Techniques of Artificial Intelligence

Exercises – WEKA & Reinforcement Learning

Dipankar Sengupta
Dipankar.Sengupta@vub.ac.be

Roxana Rădulescu
Roxana.Radulescu@vub.ac.be

May 2, 2016

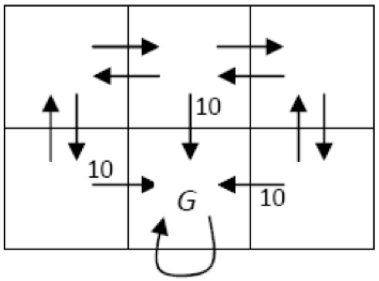
notation	meaning
$V^*(s)$	optimal state-value for state s
$V^\pi(s)$	state-value for state s in policy π
$Q^*(s,a)$	the Q-value for taking action a in state s for the optimal policy
$\hat{Q}(s,a)$	the agents approximation of $Q(s,a)$

35. **Knowledge flow interface**
Open the Knowledge flow interface.

Use an arffloader to configure the input dataset. Connect the loader to classassigner and select the attribute to be used. Make a connection with crossvalidationfoldmarker (10 folds). Make a training and test set connection with Naïve Bayes learner. Make a connection to Classifierperformanceevaluator and then connect to a textviewer. Describe and motivate the observed trend in the performance.

36. **MDP**
What is meant by *Markov* in the context of a *Markov Decision Process* (MDP) ?

37. **Reinforcement Learning** (exercise 13.2 in the course book)
Consider the deterministic grid world shown below with the absorbing goalstate G. Here the immediate rewards are 10 for the labelled transitions and 0 for all unlabelled transitions. Use $\gamma = 0.8$.



(a) Give the $Q^*(s,a)$ value for every transition .

(b) Give the V^* value for every state in this grid world.

(c) Show an optimal policy.

(d) Suggest a change to the reward function $r(s,a)$ that alters the $Q(s,a)$ values, but does not alter the optimal policy.

(e) Suggest a change to $r(s,a)$ that alters $Q(s,a)$ but does not alter $V^*(s)$.

(f) Now consider applying the Q function learning algorithm to this grid world, assuming the table of Q values is initialized to zero. Assume the agent begins in the bottom left grid square and then travels clockwise around the perimeter of the grid until it reaches the absorbing goal state, completing the first training episode. Describe which Q values are modified as a result of this episode, and give their revised values.

(g) Answer the question again assuming the agent now performs an identical episode.

(h) Answer the question again assuming the agent now performs an identical episode.

(i) Answer the question again assuming the agent now performs N identical episodes.

(j) Answer the question again assuming the agents starts in the left upper state, goes to the right and then goes down.

Answer: Use the formula $Q(s,a) = r(s,a) + \gamma max_{a'} Q(s',a')$

(a) Q(a1, right) = 8, Q(a1, down) = 8
Q(a2, lef t) = 6.4, Q(a2, down) = 10, Q(a2, right) = 6.4
Q(a3, lef t) = 8, Q(a3, down) = 8
Q(b1, up) = 6.4, Q(b1, right) = 10
Q(b2, loop) = 0
Q(b3, left) = 10, Q(b3, up) = 6.4

(b)

	1	2	3
a	8	10	8
b	10	G	10

(c)

	1	2	3
a	down	down	down
b	right	loop	left

(d) Set for example r(a2, down) = 9.

(e) Set for example r(b1, up) = 2.

(f) Q(b3, left) = 10 + 0.8 x 0 = 10

(g) Q(a3, down) = 0 + 0.8 x 10 = 8

(h) Q(a2, right) = 0 + 0.8 x 8 = 6.4

(i) Q(a1, right) = 0 + 0.8 x 6.4 = 5.12
Q(b1, up) = 0 + 0.8 x 5.12 = 4.096

(j) First episode: Q(a1, right) = 0 + 0.8 x 6.4 = 5.12 and Q(a2, down) = 10 + 0.8 x 0 = 10
Second episode: Q(a1, right) = 0 + 0.8 x 10 = 8 and Q(a2, down) = 10 + 0.8 x 0 = 10

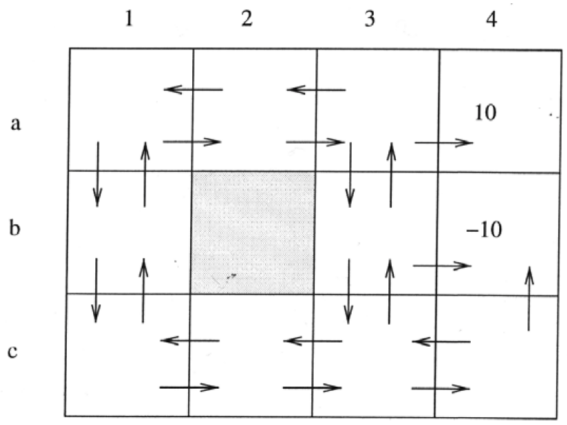
38. Simulate Q function learning for a robot walking around in the following environment. Indicate Q-values after the following episodes using the back-propagated Q update rule (as in the previous exercise). Thus, after getting in a goal state (b4 or a4), update the $\widehat{Q} - values$ in reverse order. The initial Q-value estimates are 0.0 and discount factor $\gamma = 0.9$.

(a) a1, a2, a3, b3, b4

(b) c2, c1, b1, a1, a2, a3, a4

(c) c4, c3, b3, a3, a4

Assume the robot will now use the policy of always performing the action having the greatest Q-value. Indicate this policy on the drawing. Is it optimal?

