Outline

Autonomous learning agents

Reinforcement learning

Copyright Tom Mitchell, 1994

Q Learning

Learn evaluation function over state x action

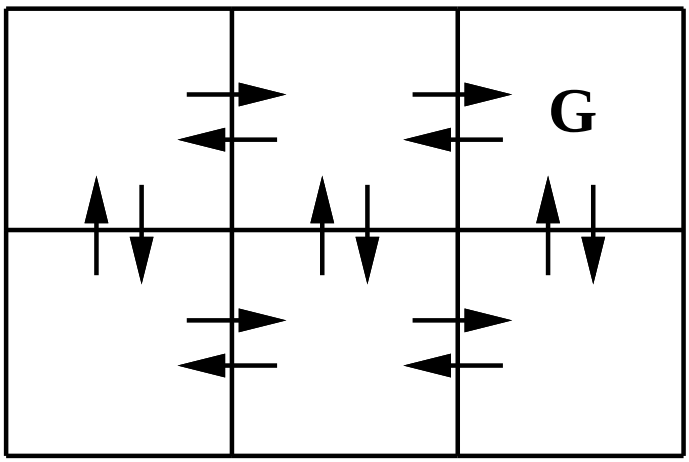
Requires no prior knowledge

Guaranteed convergence to optimal strategy under (unrealistic) assumptions

Idea: Learn approximation to Q(s,a)

Copyright Tom Mitchell, 1994

Example: Grid World



Reward:

- +100 for reaching goal state **G**
- 0 otherwise

What is Q(s,a)?

Copyright Tom Mitchell, 1994

Q Learning Algorithm

Q' is agent's estimate of the true Q function

Initialize Q'(s,a) to zero for all s,a

Observe initial state ($s_t \leftarrow$ observed state)

Do forever

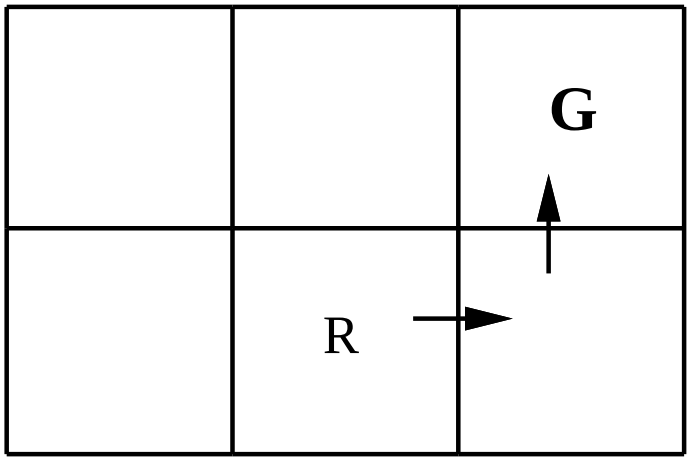
1. Select an action a_t, according to Prob(a)
   - $Prob(a) \leftarrow e^{Q'(s_t,a)/T} / \sum_i e^{Q'(s_t,a_i/T)}$
2. Execute action a_t
3. Observe resulting reward and state
   - $s_{t-1} \leftarrow s_t$
   - $r_{t-1} \leftarrow$ observed reward
   - $s_t \leftarrow$ observed new state
4. Update Q'
   a. Estimate training value for Q' according to $Q_{target}(s_{t-1},a) \leftarrow r_{t-1} + \gamma \, MAX_{a_i} \, Q'(s_t,a_i)$
   b. Update Q' using the training example s_{t-1}, $Q_{target}(s_{t-1},a)$

Copyright Tom Mitchell, 1994

Example Trace

Assume rote learning:

- store table of s, a, Q
- if multiple entries, take mean of current and new training value



Copyright Tom Mitchell, 1994