Dynamic Programming

练习测验, 14 个问题

14/14 分 (100%)

恭喜！您通过了！

下一项

1 / 1
分数

1.
The value of any state under an optimal policy is ___ the value of that state under a non-optimal policy. [Select all that apply]

- [x] Strictly greater than

未选择的是正确的

- [x] Greater than or equal to

正确
Correct! This follows from the policy improvement theorem.

- [x] Strictly less than

未选择的是正确的

- [x] Less than or equal to

未选择的是正确的

---

1 / 1
分数

2.
If a policy is greedy with respect to the value function for the equiprobable random policy, then it is **guaranteed** to be an optimal policy.

- ( ) True
- (x) False

Dynamic Programming

练习测验, 14 个问题

14/14 分 (100%)

正确
Correct! Only policies greedy with respect to the optimal value function are guaranteed to be optimal.

---

1 / 1
分数

3.
If a policy π is greedy with respect to its own value function v_π, then it is an optimal policy.

- True

正确
Correct! If a policy is greedy with respect to its own value function, it follows from the policy improvement theorem and the Bellman optimality equation that it must be an optimal policy.

- False

---

1 / 1
分数

4.
Let v_π be the state-value function for the policy π. Let π' be greedy with respect to v_π. Then $v_{\pi'} \geq v_\pi$.

- False
- True

正确
Correct! This is a consequence of the policy improvement theorem.

---

1 / 1
分数

5.
Let v_π be the state-value function for the policy π. Let $v_{\pi'}$ be the state-value function for the policy π'. Assume $v_\pi = v'_\pi$. Then this means that $\pi = \pi'$.

- False

正确
Correct! For example, two policies might share the same value function, but differ due to random tie breaking.

True

Dynamic Programming

练习测验, 14 个问题

14/14 分 (100%)

1 / 1
分数

6.
What is the relationship between value iteration and policy iteration? [Select all that apply]

- [x] Value iteration and policy iteration are both special cases of generalized policy iteration.

正确
Correct!

- [ ] Policy iteration is a special case of value iteration.

未选择的是正确的

- [ ] Value iteration is a special case of policy iteration.

未选择的是正确的

---

1 / 1
分数

7.
The word synchronous means "at the same time". The word asynchronous means "not at the same time". A dynamic programming algorithm is: [Select all that apply]

- [ ] Asynchronous, if it updates some states more than others.

正确
Correct! Only algorithms that update every state exactly once at each iteration are synchronous.

- [ ] Asynchronous, if it does not update all states at each iteration.

正确
Correct! Only algorithms that update every state exactly once at each iteration are synchronous.

- [ ] Synchronous, if it systematically sweeps the entire state space at each iteration.

正确
Correct! Only algorithms that update every state exactly once at each iteration are synchronous.

Dynamic Programming

练习测验, 14 个问题

14/14 分 (100%)

1 / 1
分数

8.
All Generalized Policy Iteration algorithms are synchronous.

- True
- False

正确
Correct! A Generalized Policy Iteration algorithm can update states in a non-systematic fashion.

1 / 1
分数

9.
Policy iteration and value iteration, as described in chapter four, are synchronous.

- True

正确
Correct! As described in lecture, policy iteration and value iteration update all states systematic sweeps.

- False

1 / 1
分数

10.
Which of the following is true?

- Asynchronous methods generally scale to large state spaces better than synchronous methods.

正确
Correct! Asynchronous methods can focus updates on more relevant states, and update less relevant states less often. If the state space is very large, asynchronous methods may still be able to achieve good performance whereas even just one synchronous sweep of the state space may be intractable.

- Synchronous methods generally scale to large state spaces better than asynchronous methods.

Dynamic Programming

练习测验, 14 个问题

14/14 分 (100%)

1 / 1 分数

11.

Why are dynamic programming algorithms considered planning methods? [Select all that apply]

- [x] They use a model to improve the policy.

正确

Correct! This is the definition of a planning method.

- [x] They compute optimal value functions.

未选择的是正确的

- [x] They learn from trial and error interaction.

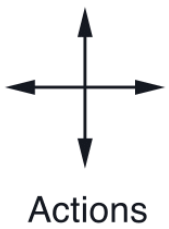
未选择的是正确的

---

1 / 1 分数

12.

Consider the undiscounted, episodic MDP below. There are four actions possible in each state, A = {up, down, right, left}, which deterministically cause the corresponding state transitions, except that actions that would take the agent off the grid in fact leave the state unchanged. The right half of the figure shows the value of each state under the equiprobable random policy. If π is the equiprobable random policy, what is $q(11, \text{down})$?



T	1	2	3
4	5	6	7
8	9	10	11
12	13	14	*T*
	15		

R = -1 on all transitions

T	-14.	-20.	-22.
-14.	-18.	-20.	-20.
-20.	-20.	-18.	-14.
-22.	-20.	-14.	T

- ( ) $q(11, \text{down}) = -1$

正确

Correct! Moving down incurs a reward of -1 before reaching the terminal state, after which the episode is over.

