Stochastic Optimisation - Problem Sheet 6

Dom Hutchinson

December 15, 2020

Answer 2) (a)
Here I formulate the problem described in 2) as a discounted reward Markov decision problem over an infinite-horizon.

- *Decision Epochs* - Start of each week.
- *Epoch t* - The beginning of time-period t.
- *Time Horizon* - $T = \{0, 1, \dots\}$.
- *System States.*

  Let X_t be the system state at epoch t defined as the wage the employ is offered in epoch t.
- *State-Space* - $S = \mathcal{W}$.
- *Agent-Actions.*

  Let Y_t be the action taken by the employee in epoch t, defined to be whether the employee accepts the job or not.

$$Y_t = \mathbb{1}\{\text{accepts job}\}$$

- *Action-Space* - $A = \{0, 1\}$.
- *Admissible Actions.*

$$\begin{array}{rcl} A(w(0)) & = & \{0\} \\ A(s) & = & \{0,1\} \quad \forall\ s \in \mathcal{W} \setminus \{w(0)\} \end{array}$$

- *Transition Probabilities.*

  Let $s, s' \in S, a \in A(s)$ and define $p_t(s'|s,a) = \mathbb{P}(X_{t+1} = s'|X_t = s, Y_t = 1)$. Consider the following two cases

Case 1 The agent accepted their last job offer (ie $a = 1$).

$$p_t(s'|s,1) \quad = \quad \begin{cases} p & \text{if } s' = s \\ 1-p & \text{if } s' = 0 \end{cases}$$

Case 2 The agent rejected their last job offer (ie $a = 0$).

$$p_t(s'|s,0) = q(s'|s)$$

- *Equivalent Rewards.*

  Let $s \in S, a \in A(s)$. In each epoch the agent earns the wage they are offered s if they accepted the job (ie $a = 1$), otherwise they earn nothing.

  $$r(s, a) = s \cdot a$$

- *Equivalent Objective.*

  Find the policy, $\pi \in HR(T)$ which maximises the expected discounted reward

  $$\mathbb{E}^{\pi}\left[\sum_{t=0}^{\infty} \alpha^t r(X_t, Y_t)\right]$$

  where $\alpha \in (0, 1)$ is the discounting factor.

Answer 2) (b)
From the specification of the question we know the following

$$\begin{array}{rcl}
\mathcal{W} & = & \{w(0), w(1), w(2)\} \\
 & = & \{0, 12/5, 16\} \\
S & = & \{0, 12/5, 16\} \\
A(0) & = & \{0\} \\
A(12/5) & = & \{0, 1\} \\
A(16) & = & \{0, 1\} \\
P & = & \begin{pmatrix} 0.2 & 0.5 & 0.3 \\ 0.3 & 0.3 & 0.4 \\ 0.3 & 0.5 & 0.2 \end{pmatrix}
\end{array}$$

Consider a policy π st $Y_t = 1$ if $X_t > 0$ and $Y_t = 0$ if $X_t = 0$.

Here I shall use the *Policy Iteration Algorithm* to determine whether this policy π is optimal. π is optimal if the algorithm does not suggest a different policy after one iteration.

Define the following

$$\begin{array}{rcl}
w_{k+1}(s, a) & = & r(s, a) + \alpha \sum_{s' \in S} v_k(s') p(s'|s, a) \\
d_{k+1}(s) & \in & \text{argmax}_{a \in A(s)} w_{k+1}(s, a)
\end{array}$$

Using the policy π, I initialise the algorithm as follows

$$d_0(w_0) = 0 \quad d_0(w_1) = 1 \quad d_0(w_2) = 1$$

Note that $d_0(s)$ can be any Markovian decision function which satisfies $d_0(s) \in A(s)\ \forall\ s \in S$.

