EL2805, Exercise 4 (Learning MDPs)

Robert Mattila

Today's central concepts

Previously, we have assumed that we could derive a model as well as obtain numerical values for the parameters.
What if everything is unknown?
Learn by interacting with the system!

(*): (or several)

In Monte-Carlo methods:

i, a(*) trajectory is generated under some policy,

ii, new estimates are computed and the policy is updated,

iii, repeat to i,.

We can focus to estimate different quantities:

$$\begin{matrix} p(s'|s,a), \\ r(s,a) \end{matrix} \longrightarrow q(s,a) \longrightarrow \pi(s)$$

(i) (ii) (iii)

(i): Directly compute $\hat{p}(s'|s,a)$ and $\hat{r}(s,a)$, then use standard DP to compute policy.

- Contain much information regarding the system
- Need to estimate $|S|^2 \times |A| + |S| \times |A|$ numbers.

(ii): The state-action function $q(s,a)$ is enough if we only want to control the system

- $|\mathcal{S}| \times |\mathcal{A}|$ numbers to estimate
- Idea: Keep track of rewards (to go) observed in (state, action)-pairs.

(iii): Actually, if our goal is to control the system, we could directly focus on $\pi(s)$. We'll do it in the next session (policy gradient).

In online/incremental/TD learning methods, the estimates as well as the policy are updated in each time-step. (We don't wait for an episode to finish.)

Note: Some problems don't even have episodes!

- Q-learning:
  i, Pick an action a (e.g. ε-greedy)
  ii, Apply a in s and observe $r(s,a)$ and s'
  iii, Update

$$\underbrace{q(s,a)}_{\text{reward-to-go from } s \text{ using } a} \leftarrow q(s,a) + \alpha \left[r(s,a) + \lambda \max_{a'} q(s',a') - q(s,a) \right] =$$

$$(1-\alpha) \underbrace{q(s,a)}_{\text{old estimate}} + \alpha \Big[\underbrace{r(s,a)}_{\text{new information, immediate reward}} + \underbrace{\lambda \max_{a'} q(s',a')}_{\text{assume greedy action in next time-step}} \Big]$$

iv, Repeat to i, .

Under suitable assumptions. See lecture.

Note:

- $q(s,a)$ will converge to $q^*(s,a)$.
- The actual action we implement in the next time-step is not necessarily the greedy!
- off-policy algorithm since we learn value of π^*, but that's not the policy we apply to the system.

- SARSA:

i, Take action a and observe $r(s,a)$ and s'.

ii, Pick an action a' (from s') (e.g. ε-greedy).

iii, Update

$$q(s,a) \leftarrow q(s,a) + \alpha\,[\,r(s,a) + \lambda \underbrace{q(s',a')}_{\text{(estimated) reward-to-go from the next time-step}} - q(s,a)\,].$$

iv, Set $a \leftarrow a'$ and repeat from i, .

Note:

- SARSA learns the optimal policy taking into account how we actually select actions (that we explore).
- On-policy since we learn value of the policy we implement.

Ex. 4.3

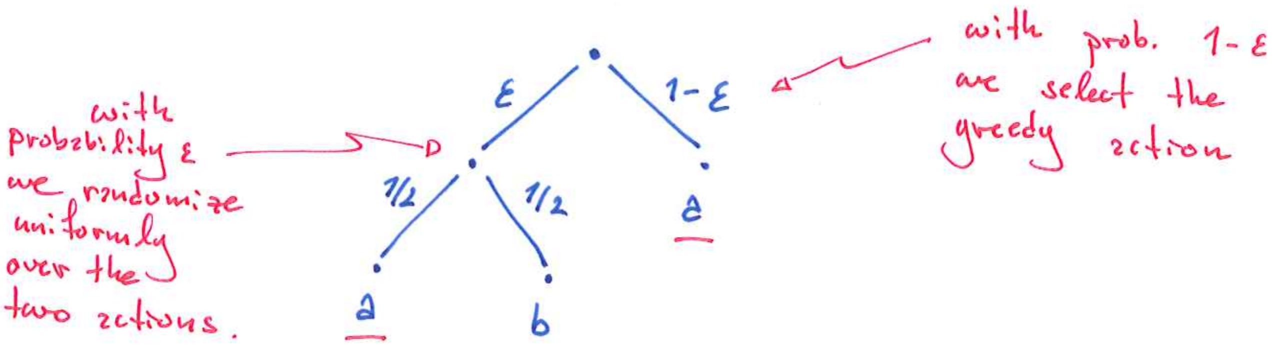
In ε-greedy action selection, what is the probability that the greedy action is selected if:

a, $\varepsilon = 0.5$ and $|\mathcal{A}| = 2$?

b, $\varepsilon = 1/5$ and $|\mathcal{A}| = 3$?