- ( ) $q(11, \text{down}) = -14$

Dynamic Programming

练习测验, 14 个问题

14/14 分 (100%)

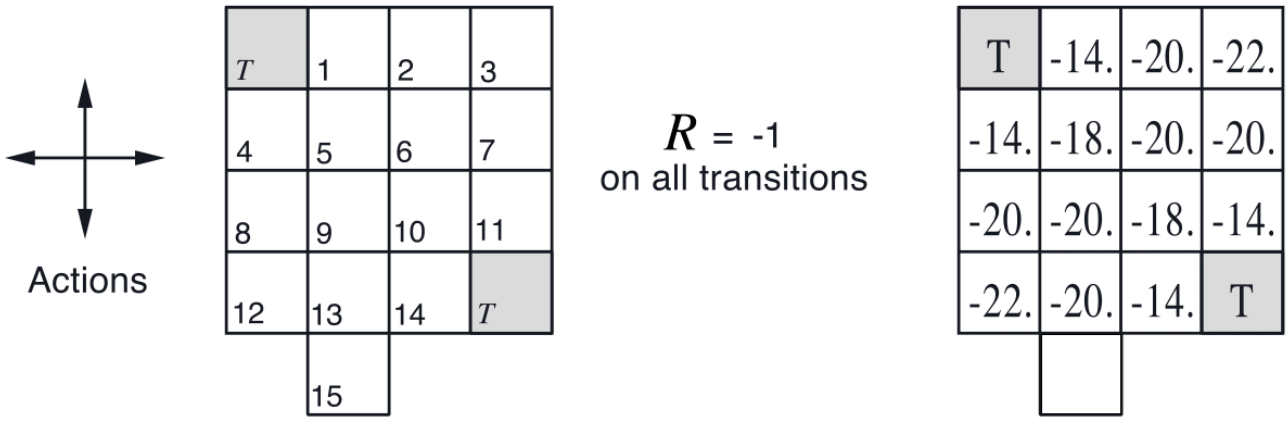
$q(11, \text{down}) = -15$

$q(11, \text{down}) = 0$

1 / 1
分数

13.

Consider the undiscounted, episodic MDP below. There are four actions possible in each state, A = {up, down, right, left}, which deterministically cause the corresponding state transitions, except that actions that would take the agent off the grid in fact leave the state unchanged. The right half of the figure shows the value of each state under the equiprobable random policy. If π is the equiprobable random policy, what is $q(7, \text{down})$?



$q(7, \text{down}) = -20$

$q(7, \text{down}) = -15$

正确

Correct! Moving down incurs a reward of -1 before reaching state 11, from which the expected future return is -14.

$q(7, \text{down}) = -21$

$q(7, \text{down}) = -14$

1 / 1
分数

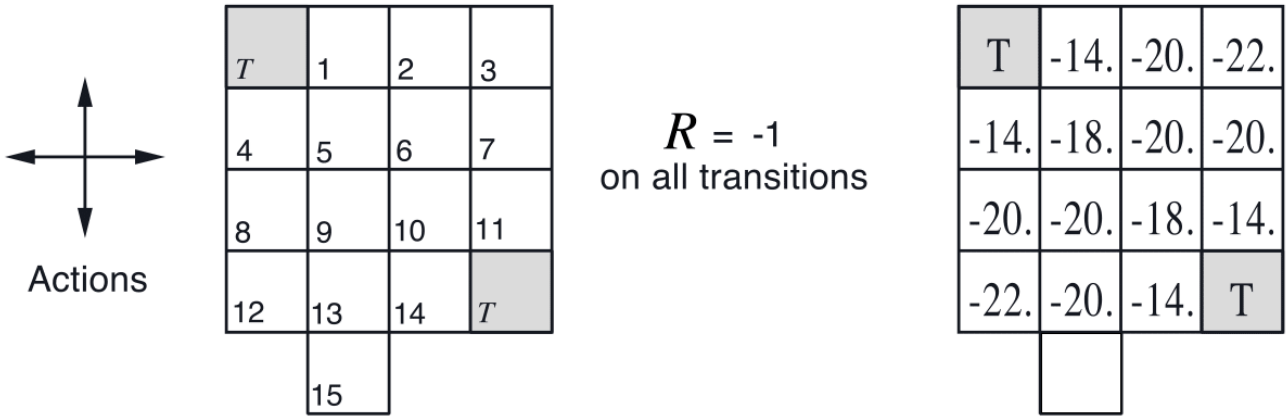
14.

Dynamic Programming

练习测验, 14个问题

14/14 分 (100%)

Consider the undiscounted, episodic MDP below. There are four actions possible in each state, A = {up, down, right, left}, which deterministically cause the corresponding state transitions, except that actions that would take the agent off the grid in fact leave the state unchanged. The right half of the figure shows the value of each state under the equiprobable random policy. If π is the equiprobable random policy, what is $v(15)$? Hint: Recall the Bellman equation $v(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma v(s')]$.



- $v(15) = -21$
- $v(15) = -22$
- $v(15) = -24$

正确

Correct! We can get this by solving for the unknown variable $v(15)$. Let's call this unknown x. We solve for x in the equation $x = 1/4(-21) + 3/4(-1 + x)$. The first term corresponds to transitioning to state 13. The second term corresponds to taking one of the other three actions, incurring a reward of -1 and staying in state x.

- $v(15) = -23$
- $v(15) = -25$