)$

Lánc-deriválás szerint:

$$\partial_{f_j} E(\cdot) = \sum_k v_{jk}\ \ \partial_{g_k} E(\cdot) g_k'(\cdot)$$

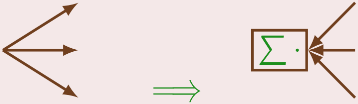


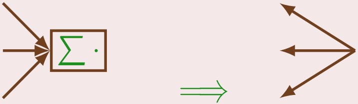
Az egyéni hibák súlyozottan összeadódnak.

Backpropagation VI

(folyt)
Tehát: amíg a működésnél a jel előre terjed, a tanulás folyamán a hiba **visszafele**; a kimeneti réteg felől a háló bemenete felé „terjed”.

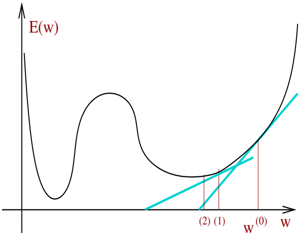
Hiba-visszaterjesztés (BP)

- 

  azaz az osztópontok összegzővé alakulnak;
- 

  azaz az összegző csomópontok meg hiba-elosztóvá.

- A BP algoritmus a gradiens szabály alkalmazása a neuronhálókra;
- Az alap a négyzetes hiba;
- Egyszerűen implementálható;
- Nagyon sok alkalmazás;



Hátrányok:

- Mivel gradiens, lassú $\Rightarrow$ nagyon sokáig tarthat a tanulás.
- Más módszerekkel állítani be a súlyokat: *Newton*, *Konjugált gradiens*.
- Nincs a becsült mennyiségre konfidencia $\Rightarrow$ valószínűségi módszerek szükségesek.

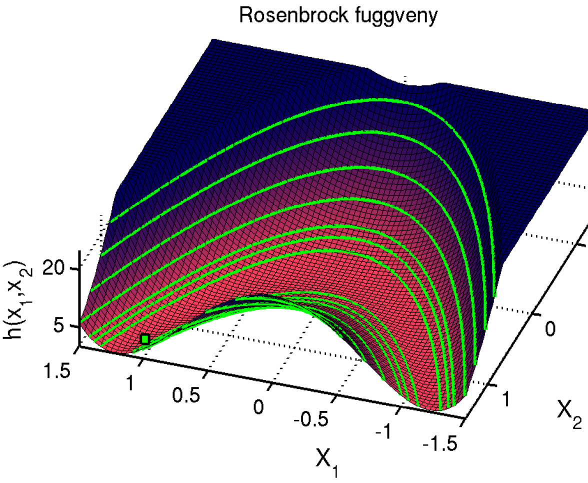
Feladat: keressük a $h(x_1, x_2) = 100\left(x_2 - x_1^2\right)^2 + (1 - x_1)^2$ függvény minimumát a gradiens szabály használatával.

$$\partial_1 h(\cdot) = 400(x_2 - x_1^2)x_1 + 2(x_1 - 1)$$
$$\partial_2 h(\cdot) = 200(x_2 - x_1^2)$$

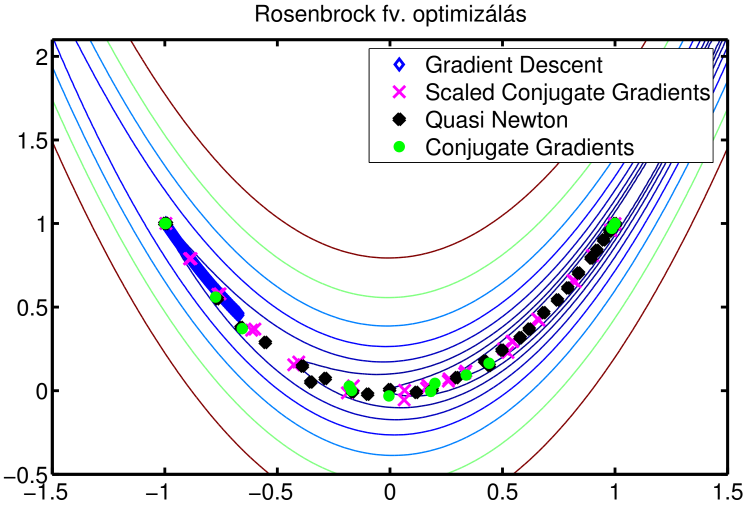
A második egyenlet $\leftrightarrow \quad x_2 = x_1$

Behelyettesítve $x_1 = 1$.

