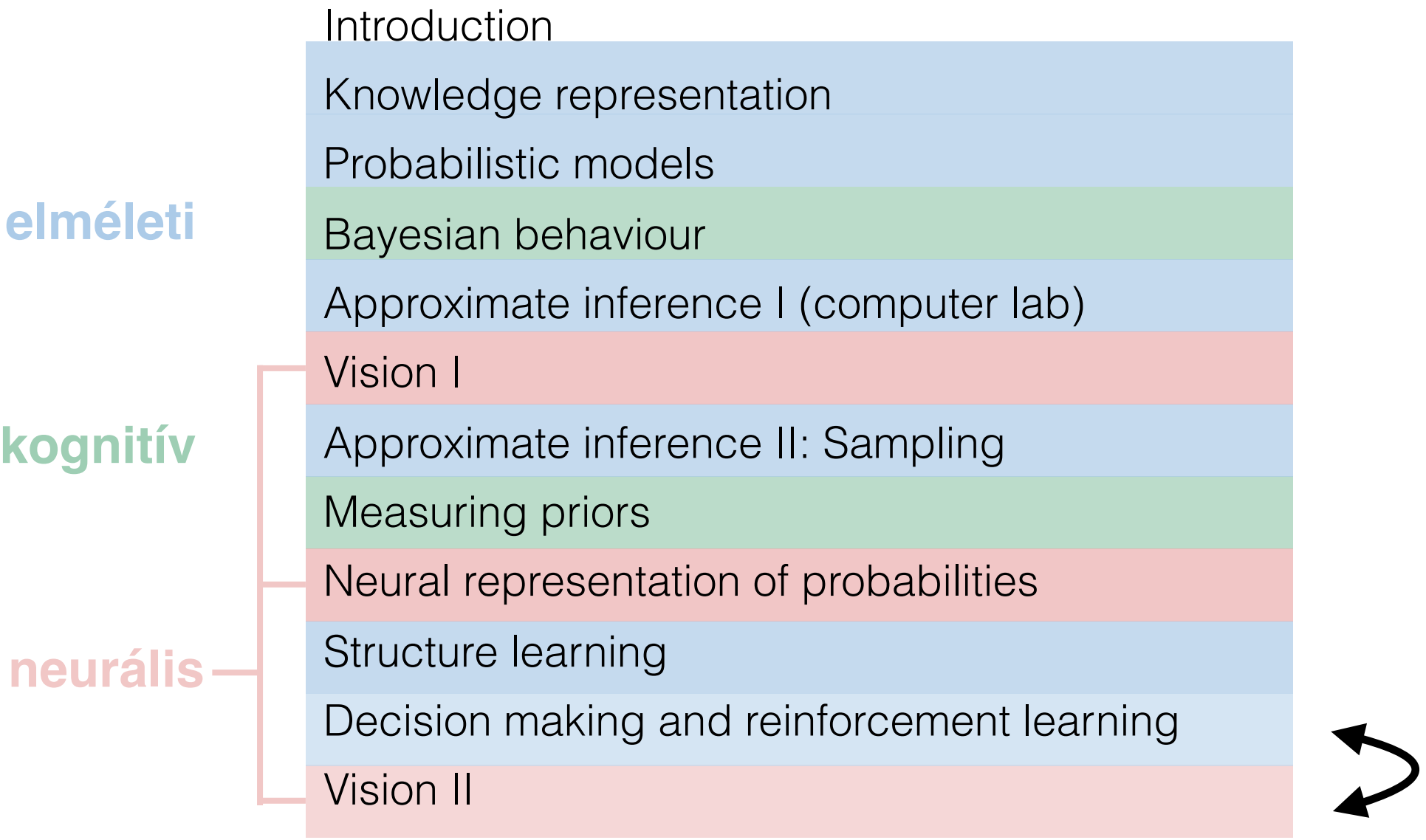
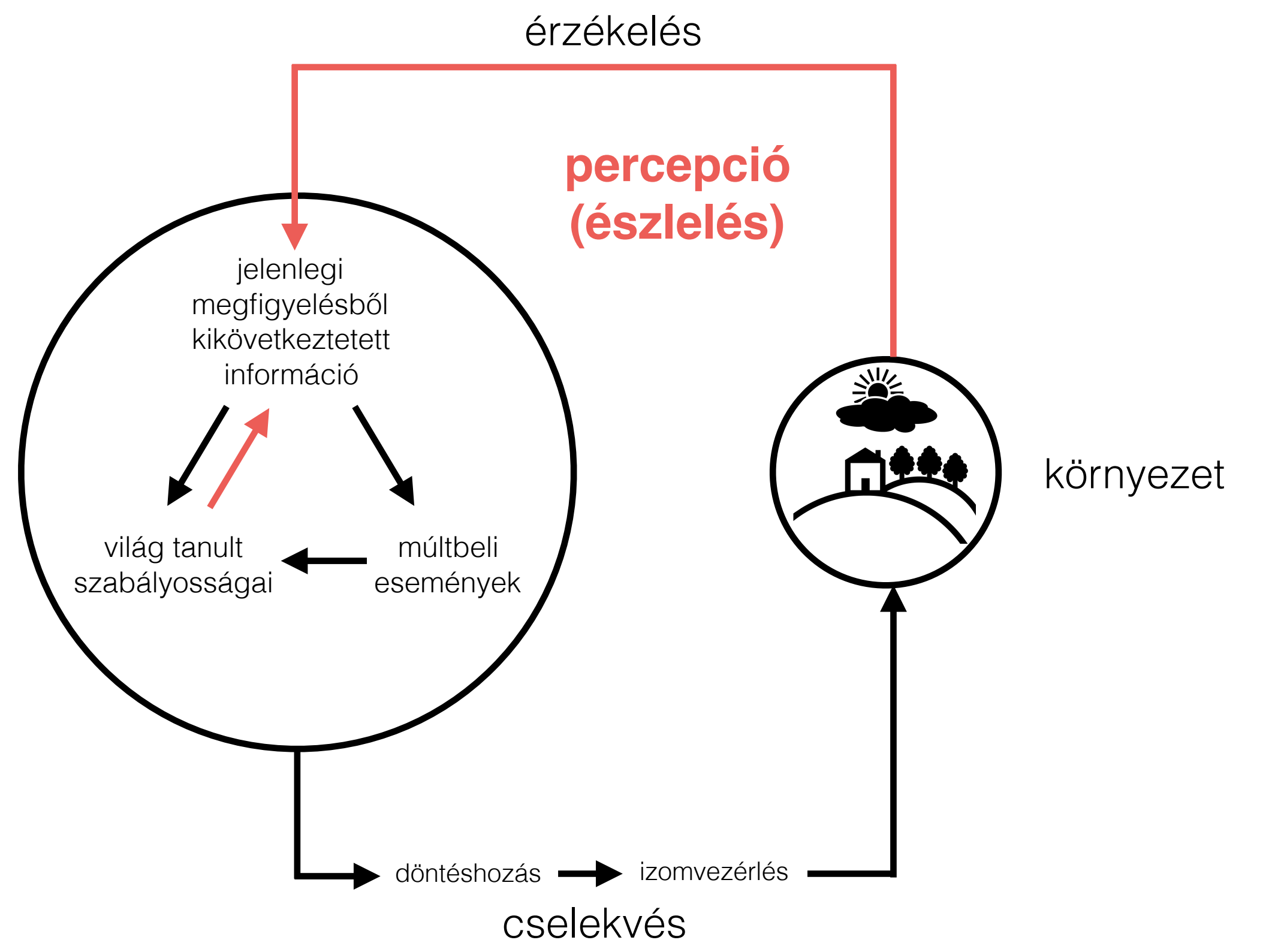
Megerősítéses tanulás


