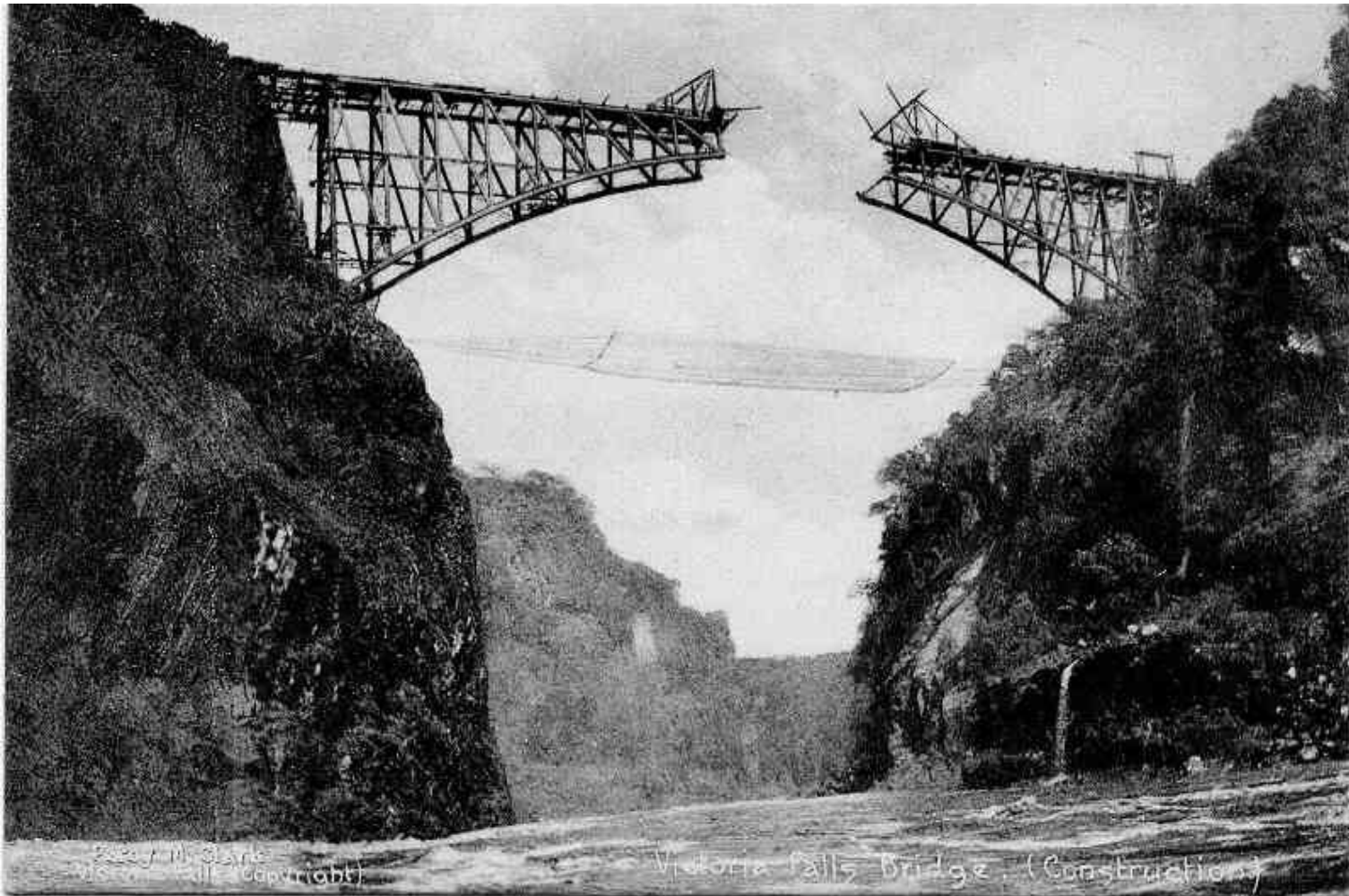


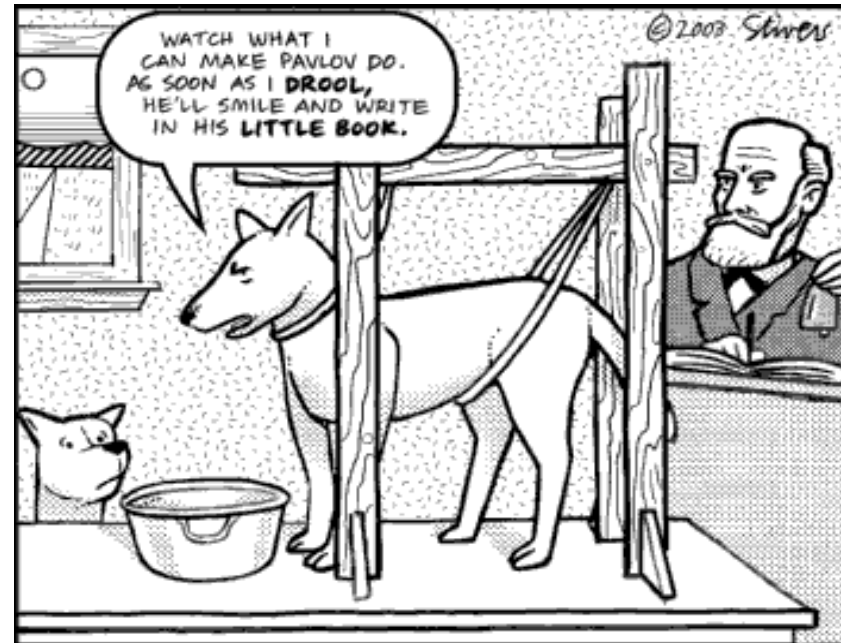
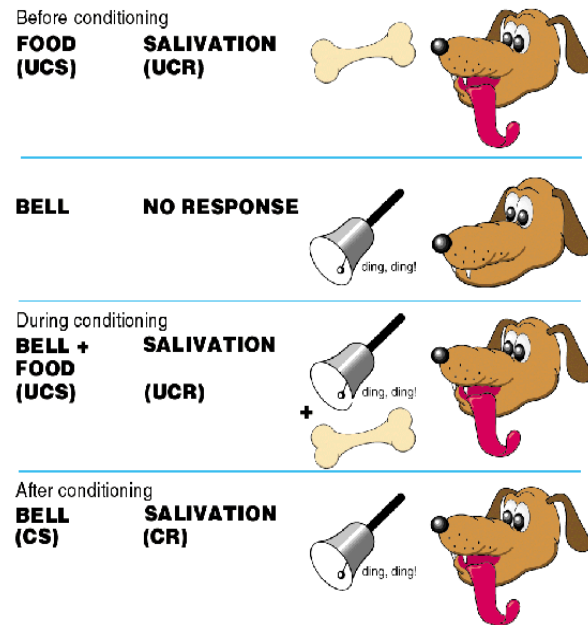
Tanulás az idegrendszerben



Structure – Dynamics – Implementation – Algorithm – Computation - Function

Tanulás pszichológiai szinten

- Classical conditioning

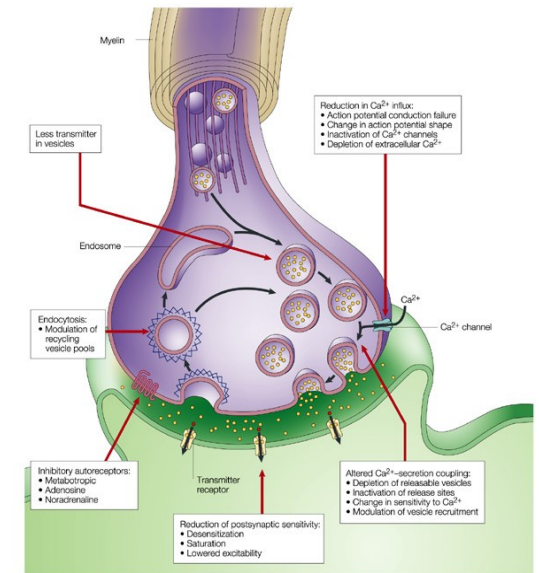
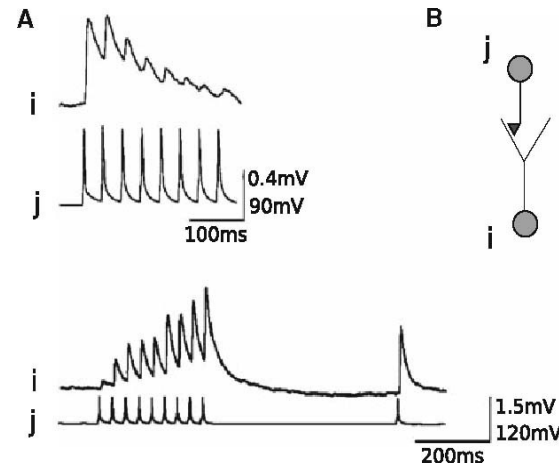


- Hebb ötlete:

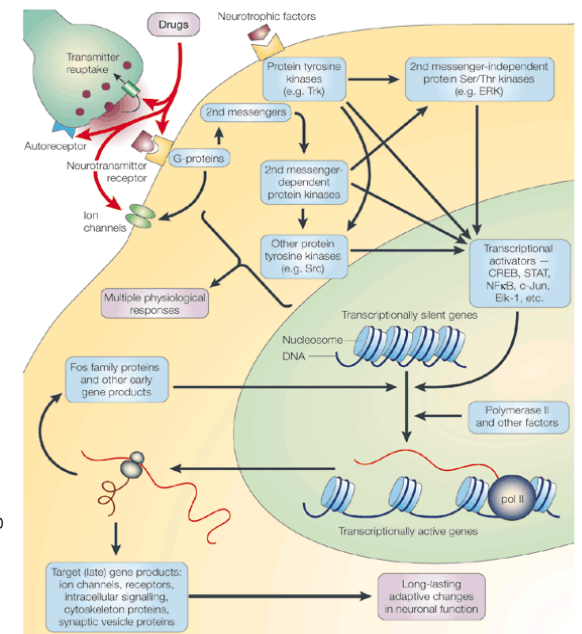
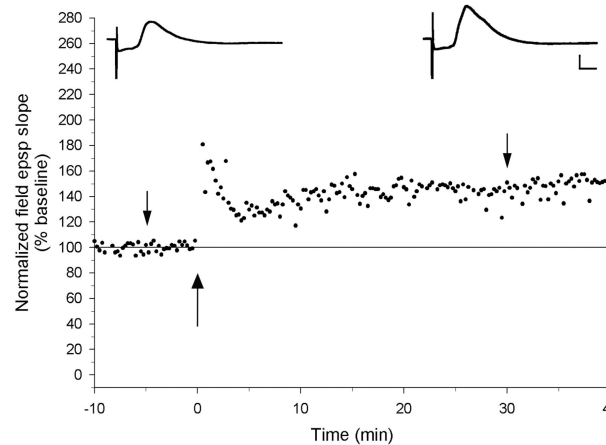
"Ha az A sejt axonja elég közel van a B sejthez, és ismétlődően vagy folyamatosan hozzájárul annak tüzeléséhez, akkor valamely, egyik vagy mindkét sejtre jellemző növekedési folyamat vagy metabolikus változás következménye az lesz, hogy az A sejt hatékonysága a B sejt tüzeléséhez való hozzájárulás szempontjából megnő."

