COMP3702/7702 Artificial Intelligence
Semester 2, 2020
Tutorial 8 - Sample Solutions

Exercise 8.1

One-year profit of buying 5 GenCar is:

$$\begin{aligned}&5(330(\$175 - \$25) - 35 \times \$30 - \$40000)\\ &= 5 \times \$8450\\ &= \$42250\end{aligned}$$

One-year profit of buying 2 Tesla Model X (no modification) is:

$$\begin{aligned}&2(0.75(330(\$500 - \$10) - 35 \times (\$30 + \$5))\\ &+0.25(330(-\$5) - 35 \times (\$30 + \$5))\\ &-\$120000)\\ &\quad = \$60712.5 - \$61437.5\\ &\quad = -\$725\end{aligned}$$

One-year profit of buying 2 Tesla Model X with modifications is the profit/loss of buying 2 Teslas plus the profit for the upgrade:

$$\begin{aligned}&-725{+}2(0.75(330 \times \$100 - \$20000)\\ &\qquad +0.25(330(-\$5) - 35 \times (\$30 + \$5))\\ &\qquad -\$120000)\\ &\qquad\quad = -\$725 + \$9500\\ &\qquad\quad = \$8775\end{aligned}$$

Therefore, UQCarRental should buy 5 GenCars.

Exercise 8.2

Derive V^π from the MDP objective function.

Start with:

$$V(s_0) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)\right]$$

Expand and collect:

$$V(s_0) = \mathbb{E}\left[R(s_0, a_0) + \gamma\left(R(s_1, a_1) + \gamma\left(R(s_2, a_2) \ldots\right)\right)\right]$$

Introduce a recursion of V:

$$V(s_0) = \mathbb{E}\left[R(s_0, a_0) + \gamma V(s_1)\right]$$

Evaluate the expectation using $\sum_{s_1} P(s_1 \mid s_0, a_0)$, and recalling $R(s,a) = \sum_{s'} P(s' \mid s,a) R(s,a,s')$:

$$V(s_0) = \sum_{s_1} P(s_1 \mid s_0, a_0)\left[R(s_0, a_0, s_1) + \gamma V(s_1)\right]$$

Introduce the policy π, such that $a = \pi(s)$:

$$V^{\pi}(s_0) = \sum_{s_1} P(s_1 \mid s_0, \pi(s_0))\left[R(s_0, \pi(s_0), s_1) + \gamma V^{\pi}(s_1)\right]$$

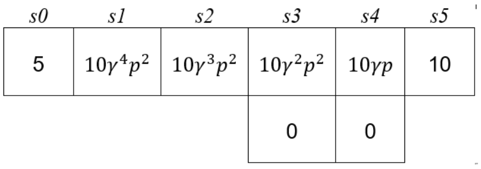
...and drop the time index (ie. set $s_0 = s$ and $s_1 = s'$):

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

Exercise 8.3

Note that for this question, once the agent arrives on a square with a value, it has only one action available to it, that is, to *exit* the environment. The rewards stated on the squares with values are the reward for *exiting* the square and the environment, so the reward is not repeatedly earned. For example, $R(s_5, \mathit{exit}) = 10$, and $R(s_{4b}, \mathit{exit}) = 0$ (where s_{4b} is the square below s_4).

a) The value of the policy π_R, which is to always move right when possible, is:



s0	*s1*	*s2*	*s3*	*s4*	*s5*
5	$10\gamma^4p^2$	$10\gamma^3p^2$	$10\gamma^2p^2$	$10\gamma p$	10
			0	0	

You could write this as:

$$\begin{aligned}
V^{\pi_R}(s_1) &= 10\gamma^4 p^2 \\
V^{\pi_R}(s_2) &= 10\gamma^3 p^2 \\
V^{\pi_R}(s_3) &= 10\gamma^2 p^2 \\
V^{\pi_R}(s_4) &= 10\gamma p
\end{aligned}$$

Exercise **b)** is very similar, so is left for you to complete on your own.

Exercise 8.4

We define an MDP via the following:

- S: state space
- A: action space
- T: transition function
- R: reward function
- γ: discount factor

State space:
The state space consists of the number of apples and bananas that the store has in stock, i.e. a state is a tuple (a, b) where a is the number of cans of apples and b is the number of cans of bananas, and a and b are non-negative integers that add to 4 or less.

Action space:
The action space consists of the possible ways we could purchase (restock) 3 or fewer cans. As such, similarly to the states, our action space consists of tuples (f, g) where s is the number of additional apples we purchase, and t is the number of additional bananas we purchase, which are non-negative integers such that $b + g \leq 3$.

In addition, we will also specify that we are not allowed to purchase cans in excess of our maximum stock, so that for a state (a, b) and action (f, g) it must be the case that $a+b+f+g \leq 4$, i.e. some actions are illegal in some states.