Introduction
Knowledge representation
Probabilistic models
elméleti
Bayesian behaviour
Approximate inference I (computer lab)
Vision I
kognitív
Approximate inference II: Sampling
Measuring priors
Neural representation of probabilities
neurális
Structure learning
Decision making and reinforcement learning
Vision II


ALPHAGO
01:58:30
LEE SEDOL
01:59:41
Mind
Match
AlphaGo
Lee Sedol


érzékelés
percepció
(észlelés)
jelenlegi
megfigyelésből
kikövetkeztetett
információ
világ tanult
szabályosságai
múltbeli
események
környezet
döntéshozás
izomvezérlés
cselekvés

The Big Program of the Brain

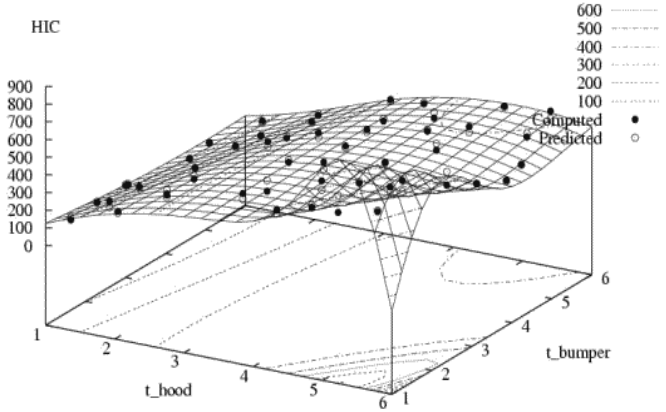
```
function MotorOutput = Brain(SensoryInput)

if (Dead ~= 1)
    for i = 1: NumSenses
        ExtractedFeatures(i) = ProcessSense(i, SensoryInput);
    end
    WorldModel = UpdateWorldModel(ExtractedFeatures,InternalState);
    MotorOutput = GenerateAction(WorldModel);
end
```

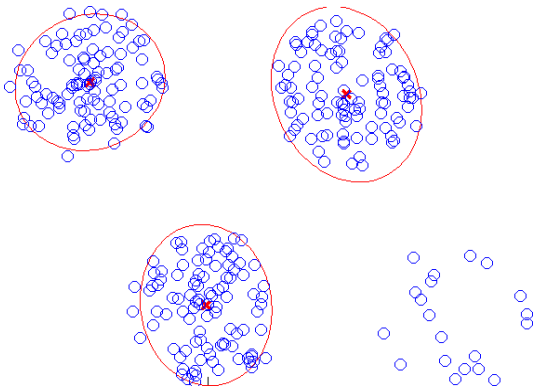
Hogyan építsünk döntési modellt az érzékelési modellre?

- Választanunk tell egy optimalizálandó célváltozót - ezt hívjuk jutalomnak (reward)
- A motoros kimenetet véges mennyiségű fix cselekvésként modellezük, amelyek módosítják a környezetet
- Az érzékelési model látens változóira adott megfigyelés mellett következtetett értékek kombinációja a környezet (érzékelt) állapota
- Ki tell találnunk, hogy a környezet mely állapotában mely cselekvést válasszuk