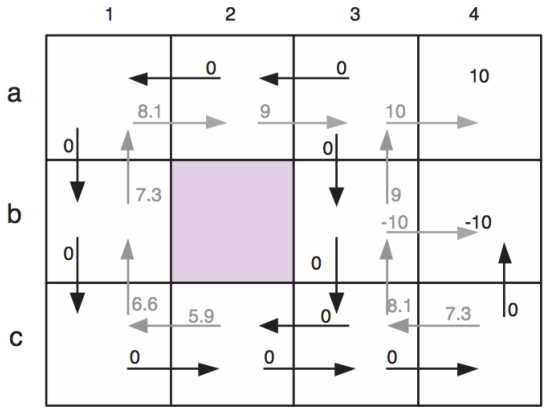


Answer: Use the formula $Q(s,a) = r(s,a) + \gamma max_{a'} Q(s',a')$. After taking action a in state s, the robot ends up in state s'. In the max() function below, I listed all Q-values (of the possible actions in state s') in the following order: up, right, down and left. Note, here, goal states have no Q-values.

(a) Q(b3, right) = –10 + 0.9 max() = –10
Q(a3, down) = 0 + 0.9 max(0, -10, 0) = 0
Q(a2, right) = 0 + 0.9 max(0, 0, 0) = 0
Q(a1, right) = 0 + 0.9 max(0, 0) = 0

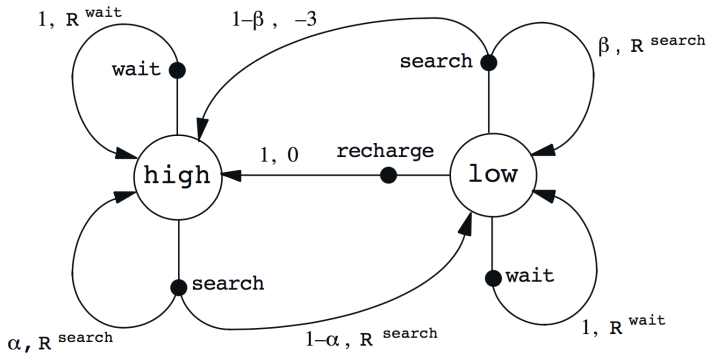
(b) Q(a3, right) = 10 + 0.9 max() = 10
Q(a2, right) = 0 + 0.9 max(10, 0, 0) = 9
Q(a1, right) = 0 + 0.9 max(9, 0) = 8.1
Q(b1, up) = 0 + 0.9 max(8.1, 0) = 7.29
Q(c1, up) = 0 + 0.9 max(7.29, 0) = 6.561
Q(c2, lef t) = 0 + 0.9 max(6.561, 0) = 5.9049

(c) Q(a3, right) = 10 + 0.9 max() = 10
Q(b3, up) = 0 + 0.9 max(10, 0, 0) = 9
Q(c3, up) = 0 + 0.9 max(9, –10, 0) = 8.1
Q(c4, lef t) = 0 + 0.9 max(8.1, 0, 0) = 7.29



39. Consider the stochastic decision process below. You may assume that it is Markovian. The states are *low* and *high*. The actions are shown as black spots connected to the states. The transition probabilities ($\alpha = \beta = \frac{2}{3}$) and expected rewards ($R^{wait} = 1$ and $R^{search} = 2$) are given along the arcs, pointing from the action to the next state. Use the TD update rule for the Q-learning algorithm. Use the learning rate $\alpha = 0.1$ and discount factor $\gamma = 0.9$. Simulate an agent making the following transitions:

high → search → high
high → search → high
high → search → low
low → wait → low
low → recharge → high



Answer: Use the formula: $\widehat{Q}(s,a) \leftarrow \widehat{Q}(s,a) + \alpha[r(s,a) + \gamma \cdot max_{a'}(\widehat{Q}(s',a')) - \widehat{Q}(s,a)]$

- $\widehat{Q}(h,s) \leftarrow \widehat{Q}(h,s)+0.1[2+0.9 \cdot max(\widehat{Q}(h,s),\widehat{Q}(h,w))-\widehat{Q}(h,s)] = 0+0.1(2+0.9 \cdot max(0,0)-0) = 0.2$
- $\widehat{Q}(h,s) \leftarrow \widehat{Q}(h,s)+0.1[2+0.9 \cdot max(\widehat{Q}(h,s),\widehat{Q}(h,w))-\widehat{Q}(h,s)] = 0.2+0.1(2+0.9 \cdot max(0.2,0)-0.2) = 0.398$
- $\widehat{Q}(h,s) \leftarrow \widehat{Q}(h,s) + 0.1[2 + 0.9 \cdot max(\widehat{Q}(l,s),\widehat{Q}(l,w),\widehat{Q}(l,r)) - \widehat{Q}(h,s)] = 0.398 + 0.1(2 + 0.9 \cdot max(0,0,0) - 0.398) = 0.55821$
- $\widehat{Q}(l,w) \leftarrow \widehat{Q}(l,w)+0.1[1+0.9 \cdot max(\widehat{Q}(l,s),\widehat{Q}(l,w),\widehat{Q}(l,r))-\widehat{Q}(l,w)] = 0+0.1(1+0.9 \cdot max(0,0,0)-0) = 0.1$
- $\widehat{Q}(l,r) \leftarrow \widehat{Q}(l,r)+0.1[0+0.9 \cdot max(\widehat{Q}(h,s),\widehat{Q}(h,w))-\widehat{Q}(l,r)] = 0+0.1(0+0.9 \cdot max(0.5582,0)-0) = 0.050238$