3/3/10 - CSE 150

Review

* Reinforcement Learning

state s_t → agent → action a_t → environment → reward r_t

state s_t
reward r_t

* Markov decision process (MDP)

$$\{\mathcal{S}, \mathcal{A}, P(s'|s,a), R(s)\}$$

states, actions, transitions, rewards

* Policy: deterministic mapping $\pi(s) \in \mathcal{A}$
* State value function

$$V^\pi(s) = E^\pi\Big[\underbrace{\sum_{t=0}^{\infty} \gamma^t R(s_t)}_{\text{discounted return}} \Big| s_t = s\Big]$$

* Bellman equation

$$V^\pi(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi(s))\, V^\pi(s')$$

* Action value function

$$Q^\pi(s,a) = E^\pi\Big[\sum_{t=0}^{\infty} \gamma^t R(s_t) \,\Big|\, s_0 = s, a_0 = a\Big]$$

* Optimality

Thm: there is always atleast one policy π^* for which
$$V^{\pi^*}(s) \geq V^\pi(s) \text{ for all } s, \pi$$
Proof: by construction; see text

* Optimal state value function

$$V^*(s) = V^{\pi^*}(s)$$

* Optimal action value function

$$Q^*(s,a) = Q^{\pi^*}(s,a)$$

There may be multiple optimal policies, but optimal value functions are unique.

* Relations

Given MDP, easy to write $V^*(s)$ and $Q^*(s,a)$ in terms of $\pi^*(s)$. Vice versa?

$$\pi^*(s) = \underset{a}{\operatorname{argmax}} \left[Q^*(s,a)\right]$$
$$= \underset{a}{\operatorname{argmax}} \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s')\right]$$
$$= \underset{a}{\operatorname{argmax}} \left[\sum_{s'} P(s'|s,a)\, V^*(s')\right]$$

$\boxed{\text{Planning}}$

Assume complete model of environment as

$$MDP = \{\mathcal{S}, \mathcal{A}, P(s'|s,a), R(s)\},$$ also $\gamma < 1$,

how to compute $\pi^*(s)$, or equivalently, $V^*(s)$ or $Q^*(s,a)$?

1) <u>Policy evaluation</u>

How to compute $V^\pi(s)$?

From Bellman equation:

$$V^\pi(s) = R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^\pi(s') \quad \text{for } s = 1, 2, 3, \ldots, N$$

$N = \#$ states in MDP

This is a system of N linear equations for N unknowns.

Put all unknowns on LHS:

$$V^\pi(s) - \gamma \sum_{s'} P(s'|s,\pi(s))\, V^\pi(s') = R(s)$$
$$\sum_{s'} \left\{\left[I(s,s') - \gamma P(s'|s,\pi(s))\right] V^\pi(s')\right\} = R(s)$$

$\uparrow$ indicator function

can write above equation as:

$$\underbrace{(I - \gamma P)}_{\text{known } N \times N \text{ matrix}} \underset{\text{unknown } n\times 1 \text{ vector}}{V} = R \leftarrow \text{known } n\times 1 \text{ vector}$$

Solution: $V^{\pi} = (I - \gamma P^{\pi})^{-1} R$

Matrix inversion is $O(N^3)$ operation

Ex: states $s \in \{0, 1\}$

transitions $P^{\pi}(s' \mid s, \pi(s)) = \begin{pmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{pmatrix}$

rewards $R(s) = \begin{pmatrix} r_0 \\ r_1 \end{pmatrix}$

value function $V^{\pi}(s) = \begin{pmatrix} V_0 \\ V_1 \end{pmatrix}$

solve:

$$\left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \gamma \begin{pmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{pmatrix} \right] \begin{pmatrix} V_0 \\ V_1 \end{pmatrix} = \begin{pmatrix} r_0 \\ r_1 \end{pmatrix}$$

2 equations for 2 unknowns

2) Policy Improvement

* How to compute π' such that $V^{\pi'}(s) \geq V^{\pi}(s)$ for all states s?
* Recall $Q^{\pi}(s,a)$ = expected return from state s, follow action a, then follow policy π.

How to compute $Q^{\pi}(s,a)$?

Evaluate policy π to compute $V^{\pi}(s)$.

Then: $Q^{\pi}(s,a) = R(s) + \gamma \sum_{s'} P(s' \mid s,a) V^{\pi}(s')$

* Define "greedy" policy:

$$\pi'(s) = \underset{a}{\operatorname{argmax}} \left[Q^{\pi}(s,a) \right] = \underset{a}{\operatorname{argmax}} \left[\sum_{s'} P(s' \mid s,a) V^{\pi}(s') \right]$$

* Theorem: greedy policy π' everywhere performs better or equal to original policy π

$$V^{\pi'}(s) \geq V^{\pi}(s) \quad \text{for all } s$$

Intuition: if better to choose action a in state s, then follow π, it's always better to choose action a.

Proof:
$$\begin{aligned} V^{\pi}(s) &= Q^{\pi}(s, \pi(s)) \\ &\leq \max_a Q^{\pi}(s,a) \\ &= Q^{\pi}(s, \pi'(s)) \\ &= R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s))\, V^{\pi}(s') \end{aligned}$$

So far: better to take one step under π', then revert to π, than to follow π.

"One-step" inequality: $V^{\pi}(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s))\, V^{\pi}(s')$

Apply one-step inequality on the RHS:

$$V^{\pi}(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s', \pi'(s'))\, V^{\pi}(s'') \right]$$

Better to take two steps under π', then revert to π, than to always follow π.

Apply "one-step" inequality t times:

Better to take $t+1$ steps under π', then revert to π, than to always follow π.

Let $t \to \infty$: it's always better to follow π' than π

$\Rightarrow V^{\pi}(s) \leq V^{\pi'}(s)$ since RHS converges to $V^{\pi'}(s')$ for $\gamma < 1$.

3) Policy Iteration

How to compute π^*?

Algorithm:

(1) initialize policy at random

(2) repeat until convergence

- compute state & action value functions of current policy
- derive greedy policy from action value function

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_1}(s) \\ Q^{\pi_1}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_2 \longrightarrow \cdots$$

* Is this guaranteed to converge?

Cannot cycle because $V^{\pi'}(s) \geq V^{\pi}(s)$ for all states s

Cannot go ~~on~~ forever because # policies is finite: $|A|^{|S|}$

Policy cannot be indefinitely improved.

Typicall converges in far less steps than $|A|^{|S|}$.

* Does it always converge to an optimal policy π^*? Yes.

Proof: later (or see text).