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

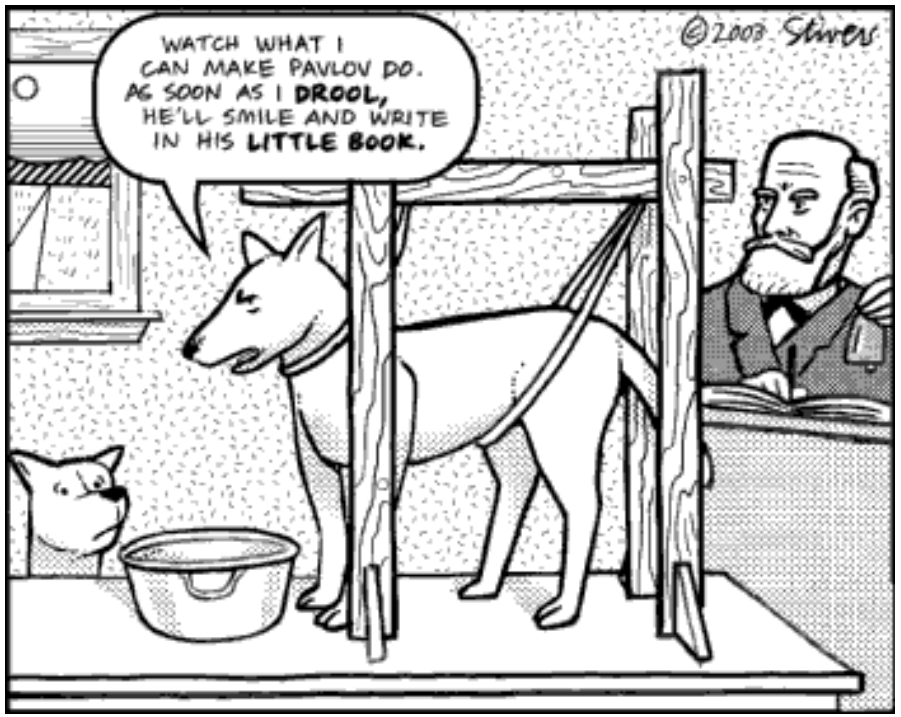


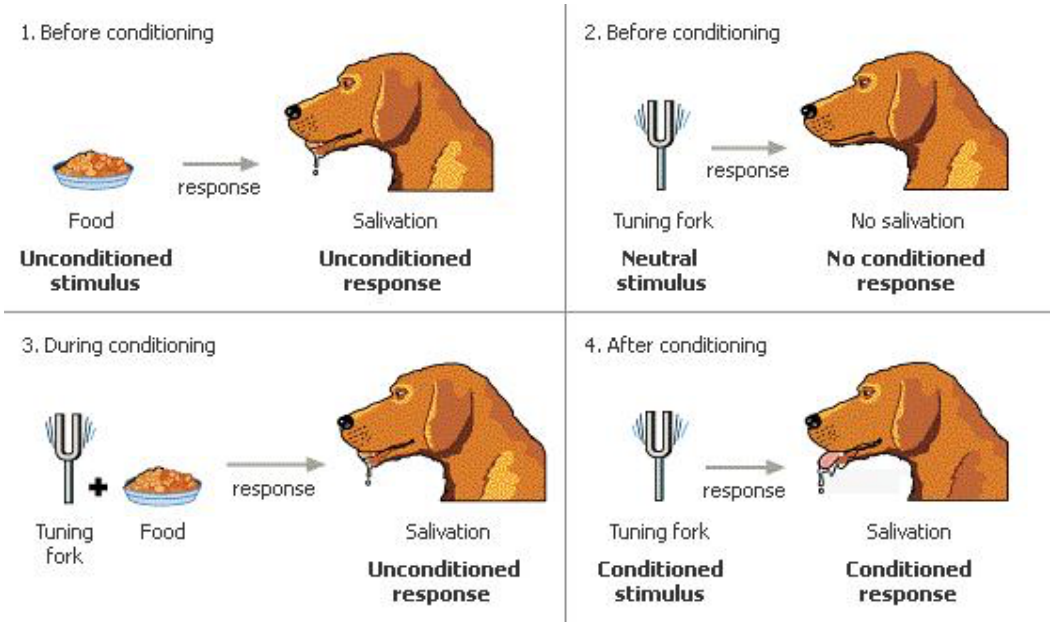


A jutalom előrejelzésének tanulása

Pavlovi - klasszikus kondicionálás

- csak az állapotok értékét tanuljuk, cselekvés nincs

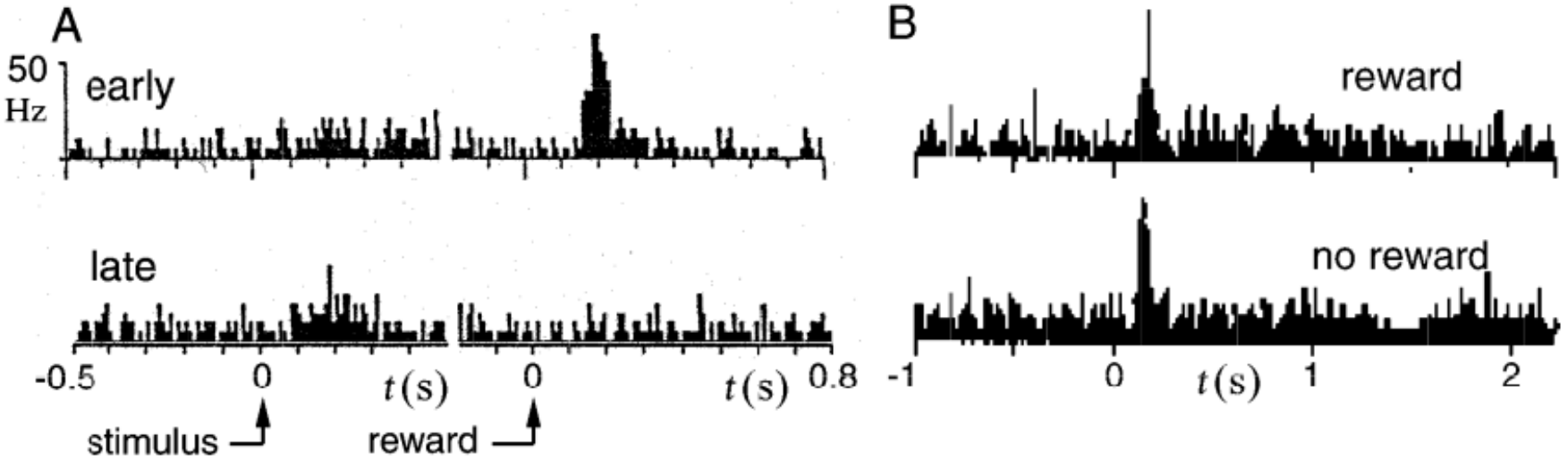




operáns kondicionálás

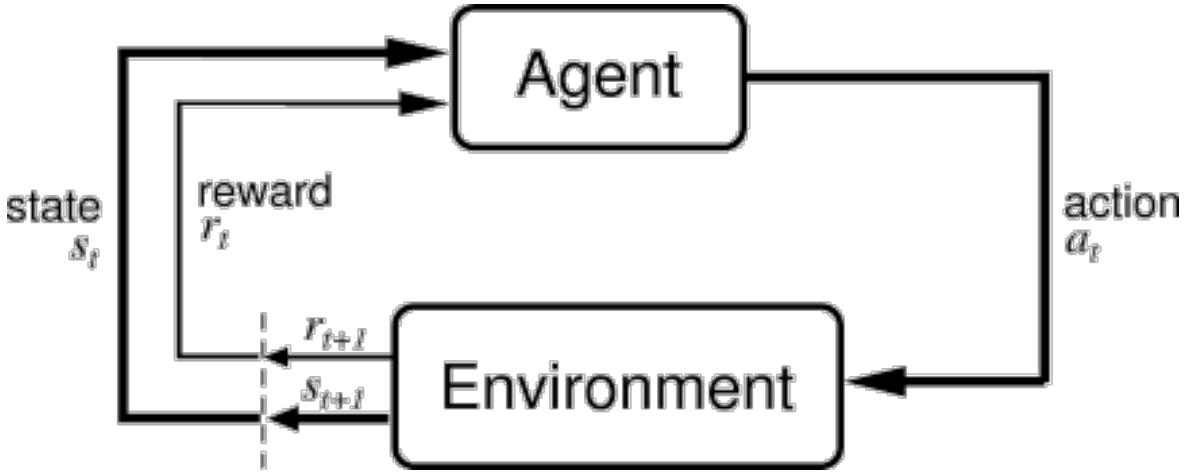
- a cselekvések határozzák meg a jutalmat

A jutalom kódolása az agykéregben



- Egyes dopamintermelő neuronok a majom agykérgében a jutalom és egyes környezeti változók között tanult kapcsolat alapján tüzelnek
- Az aktivitás a meglepetés mértékével és előjelével arányos