Transition function:
The transition function defines the probabilistic transitions to next states, given current states and actions. From the problem definition, we know that the transitions for apples and bananas are independent of one another, so we can express the transition function in terms of two separate functions T_a for apples, and T_b for bananas, with the overall transition probability being defined as:

$$T\big((a, b), (f, g), (a', b')\big) = T_a(a, f, a') \cdot T_b(b, g, b')$$

The probabilities for T_a and T_b are derived from the consumer's probabilities of consuming apples or bananas, which depend upon the amounts of apples $a + f$ and bananas $b + g$ that we have at the start of the day. For convenience, we will use the notation $P_a[i, j]$ to refer to entries from the matrix P_a. If $a' > a + f$, we know that $T_a(a, f, a') = 0$ since fruit does not appear out of nowhere. For $0 < b' \leq b + g$ we know that:

$$T_a(a, f, a') = P_a[a + f, a + f - a']$$

Finally, for $a' = 0$ we know that the users wanted to consume at least $a + f$ cans of apples, but might have wanted to consumer more, so that:

$$T_a(a, f, a') = \sum_{j=a+f}^{4} P_a[a + f, j]$$

We can present this in combined form as:

$$T_a(a,f,a') = \begin{cases} 0 & \text{if } a' > a+f \\ P_a[a+f, a+f-a'] & \text{if } 0 < a' \leq a+f \\ \sum_{j=a+f}^{4} P_a[a+f, j] & \text{if } a' = 0 \end{cases}$$

Similarly, for bananas we have:

$$T_b(b,g,b') = \begin{cases} 0 & \text{if } b' > b+g \\ P_b[b+g, b+g-b'] & \text{if } 0 < b' \leq b+g \\ \sum_{j=b+g}^{4} P_b[b+g, j] & \text{if } b' = 0 \end{cases}$$

Finally, the complete transition function (as stated before) is:

$$T\big((a,b),(f,g),(a',b')\big) = T_a(a,f,a') \cdot T_b(b,g,b')$$

This is sufficient to answer the question – we don't need the actual numbers until we're asked to do some kind of calculation with them. Note that this is an example of a *factored* transition function, in which the two state factors are not explicitly linked (they are the product of independent distributions, also know as *product distributions*).

For example, let's say we're asked to find the probability of ending up with 3 apples and zero bananas after starting the day with 3 apples and a banana (having restocked one banana that morning), i.e. we want to know $T((3,0),(0,1),(3,0)$. For this we calculate $T_a(3,0,3) = P_a[3,0] = 0.3$ and $T_b(0,1,0) = P_b[1,1] + P_b[1,2] + P_b[1,3] + P_b[1,4] = 0.2 + 0.2 + 0.1 + 0.3 = 0.8$. Then we multiply to get $T((3,0),(0,1),(3,0)) = 0.3 \times 0.8 = 0.24$.

Reward function:
According to the problem definition, we want to minimise the number of customers who don't get their choice of fruit. For this, we specify a reward of -1 every time a customer wants a fruit but it is not available.

In principle, this reward depends on a stochastic variable that is not fully specified by the state and action, i.e., the number of customers for apples and bananas on any particular day. However, since in an MDP we care about maximising the expected total utility, we can simply replace the stochastic reward with the expected reward, by summing over the different possible outcomes.

As with the transition function, we can split the reward function into components, i.e.

$$R\big((a,b),(f,g)\big) = R_a(a,f) + R_b(b,g)$$

where R_a is the penalty for unsatisfied customers who wanted apples, and R_b is the penalty for unsatisfied customers who wanted bananas.

Additionally, we can derive R_a as:

$$R_a(a,f) = -1 \sum_{k=a+f+1}^{4} (k-a-f) P_a[a+f,k]$$

and similarly R_b as:

$$R_b(b,g) = -1 \sum_{k=b+g+1}^{4} (k-b-g)P_b[b+g,k]$$

Note that for $a+f=4$ this definition results in $R_a(a,f)=0$, and the same for bananas, since it is not possible for there to be more than 4 customers for either fruit.

For an example, we can derive the reward for state (3,0) and action (0,1) via $R_a(3,0) = -1(1 \cdot P_a[3,4]) = -0.2$ and $R_b(0,1) = -1(P_b[1,2] + 2P_b[1,3] + 3P_b[1,4]) = -1(0.2+0.2+0.6) = -1$, which gives us $R\big((3,0),(0,1)\big) = -0.2-1 = -1.2$.

Discount factor:
We can set the discount factor $\gamma \in [0,1]$. Since it's mathematically convenient for making things converge, we can use a discount factor of 0.99. This is a reasonable choice since the sales happen on a day-to-day basis, possibly over many days. One possible justification for a discount factor in this problem is that reputation spreads over time, and thus our performance earlier on has more impact than our later performance.