Reinforcement Learning

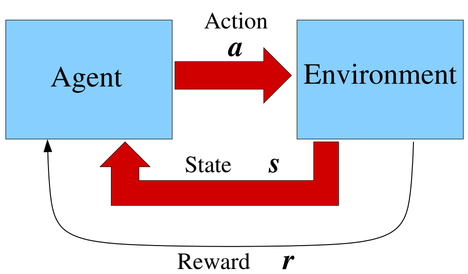
Reinforcement Learning

Learning of a behavior without
explicit information about correct actions

- A reward gives information about success
- The reward does not necessarily arrive *when* you do something good
  Temporal credit assignment
- The reward does not say *what* was good
  Structural credit assignment

Model of the learning situation

- An agent interacts with its environment
- The agent makes actions
- Actions affect the environments state
- The agent can observe the environments state
- The agent receives reward from the environment



Task for the Agent

Find a behavior which maximizes the expected *total* revard.

How long future should we consider?

- Finite Horizon

$$\max \left[\sum_{t=0}^{h} r_t \right]$$

- Infinite Horizon

$$\max \left[\sum_{t=0}^{\infty} \gamma^t r_t \right]$$

Requires discount of future reward ($0 < \gamma < 1$)

Reward Function

The Reward function controls which task should be solved

- Game (Chess, Backgammon)
  Reward only at the end: $+1$ when winning, -1 when loosing
- Avoiding mistakes (cycling, balancing, ...)
  Reward -1 at the end (when failing)
- Find a short/fast/cheap path to a goal
  Reward -1 at each step

Simplifying Assumptions

- Discrete time
- Finite number of actions a_i

$$a_i \in a_1, a_2, a_3, \ldots, a_n$$

- Finite number of states s_i

$$s_i \in s_1, s_2, s_3, \ldots, s_m$$

- Environment is a stationary *Markov Decision Process*
  Reward and next state depends only on s, a and chance
- Deterministic or non-deterministic environment

The Agents Internal Representation

- *Policy*
  The action chosen by the agent for each state

$$\pi(s) \mapsto a$$

- *Value Function*
  Expected total future reward from s when following policy π

$$V^{\pi}(s) \mapsto \Re$$

Classical model problem: *Grid World*

- Each state is represented by a position in a grid
- The agent acts by moving to other positions

G			
			G

Trivial labyrinth

Reward: -1 at each step until a goal state (G) is reached

The values of a state depends on the current policy.

0	−1	−2	−3
−1	−2	−3	−2
−2	−3	−2	−1
−3	−2	−1	0

V with an optimal policy

0	−14	−20	−22
−14	−18	−22	−20
−20	−22	−18	−14
−22	−20	−14	0

V with a random policy