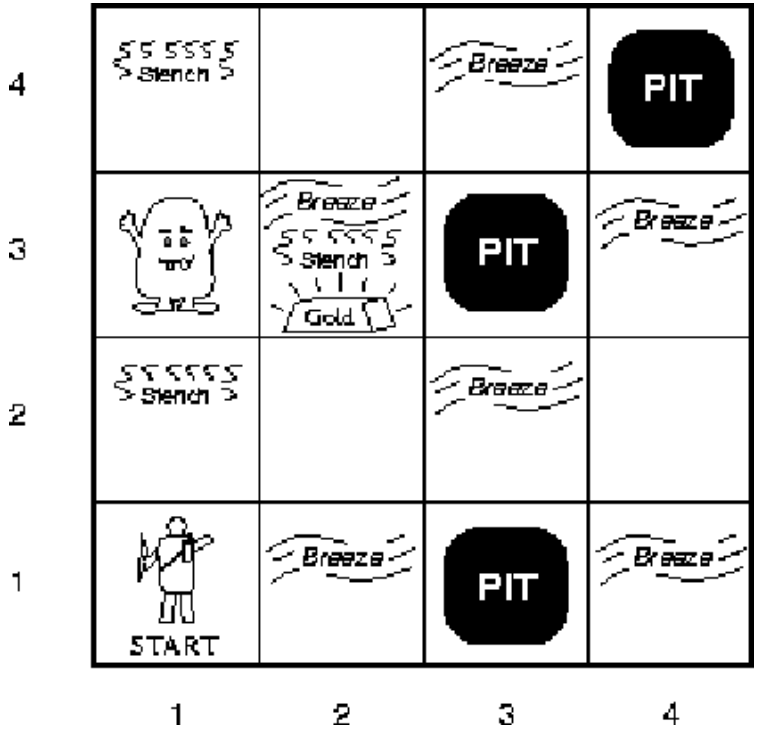
Az ágens-környezet modell



- Ágens: a tanulórendszer
- Környezet: minden, amit nem belsőleg állít elő
- Bemenet: aktuális állapot, jutalom
- Kimenet: cselekvés

Állapotok értékének tanulása

- Legegyszerűbb eset: véges mennyiségű állapotból áll a környezet
- minden állapot, *s,* a mentális model lateens változóinak ego kikövetkeztetett értékkombinációja (pl. vehetjük az *a posteriori* legvalószínűbb értékeket)
- minden állapothoz hozzárendelünk egy értéket, *V(s)*, amely az állapot kívánatosságát kódolja
- minden döntés után, *t* időpillanatban, frissítjük az állapotértékek becslését a korábbi becslések és az aktuális jutalom alapján
- intuíció: egy állapot értékét meghatározza az abban kapható jutalom és a belőle kevés cselekvéssel elérhető további állapotok értéke



Russell & Norvig, 1995

$$V_{t+1}(s_t) = V_t(s_t) + \alpha[r_{t+1} + \gamma V_t(s_{t+1}) - V_t(s_t)]$$

A jutalom hosszútávú maximalizációja

- Összesített jutalom: valamilyen becslést kell alkalmazni a jövőre nézve
- Különbség a felügyelt tanulástól: egyiknél azt tudjuk, hogy a képtérben milyen messze vagyunk az optimálistól, a másiknál arra kapunk egy becslést, hogy a paramétertérben mennyire térünk el az optimálistól

Temporal Difference tanulás

- A predikciós hibából tanulunk
- Az aktuálisan meglátogatott állapot értéke az előzőleg meglátogatottét önmagához hasonlóbbá teszi
- Korábban meglátogatott állapotokra is visszaterjeszthetjük a változtatást

$$V_{t+1}(s_t) = V_t(s_t) + \epsilon \left[r_{t+1} + \gamma V_t(s_t + 1) - V_t(s_t) \right]$$

Modellalapú és modell nélküli RL

- Ha tudjuk az állapotok értékét, stratégiát kell alkotnunk a kiválasztandó cselekvésekre
  - Ehhez tudnunk kell, melyik cselekvés melyik állapotba visz át (milyen valószínűséggel)
  - Ezt is tanulhatjuk tapasztalatokból
  - Vagy tanulhatjuk közvetlenül állapot-cselekvés párok értékét
- Modell nélküli vagy Q-learning
  - több paramétere van, de gyorsabban tudunk számolni vele

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \epsilon \left[r_{t+1} + \gamma \max_a Q_t(s_t + 1, a) - Q_t(s_t, a_t) \right]$$

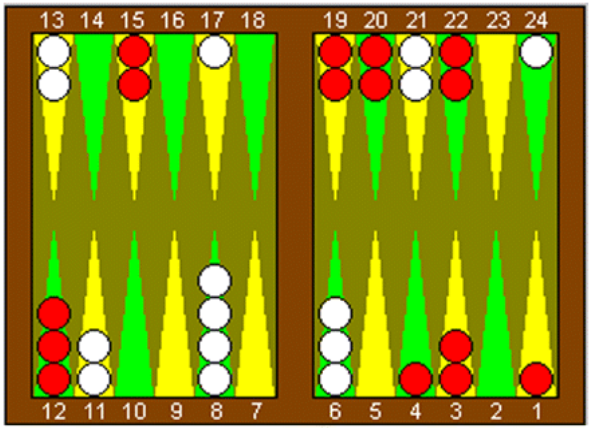
Exploration vs. exploitation

- Amikor még keveset tudunk a környezetről, nincs értelme az első jutalomhoz vezető cselekvéssort ismételni a végtelenségig
- Amikor többet tudunk, akkor viszont érdemes a lehető legjobb stratégiát alkalmazni folyamatosan
- Szokásos megoldás: próbálkozzunk véletlenszerűen az elején, és fokozatosan emeljük a legjobbnak becsült cselekvés választásának valószínűségét

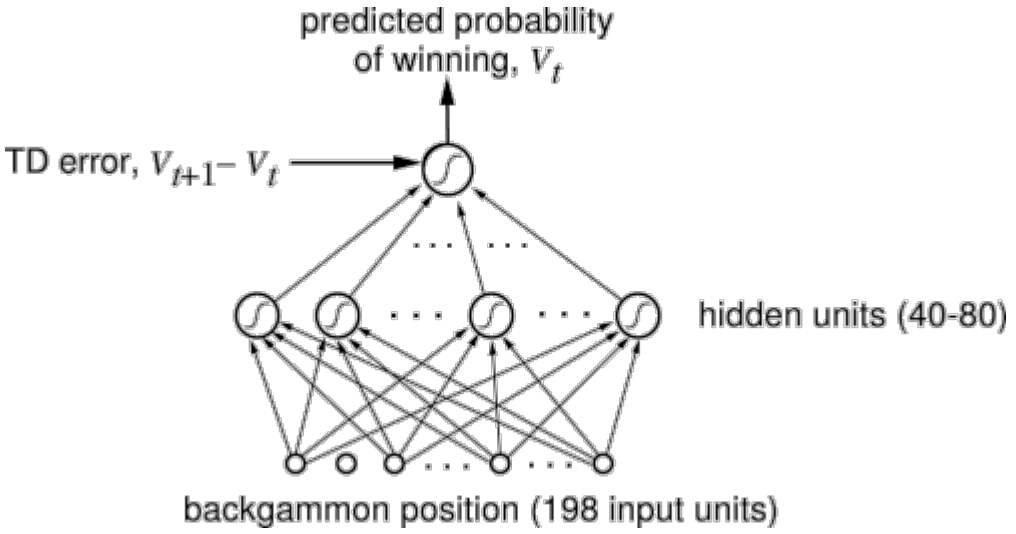
Az állapotérték-függvény reprezentációja