Induljunk a $(-1, 1)$ pontból.



BP szabály = gradiens – általában nagyon lassú.



- Neurális hálók **fekete-doboz** módszerek:
  1. minden feladat megoldására potenciálisan alkalmazhatóak;
  2. jó eredményekhez (majdnem mindig) kell egy kis igazítás a módszeren;
  3. siker függ a tapasztalattól;[5]

- **BackPropagation = gradiens**
  1. általában nagyon lassú;
  2. fejlett módszerek: konjugált gradiens módszerek, quasi-Newton, *etc.* gyorsítanak a konvergencián.
  3. lokális optimumok elkerülésére többszálú optimizálás, pl. mintavételezéssel kijelölni a kezdőértékeket.

[5] Érdemes tehát tovább tanulni.

D.O. Hebb
The organization of behavior

A neurális moduláció alapelvei:

- Két neuronok közötti kapcsolat erőssége arányos a neuronok pre-, illetve poszt-szinaptikus tevékenységével;
- Azon neuron–csoportok, melyek általában egyszerre tüzelnek, egy egységet alkotnak; (modularitás)
- A *gondolkodás* folyamata ezen sejt-csoportok szekvenciális aktivációja;

Organization of Behavior:
When an axon of cell A is near enough to excite B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.

(p. 62)

Neuroscience $\Leftrightarrow$ „Hebb synapse”

„Hebb szabály”

$$\Delta w_{ij} = \alpha a_i a_j$$

Ahol

a_i, a_j – neuronok aktivitásai;
w_{ij} – az összekötő szinapszis erőssége;
α – tanulási együttható.

Mesterséges Intelligencia
12
Csató Lehel
Multilayer
B.P.
Hebb
Self. Org.
Hebb-hálók
S.O.M.

Haykin: Neural Networks
A comprehensive foundation, IEEE press, 1994

Cél:

- Struktúrák felfedezése az adatokban;
- „Felügyelet” nélküli működés $\Rightarrow$ önszervezés.

Tanulái szabályok:

- lokális szabályok, melyek – sok neuronra alkalmazva – egy globális függvényt határoznak meg.
- Alapja (Turing 1952):

„Global order can arise from local interactions.”

Önszervező rendszerek alapelvei (von der Marlsburg):

- *A szinapszisok önmegerősítőek*

  A pre-, illetve posztszinaptikus neuronok egyidejű aktivációja erősíti a szinapszist (lásd Hebb).

- *A szinapszisok versengenek az erőforrásokért*

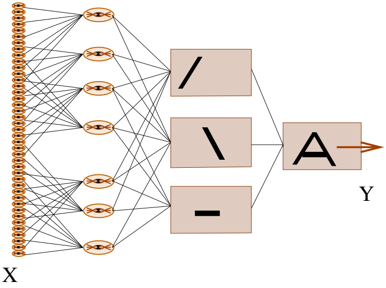
  A legerősebb szinapszis biztos, hogy túléli. Ez a többi rovására történik (winner-takes-all).

- *A szinapszisok változásai koordináltak*

  Egy szinapszis nem kódol robusztusan, ezért egy régió neuronjai – majdnem – azonos bemenet–kimenet kapcsolatot kódolnak.

Önszervező tanulás ≡ Struktúra-felismerés

- A vizuális rendszer önszervező:
  - fejlődésének elején csak a felismerésre való képesség van jelen;
  - a működés során egyes kép-csoportok – klaszterek – együttesen lesznek ábrázolva;
- A származtatott fogalmak ábrázolása a hierarchia egy felsőbb szintjén történik.



Példa:

- neurális rendszer pixelcsoportokat keres.
- némely pixelcsoport gyakori; ezen csoportok rözgülnek a rendszerben.
- a felső szinten a rendszer címkézést végez.