Consider an iteration of the algorithm where $k = 0$. We want to compute a solution $v_0(s)$ for the following

$$\begin{array}{rcl}
v(s) & = & (T_{d_0} v)(s) \\
 & = & r(s, d_0(s)) + \alpha \sum_{s' \in S} v(s') p(s'|s, d_0(s))
\end{array}$$

This can be expanded to the following system of three equations

$$\begin{aligned}
v(w_0) &= \frac{3}{4}\left(\frac{v(w_0)}{5}+\frac{v(w_1)}{2}+\frac{3v(w_2)}{10}\right)\\
&= \frac{3}{20}v(w_0)+\frac{3}{8}v(w_0)+\frac{9}{40}v(w_2)\\
v(w_1) &= \frac{12}{5}+\frac{3}{4}\left(\frac{3v(w_0)}{10}+\frac{3v(w_1)}{10}+\frac{2v(w_2)}{5}\right)\\
&= \frac{12}{5}+\frac{9}{40}v(w_0)+\frac{9}{40}v(w_1)+\frac{3}{10}v(w_2)\\
v(w_2) &= 16+\frac{3}{4}\left(\frac{3v(w_0)}{10}+\frac{v(w_1)}{2}+\frac{v(w_2)}{5}\right)\\
&= 16+\frac{9}{40}v(w_0)+\frac{3}{8}v(w_1)+\frac{3}{20}v(w_2)
\end{aligned}$$

I CANT BE ASKED TO SOLVE THIS

Answer 3) (a)

Here I formulate the problem described in 3) as a discounted reward Markov decision problem over a finite-horizon.

- *Decision Epochs* - Start of each month.
- *Epoch t* - Start of time-period t.
- *Time Horizon* - $T = \{0, 1, \dots, N\}$
- *System States.*

  Let X_t be the system state at epoch t, defined as the number of sales in the previous month.
- *State-Space* - $S = \{b(1), \dots, b(n)\}$
- *Agent-Actions.*

  Let Y_t be the action taken by the company in epoch t, defined as the strategy the company chooses to use.
- *Action-Space* - $A = \{1, \dots, m\}$
- *Admissible Actions* - $A(s) = S \ \forall\ s \in S$.
- *Transition Probabilities* - $\mathbb{P}(X_{t+1} = s'|X_t = s, Y_t = a) = p(s'|s, a)$ as defined in question.
- *Equivalent Rewards.*

  In each epoch the companies earns from sale X_t but has to pay for its marketing campaign $c(Y_t)$.
  $$r(s, a) = s - c(a)$$
- *Equivalent Objective.*

  Find the policy $\pi \in HR(T)$ which maximises the expected discounted reward
  $$\mathbb{E}^\pi\left[\sum_{t=0}^{N}\alpha^t r(X_t, Y_t)\right]$$
  where $\alpha \in (0, 1)$ is the discounting factor.

Answer 3)(b)
From the specification of the question we know the following

$$\begin{array}{rcl} T & = & \{0,\dots,N\} \\ S & = & \{b_1,b_2\}=\{2,8\} \\ A & = & \{1,2\} \\ A(s) & = & \{1,2\}\ \forall\ s\in S \\ p(s'|s,1) & = & \begin{array}{c|cc} \text{s'}\backslash s & \text{b}_1 & \text{b}_2 \\ \hline \text{b}_1 & .8 & .7 \\ \text{b}_2 & .2 & .3 \end{array} \\ \\ p(s'|s,2) & = & \begin{array}{c|cc} \text{s'}\backslash s & \text{b}_1 & \text{b}_2 \\ \hline \text{b}_1 & .4 & .2 \\ \text{b}_2 & .6 & .8 \end{array} \end{array}$$

I shall now implement the *Policy Iteration Algorithm* in order to find an optimal policy from this specification of the problem in `3) (a)`.

Initialisation - Let $k=0$ and define $d_0(b_1=2,d_0(b_2)=2$.

Iteration 1 - $k=1$

Policy Evaluation - I shall compute a solution $v_0(s)$ for the following equation

$$\begin{array}{rcl} v(s) & = & (T_{d_0}v)(s) \\ & = & r(s,d_0(s))+\alpha\displaystyle\sum_{s'\in S}v(s')p(s'|s,d_0(s)) \end{array}$$