- Ha nincs olyan sok érték, használhatunk táblázatot
- Nagy illetve esetleg folytonos állapotterekben
  - csak az állapotváltozókat tudjuk közvetlenül reprezentálni, nem az összes lehetséges érték-kombinációt
  - általánosítanunk kell olyan állapotokra, amiket sosem látogattunk meg korábban
  - a többrétegű neurális hálózatok alkalmasak mindkét probléma megoldására
  - mivel ezek tanításához szükséges egy elvárt kimenet minden lépésben, ezt a TD algoritmus predikciós hibájából kell megalkotnunk

TD tanulás neurális hálózattal



- Gerald Tesauro: TD-Gammon
  - Előrecsatolt hálózat
  - Bemenet: a lehetséges lépések nyomán elért állapotok
  - Kimenet: állapotérték (nyerési valség)
- Minden lépésben meg kell határozni a hálózat kimeneti hibáját
  - Reward signal alapján
- Eredmény: a legjobb emberi játékosokkal összemérhető



TD neurális reprezentációval

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

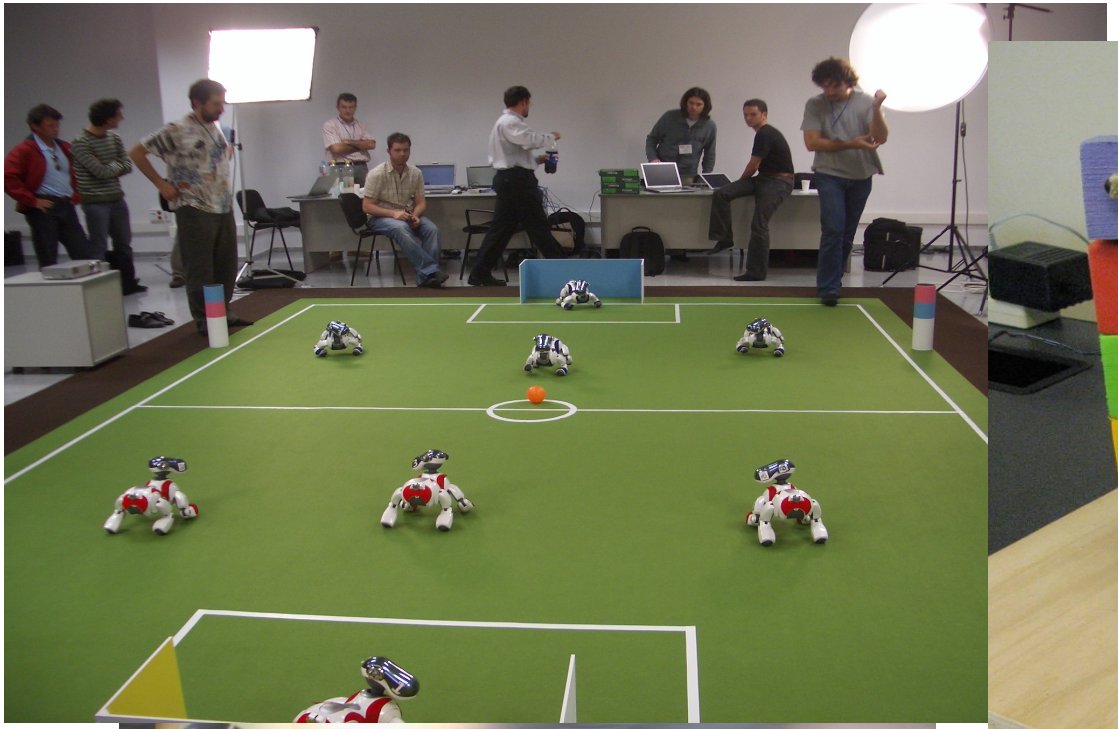
$$w(\tau) \leftarrow w(\tau) + \varepsilon\delta(t)u(t-\tau) \qquad \delta(t) = \sum_t r(t+\tau) - v(t)$$

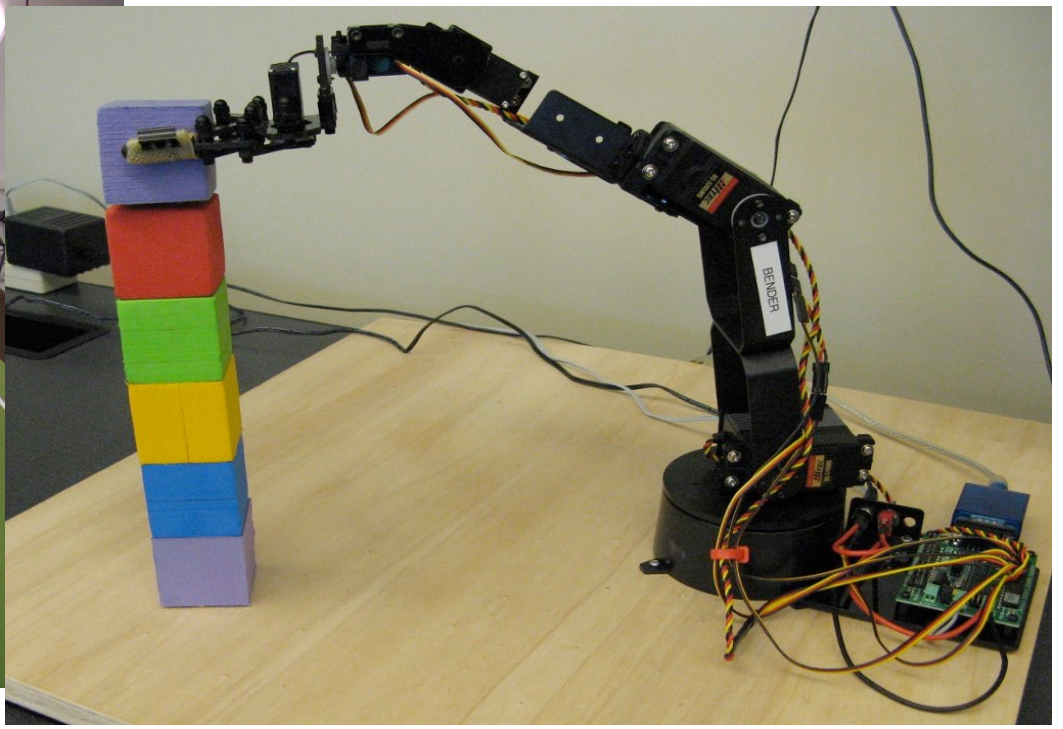
- A prediction error kiszámítása
  - A teljes jövőbeli jutalom kellene hozzá
  - Egylépéses lokális közelítést alkalmazunk

$$\sum_t r(t+\tau) - v(t) \approx r(t) + v(t+1)$$

  - Ha a környezet megfigyelhető, akkor az optimális stratégiához konvergál
- A hibát visszaterjeszthetjük a korábbi állapotokra is