Idegrendszeri plaszticitás

- A plaszticitás helye: szinapszisok, posztzinaptikus sejtek tüzelési küszöbei (excitabilitás)
- Potenciáció, depresszió
- STP: kalciumdinamika, transzmitterkimerülés tartam < 1 perc



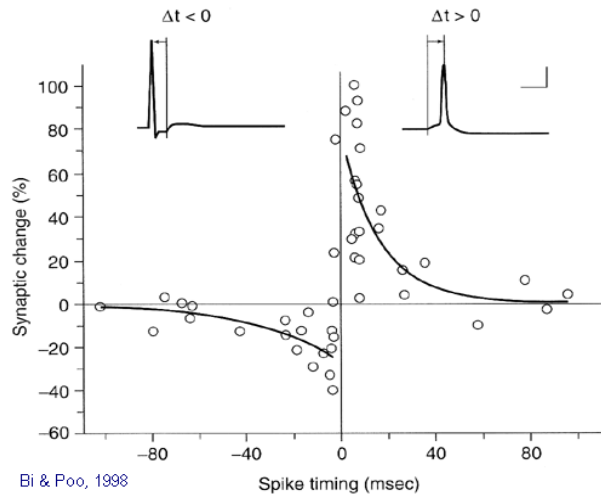
- LTP: génexpresszió (induction, expression, maintenance), NMDA magnézium-blokkja tartam > 1 perc



- Korreláció a molekuláris és pszichológiai szint között

A Hebb-szabály

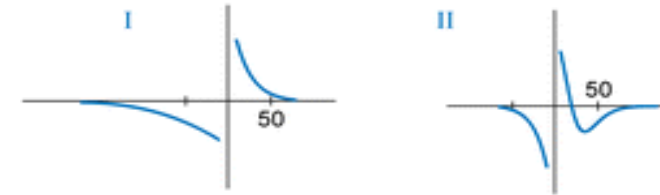
- Timing-dependent plasticity:
 - Ha a posztzinaptikus neuron nagy frekvenciával közvetlenül a preszinaptikus után tüzel, akkor erősödik a kapcsolat
 - Az alacsony frekvenciájú tüzelés gyengíti a kapcsolatot
 - Sok más lehetőség



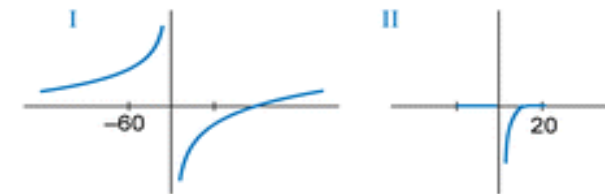
- A Hebb-szabály formalizációja:
lineáris ráta-modellben

$$\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{v} \mathbf{u}$$

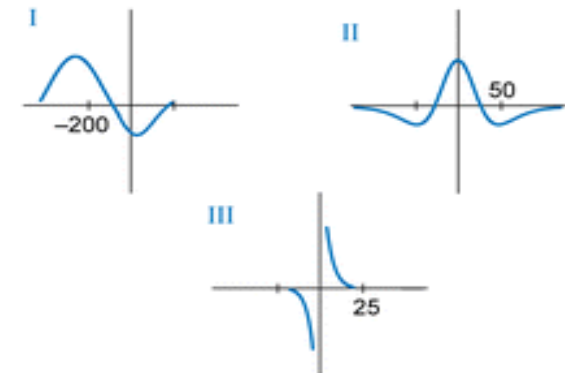
a Excitatory to excitatory



b Excitatory to inhibitory



c Inhibitory to excitatory



Stabilizált Hebb-szabályok

- Problémák a Hebb szabállyal:
 - csak nőni tudnak a súlyok
 - Nincs kompetíció a szinapszisok között – inputszelektivitás nem megvalósítható
- Egyszerű megoldás: felső korlát a súlyokra
- BCM: a posztszinaptikus excitabilitás felhasználása a stabilizásra

$$\tau_w \frac{d \mathbf{w}}{dt} = v \mathbf{u} (v - \theta_u) \qquad \tau_\theta \frac{d \theta_u}{dt} = v^2 - \theta_u$$

- Szinaptikus normalizáció

- Szubsztraktív normalizáció

$$\tau_w \frac{d \mathbf{w}}{dt} = v \mathbf{u} - \frac{v (\mathbf{1} \cdot \mathbf{u}) \mathbf{1}}{N_u}$$

Globális szabály, de generál egyes megfigyelt mintázatokat (Ocular dominance)