Solution:

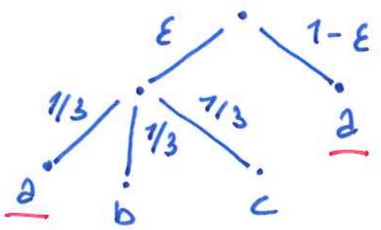
a, Assume $\mathcal{A} = \{a, b\}$ and a is the greedy action.



The total greedy action can be reached with probability:

$$\varepsilon \cdot \frac{1}{2} + (1-\varepsilon) = \{\varepsilon = 1/2\} = \frac{1}{4} + 1 - \frac{1}{2} = \frac{3}{4}.$$

b, Assume $\mathcal{A} = \{a, b, c\}$ and that a is greedy:



The total probability is now: $\varepsilon \cdot \frac{1}{3} + (1-\varepsilon) = \{\varepsilon = 1/5\} = \frac{1}{5} \cdot \frac{1}{3} + \frac{4}{5} = \frac{13}{15}.$

Ex. 4.6

Consider the following observed trajectory:

time	current state	reward	action	next state
1	s_1	-2	a_1	s_3
2	s_3	6	a_1	s_3
3	s_3	4	a_2	s_2
4	s_2	-2	a_1	s_2
5	s_2	2	a_2	s_1

a, Perform Q-learning with $\gamma = 0.8$ and $\alpha = 0.5$.

b, What is the system's optimal policy, assuming that the algorithm has converged after these five steps? (Note: it has not.)

Solution:

a, Recall the Q-learning update equation:

$$q(s,a) \leftarrow q(s,a) + \alpha\left[r(s,a) + \gamma \max_{a'} q(s',a') - q(s,a)\right].$$

Assume we initialize the q-table with zeroes.

Then:

	a_1	a_2
s_1	$\boxed{0}$	0
s_2	0	0
s_3	0	0

$$q(s,a) = q(s_1,a_1) \longleftarrow$$

$$q(s,a) + \alpha\left[r(s,a) + \gamma \max_{a'} q(s',a') - q(s,a)\right] =$$

$$\underbrace{q(s_1,a_1)}_{=0} + \alpha\left[\underbrace{r(s_1,a_1)}_{=-2} + \gamma \max_{a'} \underbrace{q(s_3,a')}_{\substack{=0 \\ \text{for all } a'}} - \underbrace{q(s_1,a_1)}_{=0}\right] =$$

$s = s_1$
$a = a_1$
$r = -2$
$s' = s_3$

$$0 + 0.5\left[-2 + 0 - 0\right] = -1.$$

	a_1	a_2
s_1	-1	0
s_2	0	0
s_3	$\boxed{0}$	0

$s = s_3$
$a = a_1$
$r = 6$
$s' = s_3$

$$q(s_3, a_1) \longleftarrow$$

$$\underbrace{q(s_3, a_1)}_{=0} + \alpha\Big[\underbrace{r(s_3, a_1)}_{=6} + \gamma \max_{a'} \underbrace{q(s_3, a')}_{=0 \text{ for all } a'} - \underbrace{q(s_3, a_1)}_{=0}\Big] =$$

$$0.5 \cdot 6 = 3.$$

	a_1	a_2
s_1	-1	0
s_2	0	0
s_3	3	$\boxed{0}$

$s = s_3$
$a = a_2$
$r = 4$
$s' = s_2$

$$q(s_3, a_2) \longleftarrow$$

$$\underbrace{q(s_3, a_2)}_{=0} + \alpha\Big[\underbrace{r(s_3, a_2)}_{=4} + \gamma \max_{a'} \underbrace{q(s_2, a')}_{=0 \text{ for all } a'} - \underbrace{q(s_3, a_2)}_{=0}\Big] =$$