Fizikai mozgás

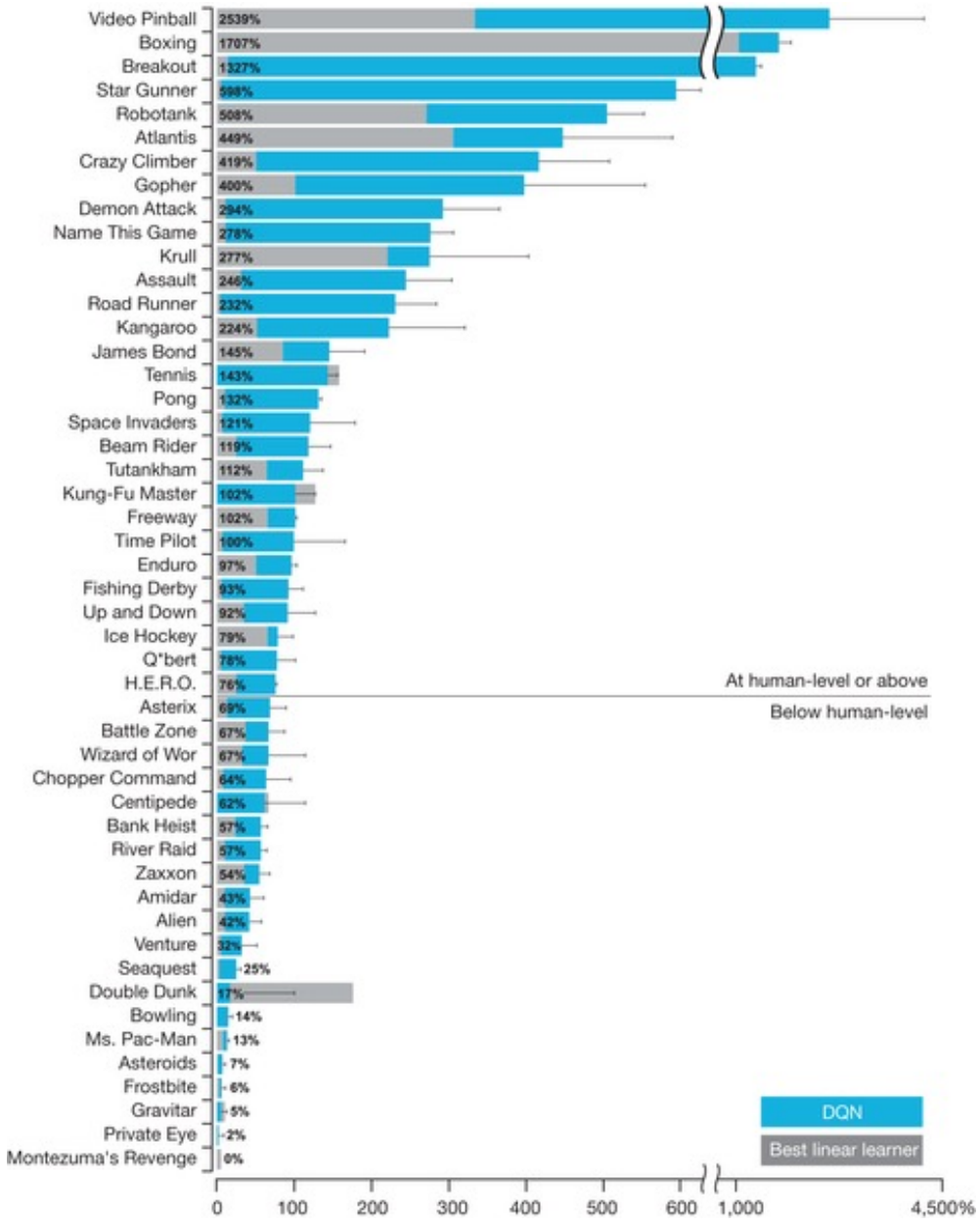


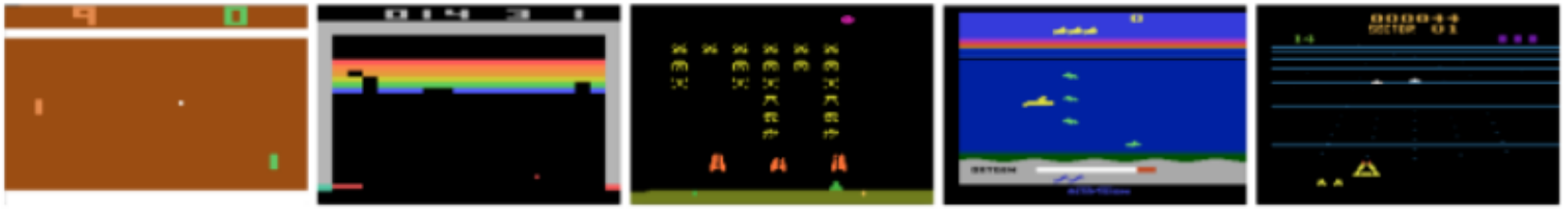


Számítógépes játékok tanulása

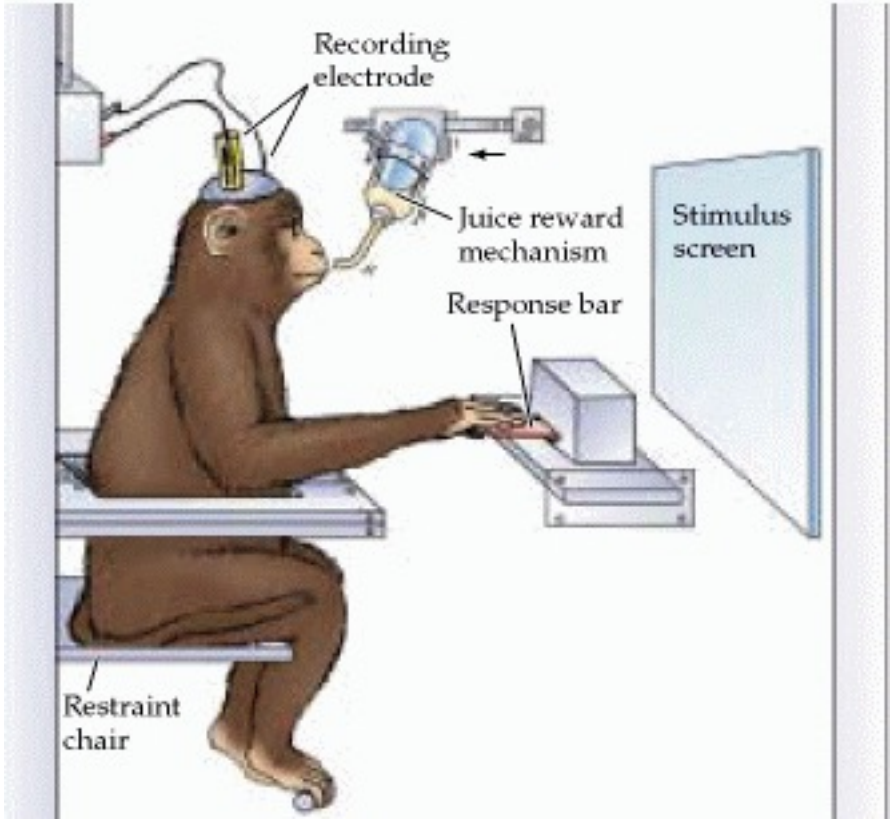
- reinforcement learning + deep networks
- megfigyelés: a képernyő pixelei + pontszám