- Oja-szabály

$$\tau_w \frac{d \mathbf{w}}{dt} = v \mathbf{u} - \alpha v^2 \mathbf{u}$$

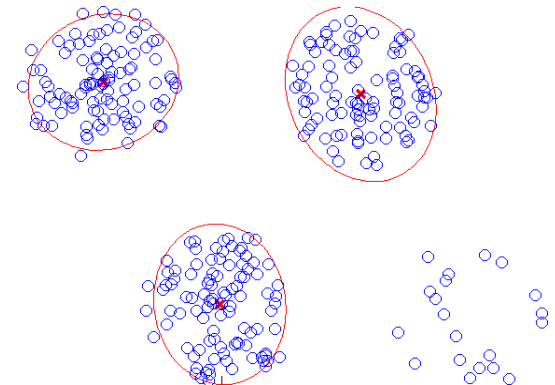
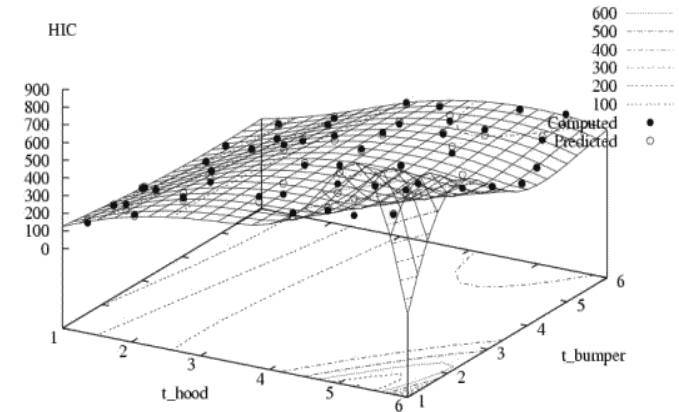
Lokális szabály, de nem generálja a megfigyelt mintázatokat

A tanulás problémája általánosan

- Modell paramétereinek hangolása adatok alapján
- Kettős dinamika
 - Változók (bemenet-kimenet leképezés) - gyors
 - Paraméterek - lassú
- Memória és tanulás különbsége
 - Memória használatánál a bemenetre egy konkrét kimenetet szeretnék kapni a reprezentáció megváltoztatása nélkül
 - Tanulásnál minden bemenetet felhasználok arra, hogy finomítsam a reprezentációt, miközben kimenetet is generálok
- Alapvető cél: predikciót adni a jövőbeli történésekre a múlt alapján

A tanulás alapvető típusai

- Felügyelt
 - Az adat: bemenet-kimenet párok halmaza
 - A cél: függvényapproximáció, klasszifikáció
- Megerősítéses
 - Az adat: állapotmegfigyelések és jutalmak
 - A cél: optimális stratégia a jutalom maximalizálására
- Nem felügyelt, reprezentációs
 - Az adat: bemenetek halmaza
 - A cél: az adat optimális reprezentációjának megtalálása / magyarázó modell felírása
- Egymásba ágyazások

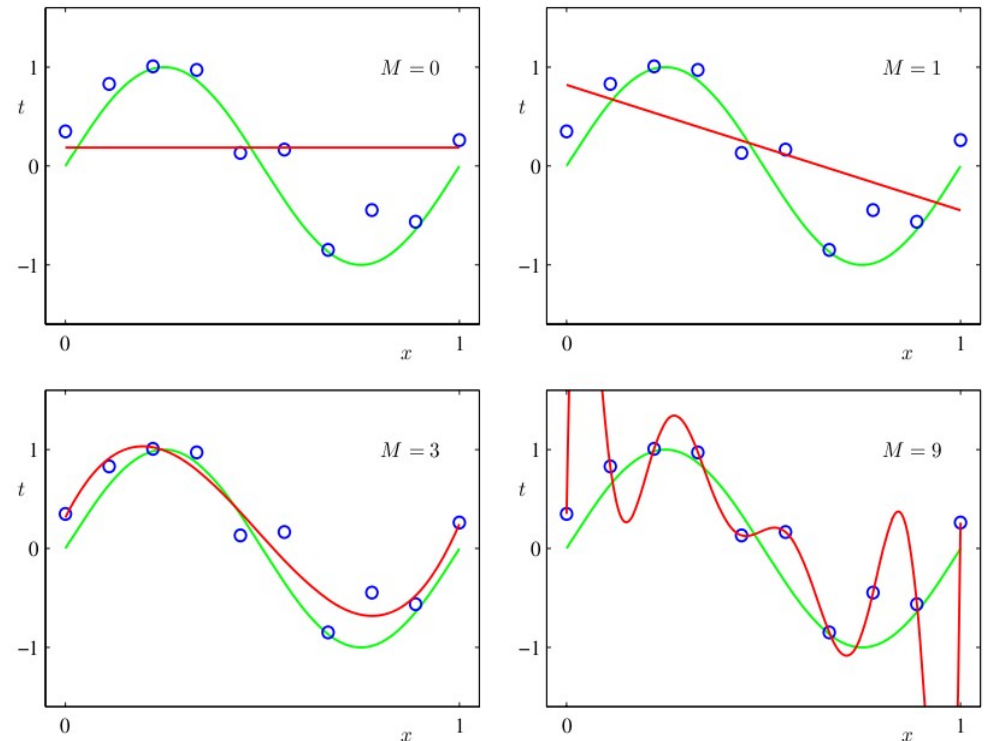


Problémák tanulórendszerekben

- Bias-variance dilemma
 - Strukturális hiba: a modell optimális paraméterekkel is eltérhet a közelítő függvényről (pl lineáris modellt illesztünk köbös adatra)
 - Közelítési hiba: a paraméterek pontos hangolásához végtelen tanítópontra lehet szükség