$$0.5 \times 4 = 2$$

	a_1	a_2
s_1	-1	0
s_2	$\boxed{0}$	0
s_3	3	2

$s = s_2$
$a = a_1$
$r = -2$
$s' = s_2$

$$q(s_2, a_1) \longleftarrow$$

$$\underbrace{q(s_2, a_1)}_{=0} + \alpha\Big[\underbrace{r(s_2, a_1)}_{=-2} + \gamma \max_{a'} \underbrace{q(s_2, a')}_{=0 \text{ for all } a'} - \underbrace{q(s_2, a_1)}_{=0}\Big] =$$

$$0.5 \cdot (-2) = -1$$

	a_1	a_2
s_1	-1	0
s_2	-1	$\boxed{0}$
s_3	3	2

$s = s_2$
$a = a_2$
$r = 2$
$s' = s_1$

$$q(s_2, a_2) \longleftarrow$$

$$\underbrace{q(s_2, a_2)}_{=0} + \alpha\Big[\underbrace{r(s_2, a_2)}_{=2} + \gamma \max_{a'} q(s_1, a') - \underbrace{q(s_2, a_2)}_{=0}\Big] =$$

$$0 + \alpha\big[2 + \gamma \max\{\overset{a_1}{-1}, \overset{a_2}{0}\} - 0\big] =$$

$$0.5 \cdot 2 = 1$$

	a_1	a_2
s_1	-1	0
s_2	-1	1
s_3	3	2

Hence, our q-table after five steps is:

$$q(s,a): \quad \begin{array}{c|cc} & a_1 & a_2 \\ \hline s_1 & -1 & 0 \\ s_2 & -1 & 1 \\ s_3 & 3 & 2 \end{array} \; .$$

b, This would correspond to an optimal policy:

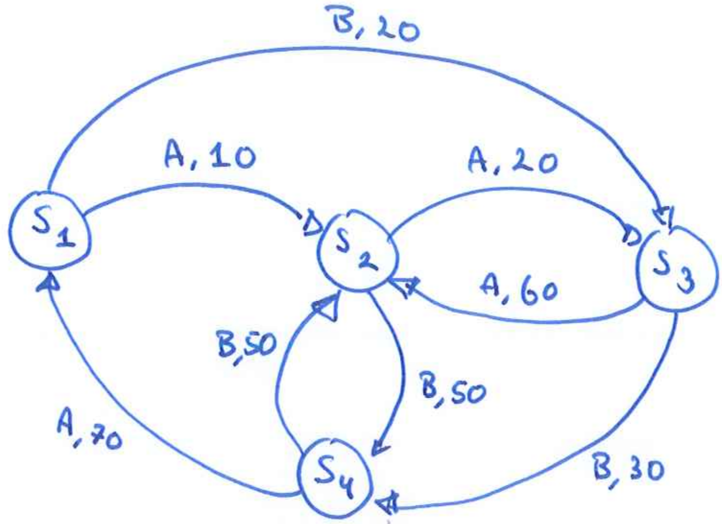
$$a^*(s_1) = \arg\max_{a \in \mathcal{A}} q(s_1, a) = a_2 ,$$

$$a^*(s_2) = \arg\max_{a \in \mathcal{A}} q(s_2, a) = a_2 ,$$

$$a^*(s_3) = \arg\max_{a \in \mathcal{A}} q(s_3, a) = a_1 .$$

(if we had converged after these iterations)

Ex. 4.7



Consider the deterministic system above with $\mathcal{S} = \{s_1, s_2, s_3, s_4\}$, $\mathcal{A} = \{A, B\}$ and reward and transitions according to the graph.

Perform Q-learning with $\lambda = 0.9$ and $\alpha = 1$ for the action-sequence $\{A, A, B, A, B, A\}$ initialized in s_1.