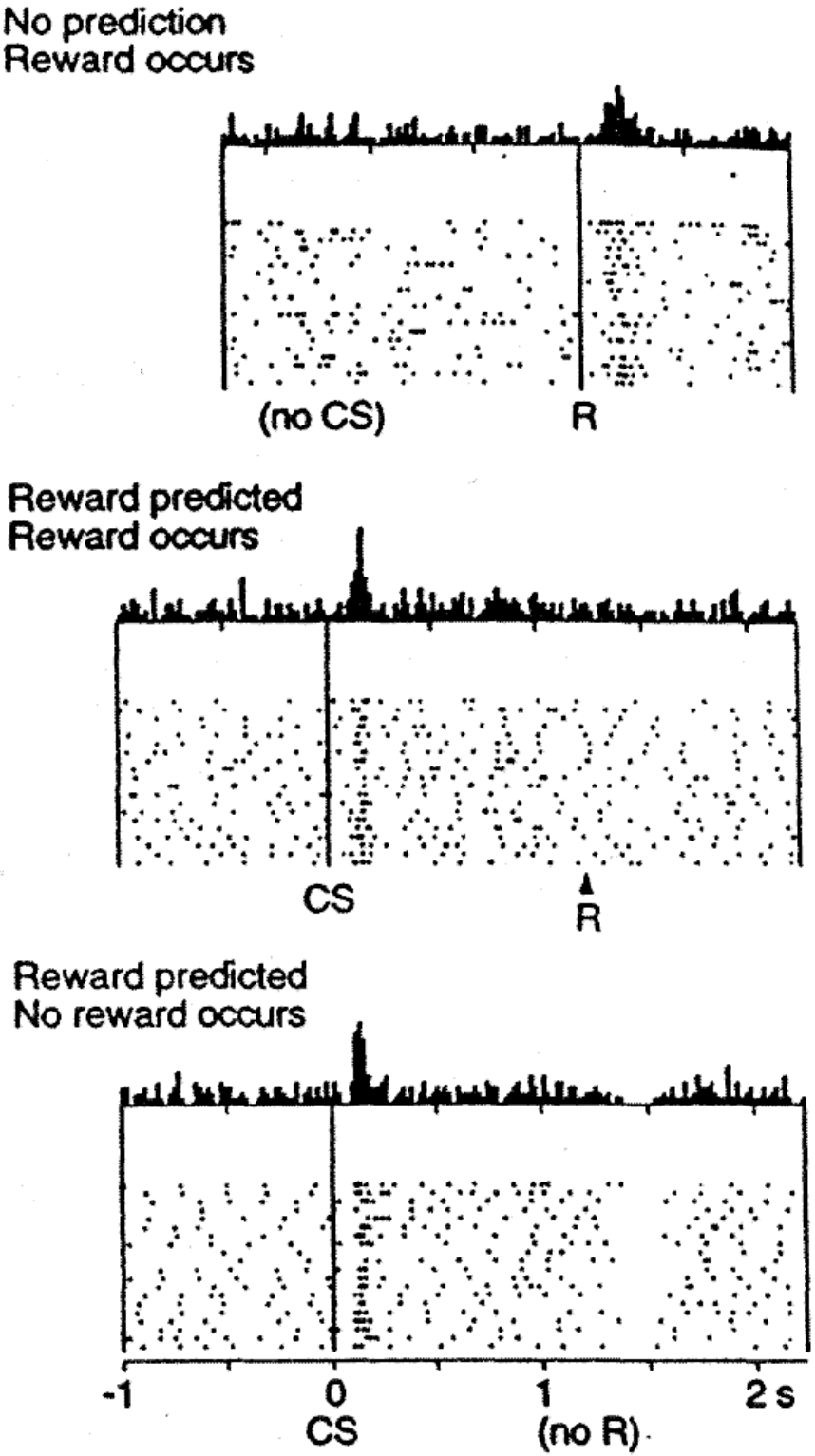
Mnih et al, 2015



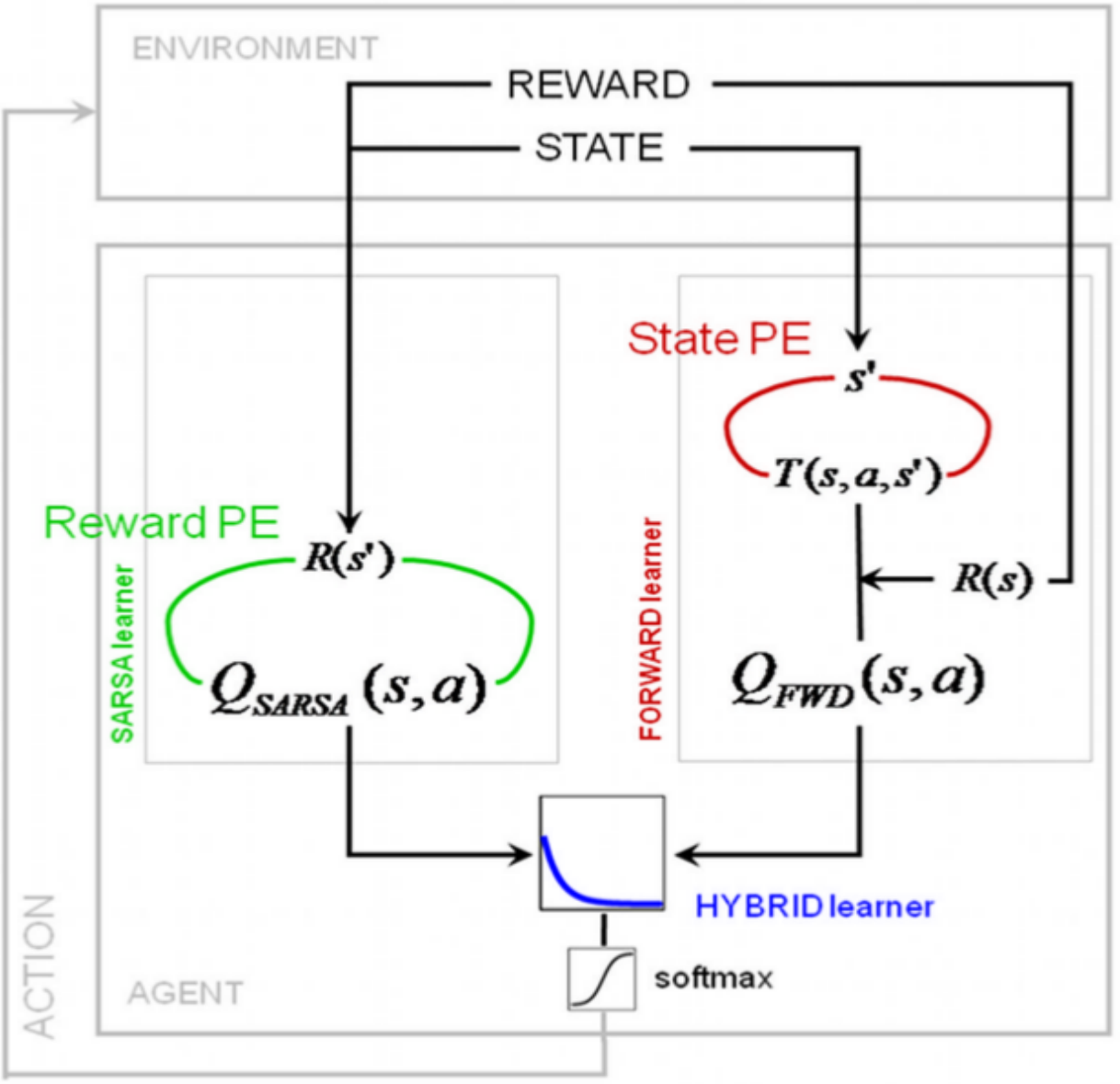


A jutalom reprezentációja a agyban

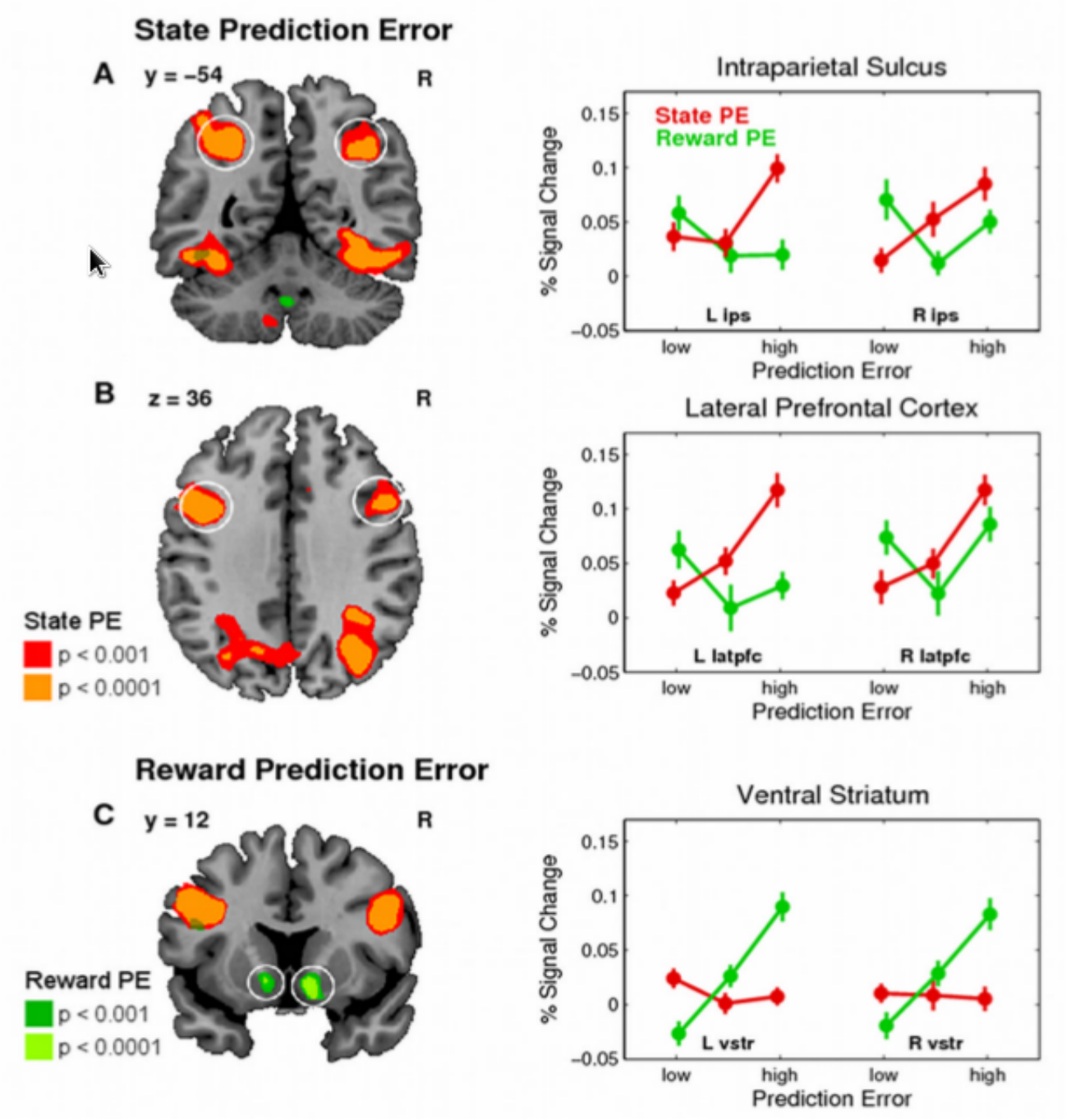




- Glascher, Daw Dayan, O'Doherty, *Neuron,* 2010.
- Modellalapú és modell nélküli algoritmusok predikciós hibájájával való korreláció



- fMRI eredmények



HF

- Add meg a kincskeresős játékhoz tartozó állapot- és cselekvésteret
- Valósítsd meg valamilyen programnyelven az ágens és a környezet szimulált interakcióját
- Q-learning segítségével tanuld meg, melyik állapotban melyik cselekvés a leghasznosabb
- Eredményedet ábrákkal illusztráld