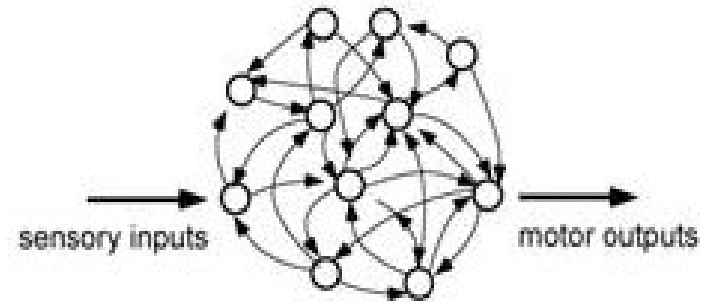
- Pontosság vs. általánosítás

- A sokparaméteres modellek jól illeszkednek, de rosszul általánosítanak: túlillesztés
- A magyarázó képességük is kisebb (lehet): Ockham borotvája



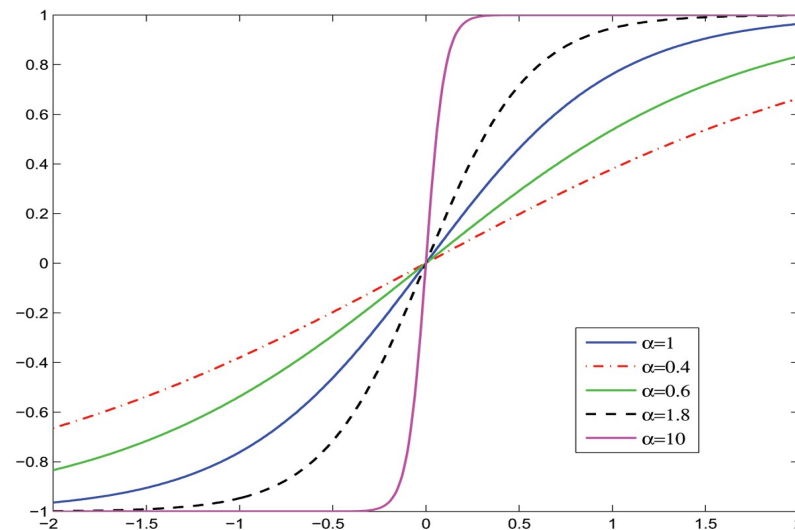
Tanulásra alkalmas neurális rendszerek

- Egyetlen sejt
- Előrecsatolt hálózat
- Rekurrens hálózat
- Ezen az órán: rátamodell

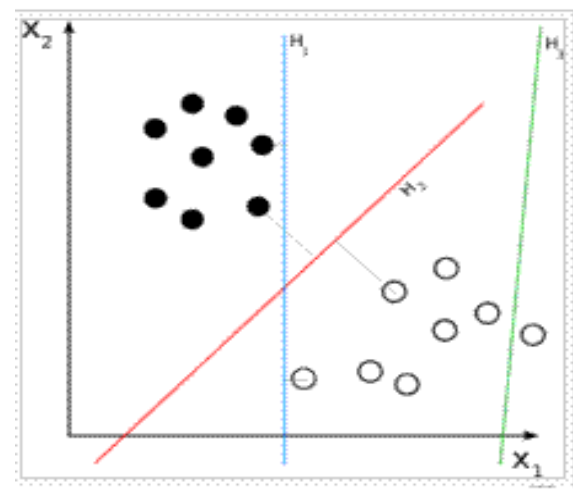
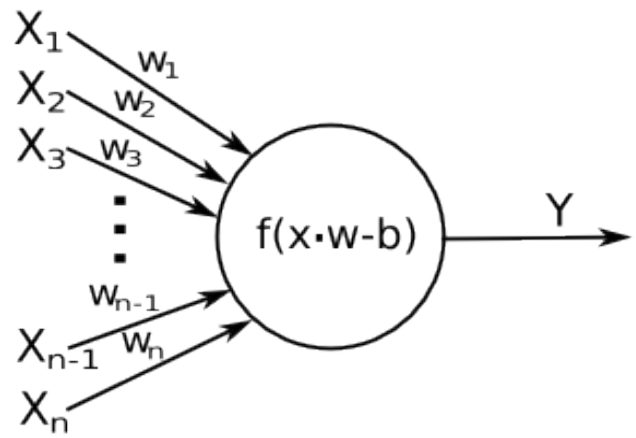


$$v = f(\mathbf{uw} - \theta)$$

- Paraméterek: súlyok, küszöbök
- Különböző kimeneti nemlinearitások
 - Lépcső
 - Sigmoid
 - Lineáris neuron