This can be expanded to the following series of equations

$$\begin{array}{rcl} v(b_1) & = & r(b_1,2)+\frac{1}{2}\big[v(b_1)p(b_1|b_1,d_0(b_1))+v(b_2)p(b_2|b_1,d_0(b_1))\big] \\ & = & r(b_1,2)+\frac{1}{2}\big[v(b_1)p(b_1|b_1,2)+v(b_2)p(b_2|b_1,2)\big] \\ & = & (b_1-c(2))+\frac{1}{2}\left[v(b_1)\cdot\frac{2}{5}+v(b_2)\frac{3}{5}\right] \\ & = & -3+v(b_1)\cdot\frac{1}{5}+v(b_2)\frac{3}{10} \\ v(b_2) & = & r(b_2,2)+\frac{1}{2}\big[v(b_1)p(b_1|b_2,d_0(b_2))+v(b_2)p(b_2|b_2,d_0(b_2))\big] \\ & = & r(b_2,2)+\frac{1}{2}\big[v(b_1)p(b_1|b_2,2)+v(b_2)p(b_2|b_2,2)\big] \\ & = & (b_2-c(2))+\frac{1}{2}\left[v(b_1)\cdot\frac{1}{5}+v(b_2)\cdot\frac{4}{5}\right] \\ & = & 3+v(b_1)\frac{1}{10}+v(b_2)\frac{2}{5} \end{array}$$

A solution to this is

$$v_0(b_1)=-2\quad v_0(b_2)=\frac{14}{3}$$

Policy Improvement - I shall compute $d_1(s)$ using the following system of equations

$$\begin{array}{rcl} w_1(s,a) & = & r(s,a)+\alpha\displaystyle\sum_{s'\in S}v_0(s')p(s'|s,a) \\ d_1(s) & \in & \text{argmax}_{a\in A(s)}w_1(s,a) \end{array}$$

The tables below summarise the values for these equations

$$\text{w}_1(s,a) \quad = \quad \begin{array}{c|cc} s\backslash a & 1 & 2 \\ \hline b_1 & 2/3 & \text{-2} \\ b_2 & 7 & 14/3 \end{array}$$

$$\text{d}_1(s) \quad = \quad \begin{array}{c|c} s & d_1(s) \\ \hline b_1 & 1 \\ b_2 & 1 \end{array}$$

$d_0(s) \neq d_1(s)$ so I perform another iteration of the algorithm.

Iteration 2 - $k = 2$

Policy Evaluation - I shall compute a solution $v_1(s)$ for the following system of equations

$$\begin{aligned}
v(b_1) &= r(b_1,1) + \frac{1}{2}\left[v(b_1)p(b_1|b_1,d_1(b_1)) + v(b_2)p(b_2|b_1,d_1(b_1))\right] \\
&= r(b_1,1) + \frac{1}{2}\left[v(b_1)p(b_1|b_1,1) + v(b_2)p(b_2|b_1,1)\right] \\
&= (2-1) + \frac{1}{2}\left[v(b_1)\frac{4}{5} + v(b_2)\frac{1}{5}\right] \\
&= 1 + v(b_1)\frac{2}{5} + v(b_2)\frac{1}{10} \\
v(b_2) &= r(b_2,1) + \frac{1}{2}\left[v(b_1)p(b_1|b_2,d_1(b_2)) + v(b_2)p(b_2|b_2,d_1(b_2))\right] \\
&= r(b_2,1) + \frac{1}{2}\left[v(b_1)p(b_1|b_2,1) + v(b_2)p(b_2|b_2,1)\right] \\
&= (8-1) + \frac{1}{2}\left[v(b_1)\frac{7}{10} + v(b_2)\frac{3}{10}\right] \\
&= 7 + v(b_1)\frac{7}{20} + v(b_2)\frac{3}{20}
\end{aligned}$$

A solution to this is

$$v_1(b_1) = \frac{69}{44} \quad v_1(b_2) = \frac{245}{22}$$

Policy Improvement - I shall compute $d_2(s)$ using the following system of equations

$$\begin{aligned}
w_2(s,a) &= r(s,a) + \alpha \sum_{s' \in S} v_1(s')p(s'|s,a) \\
d_2(s) &\in \text{argmax}_{a \in A(s)} w_2(s,a)
\end{aligned}$$

The tables below summarise the values for these equations

$$\text{w}_2(s,a) \quad = \quad \begin{array}{c|cc} s\backslash a & 1 & 2 \\ \hline b_1 & 600/220 & 36/55 \\ b_2 & 8113/880 & 6698/880 \end{array}$$

$$\text{d}_2(s) \quad = \quad \begin{array}{c|c} s & d_1(s) \\ \hline b_1 & 1 \\ b_2 & 1 \end{array}$$

$d_1(s) = d_2(s)\ \forall\ s \in S$. Thus the algorithm terminates and our optimal policy is

$$d(b_1) = 1 \quad d(b_2) = 1$$