# Laborator IAut 5: Q-Learning

Andrei Olaru

19.03.2012

**Q-Learning**[1] este o tehnica de invatare automata care asociaza o utilitate pentru fiecare **pereche stare-actiune**. Elementele de baza sunt aceleasi ca si la laboratorul trecut: **agent, stare, actiune, recompensa**. La orice moment, agentul se afla intr-o anumita stare si decide asupra uneia dintre mai multe actiuni; pentru actiunea sa, agentul primeste o recompensa. Scopul agentului este de a obtine o recompensa totala maxima.

Agentul lucreaza cu o functie de **calitate** (quality), pe care si-o adapteaza pe masura ce exploreaza mediul: $Q : S \times A \to \mathbb{R}$. Actualizarea $Q$ se face dupa alegerea unei actiuni $a_t$ in starea $s_t$, care duce agentul in starea $s_{t+1}$, tinand cont de fosta valoare a lui $Q$, $\alpha$ – rata de invatare, $\gamma$ – factorul de atenuare, $R_{t+1}$ – recompensa primita dupa realizarea actiunii $a_t$, si valorile $Q$ pentru starea $s_{t+1}$ si actiunile posibile $a_{t+1}$ din $s_{t+1}$:

$$Q(s_t, a_t) = (1 - \alpha) \cdot Q(s_t, a_t) + \alpha \cdot (R_{t+1} + \gamma \cdot \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}))$$

**Problema:** Avem urmatorul joc, jucat intr-un grid de $3 \times 3$: un agresor si un aparator se pot misca orizontal pe liniile de sus si, respectiv, de jos ale gridului. La fiecare pas, bazat pe situatia curenta, agresorul si aparatorul decid asupra unei actiuni, dintre *Stanga*, *Dreapta* si *Trage*.

| | | |
|---|---|---|
| – | – | **A** |
| = | = | = |
| – | **d** | – |

La un pas de timp se poate realiza o singura actiune. Daca agresorul trage si aparatorul este pe aceeasi coloana cu agresorul, aparatorul pierde. Daca aparatorul trage si agresorul este pe aceeasi coloana dar nu a tras, aparatorul castiga. Strategia agresorului este urmatoarea: daca aparatorul este pe aceeasi coloana, trage; altfel, se misca spre coloana aparatorului. Aparatorul nu cunoaste strategia agresorului, si trebuie sa joace jocuri succesive pentru a invata, prin Q-learning, cum sa invinga pe agresor. Starea initiala a jocului este cu agresorul si aparatorul pe coloana din mijloc.

**Bonus:** introduceti urmatoarea regula in joc: pentru a ataca, inainte de a trage, aparatorul trebuie sa avanseze o casuta (sa ajunga pe randul de mijloc), realizand actiunea *Avans*. Actiunea *Avans* poate fi urmata doar de *Trage* sau *Retragere* – aparatorul nu se poate misca pe orizontala cat timp este pe randul din mijloc.

**Ciclu program:** jocul se afla in starea $s_t$; alege $a_t$ cu $Q(s_t, a_t)$ maxim; alege mutarea agresorului; realizeaza cele doua mutari simultan, jocul ajunge in starea $s_{t+1}$, cu recompensa $R_{t+1}$; actualizeaza $Q(s_t, a_t)$; $t++$.

**Nota:** Reprezentarea starii este la alegerea voastra. Cuantumul recompenselor este de asemenea la alegere, dar recompensa trebuie sa fie nenula doar la sfarsitul unui joc.

**Alte resurse**

Curs IAut 4: http://cs.curs.pub.ro/2011/mod/resource/view.php?id=4458
Tutorial Q-learning (**Nota:** randurile indentate sunt linkuri):
http://people.revoledu.com/kardi/tutorial/ReinforcementLearning/index.html

[1] http://en.wikipedia.org/wiki/Q_learning