Hebb tanulás - auto-asszociatív rendszerek

- A rendszer bemenete egy minta - nincs kimeneti jel.
- A „kimeneten” vagy:
  - az eredeti mintát; vagy
  - egy csoportosítást, „sűrített” ábrázolást

  szeretnénk visszakapni.
- csoportosításkor a kimeneti neuronok **versengenek**:

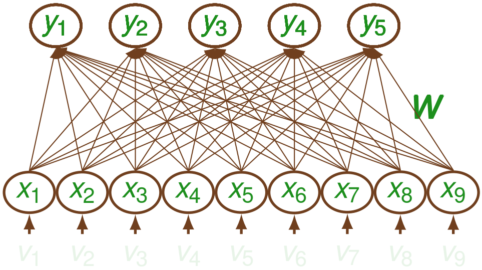
$$\hat{y}_j = \sum_i x_i w_{ij}$$

$$y_l = \begin{cases} 1 & l = k \\ 0 & l \neq k \end{cases} \qquad \text{ahol } k = \underset{j}{\operatorname{argmax}}\, \hat{y}_j$$

Hebb-féle tanulás II

$$y_l = \begin{cases} 1 & l = k \\ 0 & l \neq k \end{cases} \qquad \text{ahol } k = \operatorname*{argmax}_j \hat{y}_j$$

- Végleges kimeneti érték az argmin művelet után.
- **Versengés** a bemeneti mintákért.

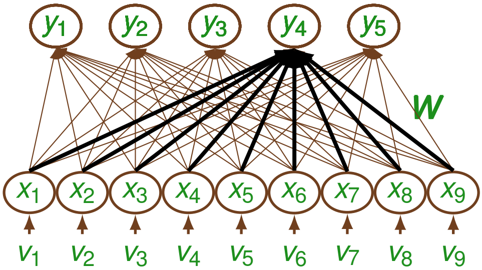


- Feltételezve egy kezdő $\mathbf{W}$ – véletlen – súlyvektort, a $\mathbf{v}$ minta bemutatása után
- csak a $w_{\cdot 4}$ súlyok változnak a kompetitív tanulás során.

Hebb-féle tanulás II

$$y_l = \begin{cases} 1 & l = k \\ 0 & l \neq k \end{cases} \qquad \text{ahol } k = \underset{j}{\operatorname{argmax}}\, \hat{y}_j$$

- Végleges kimeneti érték az argmin művelet után.
- **Versengés** a bemeneti mintákért.



- Feltételezve egy kezdő $\boldsymbol{W}$ – véletlen – súlyvektort, a $\boldsymbol{v}$ minta bemutatása után
- csak a $w_{\cdot 4}$ súlyok változnak a kompetitív tanulás során.

Hebb-féle tanulás III

$$\Delta w_{ij} = \alpha x_i y_j$$

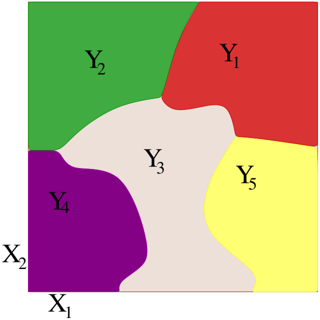
- minden y_k neuron attraktorként működik.

  receptív mezők alakulnak ki

- $\alpha \to 0$ tanulási mód, pl.

$$\alpha_t = \frac{K}{t}$$

- az **asszociatív háló** konvergál (lásd).



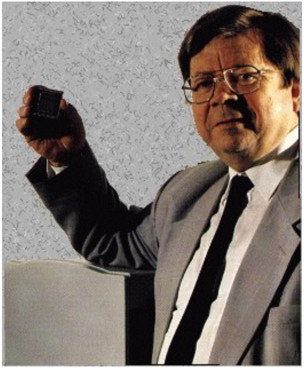
ami nincs: kapcsolatok a kimeneti neuronok között: **Kohonen-hálók**

Teuvo Kohonen

Since the 1960's, ~ introduced several new concepts to neural computing: theories of distributed associative memory and optimal associative mappings, the self-organizing feature maps (SOMs), the learning vector quantization (LVQ), . . .

the Adaptive-Subspace SOM (ASSOM) in which invariant-feature filters emerge . . .