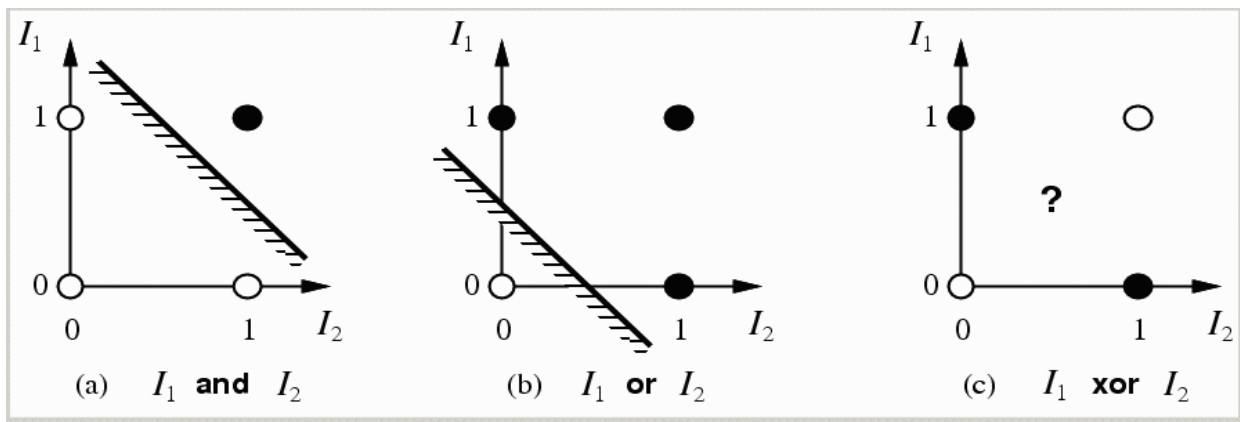
Perceptron



- Bináris neuron: lineáris szeparáció
 - Két dimenzióban a szeparációs egyenes:

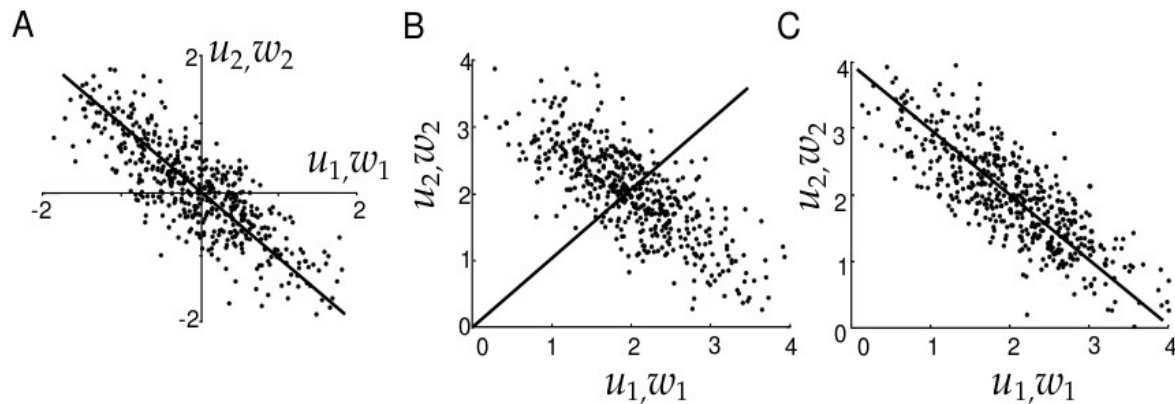
$$\theta = x_1 w_1 + x_2 w_2 \quad \longrightarrow \quad x_2 = \frac{-w_1}{w_2} x_1 + \frac{\theta}{w_2}$$

- Logikai függvények



Hebbi tanulás perceptronnal

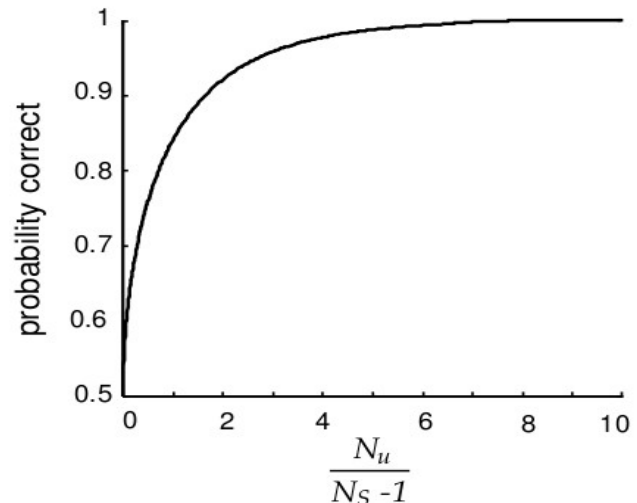
- Nem felügyelt
A bemeneti adatok főkomponensei irányába állítja be a súlyokat



- Felügyelt

$$\tau_w \frac{d\mathbf{w}}{dt} = \frac{1}{N_s} \sum_{m=1}^{N_s} v^m \mathbf{u}^m$$

- A kimenetet is meghatározzuk
- Nem minden tanítóhalmazra lehet így megkonstruálni a szeparáló súlyokat



Error-correcting tanulási szabályok

- Felhasználjuk azt az információt, hogy milyen messze van a céltól a rendszer
- Rosenblatt-algoritmus – bináris neuron

$$\mathbf{w} \rightarrow \mathbf{w} + \epsilon (v^m - v(\mathbf{u}^m)) \mathbf{u}^m$$

- Delta-szabály

- Folytonos kimenetű neuron – gradiens-módszer

$$w_b \rightarrow w_b - \epsilon \frac{\partial E}{\partial w_b} \quad E = \frac{1}{2} \sum_m^{N_s} (v^m - v(\mathbf{u}^m))^2 \quad \frac{\partial E}{\partial w_b} = - \sum_m^{N_s} (v^m - v(\mathbf{u}^m)) \mathbf{u}^m$$