Solution:

Recall the update equation for Q-learning:

$$q(s,a) \leftarrow q(s,a) + \alpha \left[r(s,a) + \lambda \max_{a'} q(s',a') - q(s,a) \right].$$

Assume we initialize $q(s,a)$ with zeros:

q:

	A	B
s_1	$\boxed{0}$	0
s_2	0	0
s_3	0	0
s_4	0	0

$s = s_1$
$a = A$
$s' = s_2$
$r = 10$

• $q(s,a) = q(s_1, A) \longleftarrow$

$$q(s,a) + \alpha \left[r(s,a) + \lambda \max_{a'} q(s',a') - q(s,a) \right] =$$

$$\underbrace{q(s_1,A)}_{=0} + \alpha \left[\underbrace{r(s_1,A)}_{=10} + \lambda \max_{a'} \underbrace{q(s_2,a')}_{\substack{=0 \\ \text{for all } a'}} - \underbrace{q(s_1,A)}_{=0} \right] =$$

$$0 + 1 \left[10 + 0.9 \cdot 0 - 0 \right] = 10$$

	A	B
s_1	10	0
s_2	$\boxed{0}$	0
s_3	0	0
s_4	0	0

- $q(s_2, A) \leftarrow$

$$\underbrace{q(s_2,A)}_{=0} + \alpha\left[\underbrace{r(s_2,A)}_{=20} + \gamma \max_{a'} \underbrace{q(s_3,a')}_{\substack{=0\\ \text{for all } a'}} - \underbrace{q(s_2,A)}_{=0}\right] =$$

$s = s_2$
$a = A$
$s' = s_3$
$r = 20$

$$0 + 1\left[20 + 0.9 \cdot 0 - 0\right] = 20$$

	A	B
s_1	10	0
s_2	20	0
s_3	0	$\boxed{0}$
s_4	0	0

- $q(s_3, B) \leftarrow$

$$\underbrace{q(s_3,B)}_{=0} + \alpha\left[\underbrace{r(s_3,B)}_{=30} + \gamma \max_{a'} \underbrace{q(s_4,a')}_{\substack{=0 \text{ for}\\ \text{all } a'}} - \underbrace{q(s_3,B)}_{=0}\right] =$$

$s = s_3$
$a = B$
$s' = s_4$
$r = 30$

$$0 + 1\left[30 + 0.9 \cdot 0 - 0\right] = 30$$

	A	B
s_1	10	0
s_2	20	0
s_3	0	30
s_4	$\boxed{0}$	0

- $q(s_4, A) \leftarrow$

$$\underbrace{q(s_4,A)}_{=0} + \alpha\left[\underbrace{r(s_4,A)}_{=70} + \gamma \max_{a'} q(s_1,a') - \underbrace{q(s_4,A)}_{=0}\right] =$$

$s = s_4$
$a = A$
$s' = s_1$
$r = 70$

$$0 + 1\left[70 + \gamma \max\{\overset{A}{10}, \overset{B}{0}\} - 0\right] =$$

$$70 + 0.9 \cdot 10 = 79$$

	A	B
s_1	10	$\boxed{0}$
s_2	20	0
s_3	0	30
s_4	79	0

- $q(s_1, B) \leftarrow$

$$\underbrace{q(s_1,B)}_{=0} + \alpha\left[\underbrace{r(s_1,B)}_{=20} + \gamma \max q(s_3,a') - \underbrace{q(s_1,B)}_{=0}\right] =$$

$s = s_1$
$a = B$
$s' = s_3$
$r = 20$

$$0 + 1\left[20 + \gamma \max\{\overset{A}{0}, \overset{B}{30}\} - 0\right] =$$

$$20 + 0.9 \cdot 30 = 47$$

	A	B
s_1	10	47
s_2	20	0
s_3	$\boxed{0}$	30
s_4	79	0

$$q(s_3, A) \longleftarrow$$

$$\underbrace{q(s_3,A)}_{=0} + \alpha \left[\underbrace{r(s_3,A)}_{=60} + \gamma \max_{a'} q(s_2, a') - \underbrace{q(s_3,A)}_{=0} \right] =$$

$s = s_3$
$a = A$
$s' = s_2$
$r = 60$

$$0 + 1 \left[60 + \gamma \max \{ \overset{A}{20}, \overset{B}{0} \} - 0 \right] =$$

$$60 + 0.9 \cdot 20 = 78$$

	A	B
s_1	10	47
s_2	20	0
s_3	78	30
s_4	79	0

Hence, our state-action function is

$q(s,a)$:

	A	B
s_1	10	47
s_2	20	0
s_3	78	30
s_4	79	0

,

which corresponds to the greedy policy:

$$a(s_1) = \arg\max_a q(s_1, a) = B,$$

$$a(s_2) = A,$$

$$a(s_3) = A,$$

$$a(s_4) = A.$$