A new SOM architecture WEBSOM has been developed in his laboratory for exploratory textual data mining.



http://websom.hut.fi/websom/

Wikipedia link

Mesterséges Intelligencia
12
Csató Lehel
Multilayer
B.P.
Hebb
Self. Org.
Hebb-hálók
S.O.M.

Jellemzők:

- kétrétegű háló, egy bemeneti és egy kimeneti réteggel;

  (mint előbb)

- kompetitív háló: csak egy neuron lesz aktív;

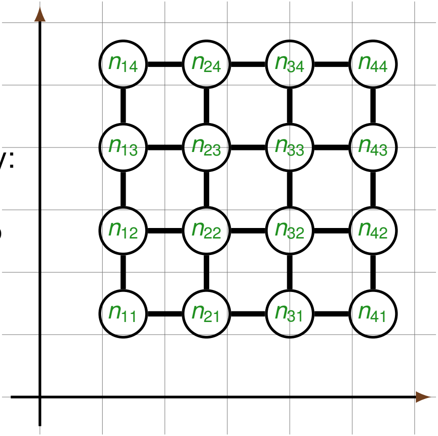
  (mint előbb)

- a kimeneti rétegen **topológia** van értelmezve

  (ÚJ)

Topológia – szomszédságot jelent

- segít a tanulásban:
  a szomszédok is tanulnak – kevésbé ...
- értelmezhető eredmény:
  a szomszédok aktivitása is aktivitása alapján pontosabb rekonstrukció
- a kimeneti n_{ij} neuronok megcímkézhetőek:
  Kohonen: ... fonéma annotáció



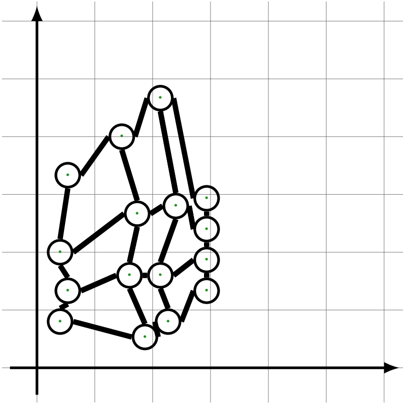
Mesterséges Intelligencia
12
Csató Lehel
Multilayer
B.P.
Hebb
Self. Org.
Hebb-hálók
S.O.M.

Az algoritmus iteratív:

1. minták egyenként; a t-edik időpontban legyen $\boldsymbol{x}_t$
2. kiszámtítjuk a kimeneti neuronok aktivitását:
$$y_j = \sum_i w_{ij} x_i(t) = \boldsymbol{W}\boldsymbol{x}(t)$$
3. megkeressük a nyertes kimeneti neuront:
$$k \stackrel{\text{def}}{=} \operatorname*{argmax}_j \; y_j$$
4. Az összes szomszédos neuron súlyát módosítjuk:
$$w_{ij}(t+1) = w_{ij}(t) + \alpha \; \eta(j,k) x_i(t)$$
Normáljuk a súlyokat $w_{.j} \Leftarrow w_{.j} / \|w_{.j}\|$
5. GOTO 1

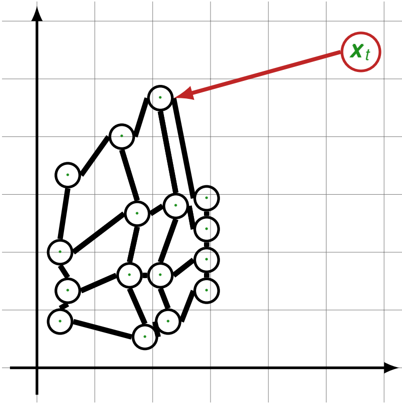
Szavakban:

- a nyertes neuron „aktivizálódik”;
- A nyertes neuront és szomszédait „közelebb” hozzuk a bemenethez;
- Mindegyik neuron specializálódik a mintákra: arra lesz a legnagyobb a kimenete;



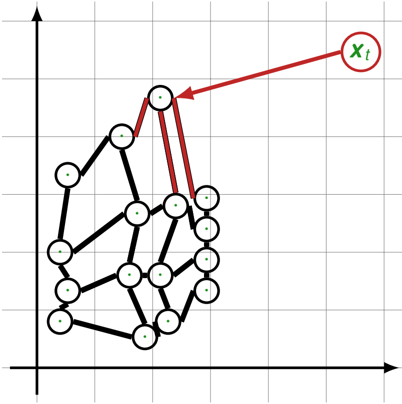
Szavakban:

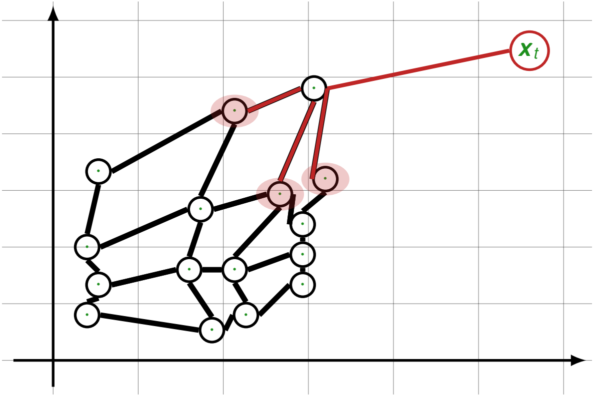
- a nyertes neuron „aktivizálódik”;
- A nyertes neuront és szomszédait „közelebb” hozzuk a bemenethez;
- Mindegyik neuron specializálódik a mintákra: arra lesz a legnagyobb a kimenete;



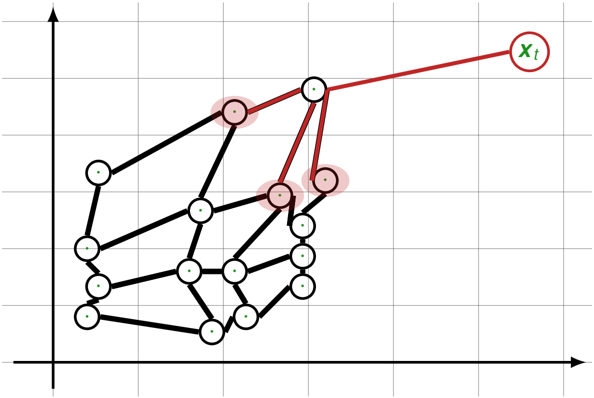
Szavakban:

- a nyertes neuron „aktivizálódik”;
- A nyertes neuront és szomszédait „közelebb” hozzuk a bemenethez;
- Mindegyik neuron specializálódik a mintákra: arra lesz a legnagyobb a kimenete;





Eredmény: a neuronok elhelyezkedését a bemeneti minták sűrűség–függvénye határozza meg – több neuron kerül a sűrűbb régiókba.



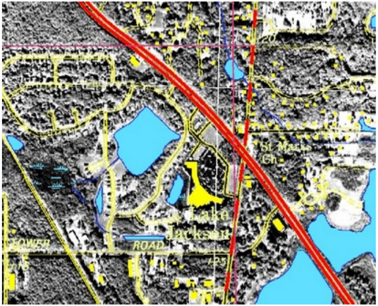
Eredmény: a neuronok elhelyezkedését a bemeneti minták sűrűség–függvénye határozza meg – több neuron kerül a sűrűbb régiókba.

- Az adatokhoz nincs kimeneti érték rendelve;
- A rendszer általában az adatok egy csoportosítását valósítja meg;
- Ez implicit módon a bemeneti adatok terének a felosztását jelenti;
- Zajszűrés/Sűrítés: a teljes bemeneti adatok helyett a hozzá tartozó osztály kódját küldjük.

Példa – Térképészgép

Walter Bischof, Jun Zhou (U. Alberta) és Terry Caelli (Australian N.U) a kartográfiát `(részben)` automatizáló programmal álltak elő.

A program embertől tanulja, miként fedezzen fel és azonosítson légi felvételeken korábban **nem szereplő vagy megváltozott** elemeket: utakat, vasutakat, épületeket.



Ágens portál *Walter Bischof projektek*

Működése:

Egyre többet tud meg – „okosabb” lesz. Kellő mennyiségű gyakorlás után az operátor rábízza a munkát. *Bayes-féle következtetést használ*: korábbi mérések alapján végez új becsléseket. A legjobbakból állapítja meg az adott országút soron következő pontjait, és ezt mindaddig folytatja, amíg az új elemek pozíciója *megfelel az elvárásoknak*.