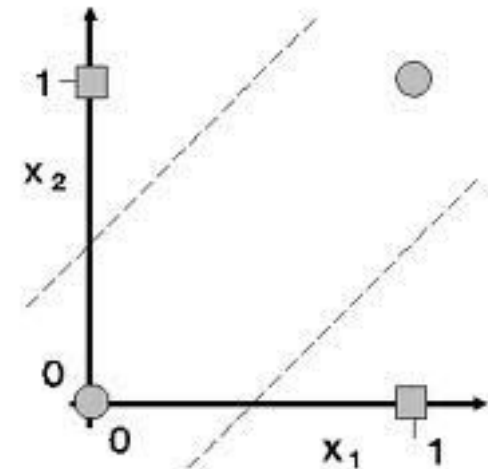
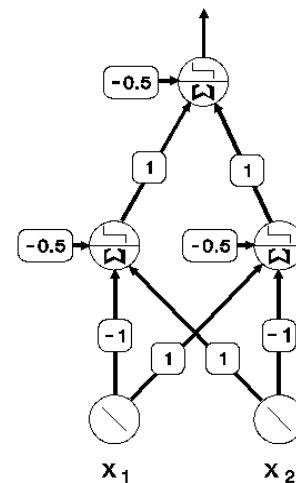
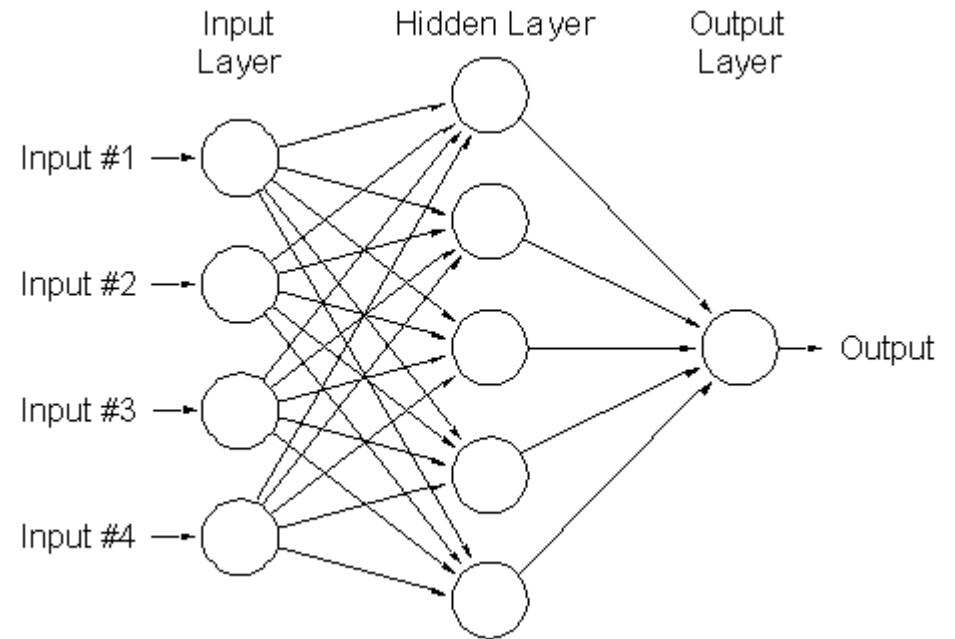
lineáris neuronra, egy pontpárra vonatkozó közelítéssel:

$$\mathbf{w} \rightarrow \mathbf{w} + \epsilon (v^m - v(\mathbf{u}^m)) \mathbf{u}^m$$

- Minsky-paper 1969: a neurális rendszerek csak lineáris problémákat tudnak megoldani

Multi-Layer Perceptron

- Nemlineáris szeparáció
- regresszió
- egyenletesen sűrű I2-ben egy rejtett réteggel
- A reprezentációs képességet a rejtett réteg növelésével tudjuk növelni
- Idegrendszerben – látórendszer ...



Error backpropagation

- A Delta-szabály alkalmazása
- Első lépés: minden neuron kimenetét meghatározzuk a bemenet alapján $o = sig(\mathbf{xw})$
- A hibafüggvény parciális deriváltjai:

$$sig'(y) = sig(y)(1 - sig(y))$$

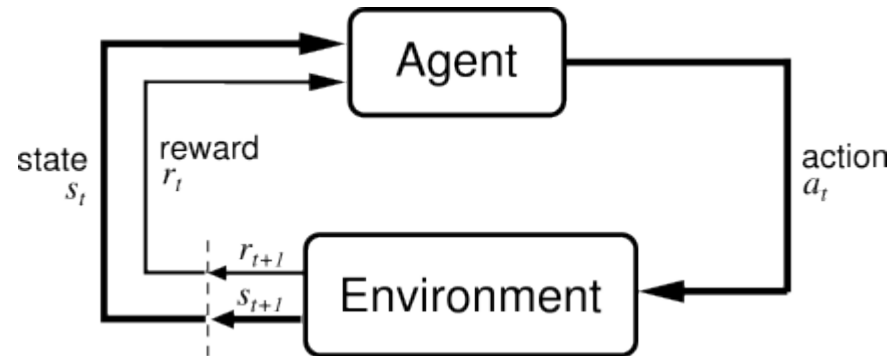
Minden neuronra: $\frac{\partial E}{\partial w_{bi}} = o_i \delta_i$

Kimeneti rétegben: $\delta_k = o_k(1 - o_k)(o_k - t_k)$

Rejtett rétegben: $\delta_j = o_j(1 - o_j) \sum w_{jq} \delta_q$

- Mivel gradiens-módszer, a hibafüggvény lokális minimumába fog konvergálni.

Megerősítéses tanulás



- Állapottér: a szenzorikus (vagy egyéb bemeneti) változók lehetséges értékeinek kombinációjából előálló halmaz
- Jutalomszignál: bizonyos állapotokban kapunk információt a cselekvésünk sikerességéről
- Cselekvés: a tanuló megvalósít egy állapotátmenetet (legalábbis megpróbálja)
- Cél: a jutalom hosszú távú maximalizálása
- Értékfüggvény: az egyes állapotokhoz rendelt hasznosság
- Értékfüggvény reprezentációja:
 - Táblázattal (machine learningben)
 - Általános függvényapproximátorral – pl. előrecsatolt neurális hálózat
 - Beágyazhatunk egy felügyelt rendszert a megerősítésesbe a háló tanítására

Temporal difference learning

- Prediction error felhasználása a tanuláshoz
- Az állapotérték frissítése neurális reprezentációban:

$$w(\tau) \rightarrow w(\tau) + \epsilon \delta(t) u(t-\tau) \quad \delta(t) = \sum_{\tau} r(t+\tau) - v(t)$$

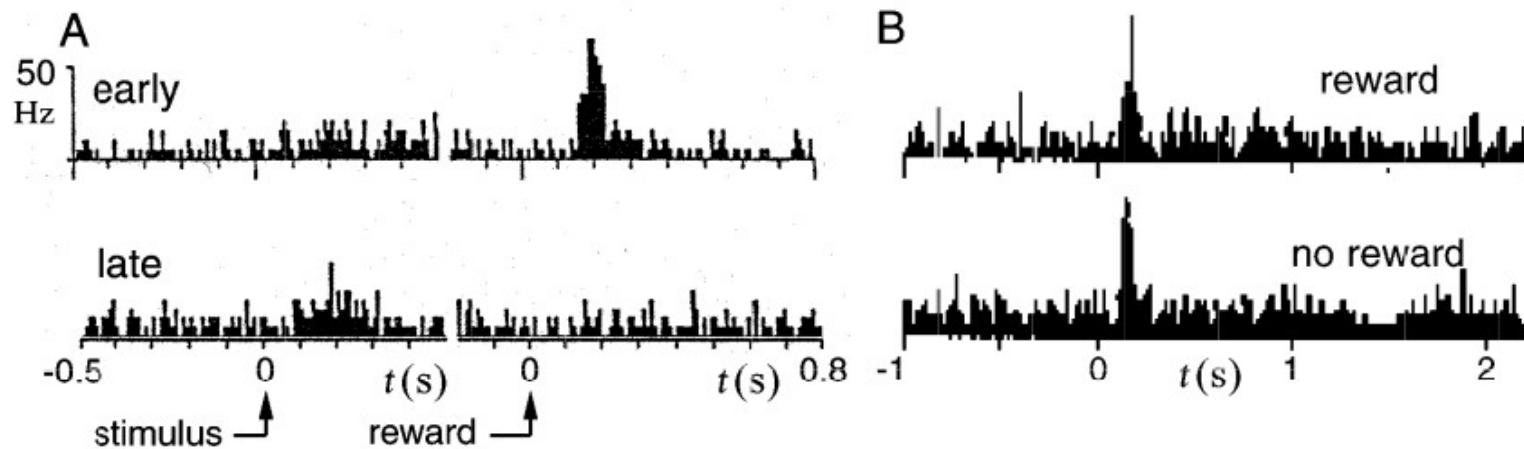
- A prediction error kiszámítása
 - A teljes jövőbeli jutalom kellene hozzá
 - Egylépéses lokális közelítést alkalmazunk

$$\sum_{\tau} r(t+\tau) - v(t) \approx r(t) + v(t+1)$$

- Ha a környezet megfigyelhető, akkor az optimális stratégiához konvergál
- A hibát visszaterjeszthetjük a korábbi állapotokra is (hasonlóan a backpropagation algoritmushoz)
- Akciókiválasztás: exploration vs. exploitation

Reward-szignál az agyban

- A ventrális tegmentális terület dopaminerg rendszerének aktivitása korrelál a jutalom predikciós hibájával:



- Stimulus-jutalom asszociációt tanítottak meg egy majomnak (A ábra)
- A dopaminerg rendszer baseline alatti inhibícióval reagál a jutalom elmaradására a tanulás után

Házi feladat

- Reprodukáljuk a D-A supervised-Hebb ábráját!
- Generáljuk bemeneti vektorok százalékes halmazát, és definiáljunk 100 különböző lineáris szeparációt közöttük!
- Állítsuk be a súlyokat a supervised Hebb-szabály és a delta-szabály segítségével, random kezdőfeltétellel!
- A neuron szinaptikus súlyainak száma legyen: 5, 15, 30
- Plottoljuk ki a helyesen megtanult szeparációk arányát a következő tengelyeken:
 - Súlyok száma / (bemenetek száma -1)
 - Helyesen megtanult szeparációk aránya