CIS 510 Assignment 3

Steven Walton

June 7, 2019

Problem P1

Note: This code uses Cplex, which does not work on python 3.7. This code was written with python 3.6.5 and will be assumed that the user is using a similarly compatible python version. 3.5 and 3.6 should be compatible with this code though it is not tested.
Below is the help file for the COBRA-C program.

```
python COBRA-C.py -h
usage: COBRA-C.py -p <parameter file> -i <payoff file> -o <output file>
-p, --params      sets the parameter file
-i, --payoff      sets the payoff file
-o, --output      sets the output file. Defaults to out.csv
-d, --delimiter   sets the delimiter of ALL files. Defaults to csv
```

We can simply run the program by using the following command

```
python COBRA-C.py -p param.csv -i payoff.csv
```

Problem Q1

Part 1)

Suppose a seller runs a second-price, sealed-bid auction for a painting. There are two bidders with independent, private values. The seller does not know their precise valuations, but knows: (a) each bidder i has one of three values, $v_i = 2$, $v_i = 4$ or $v_i = 8$; and (b) each of these values is equally likely (i.e., occurs with probability $\frac{1}{3}$). When running the auction, if the two bids are tied (say, at x), the winner is chosen at random (and pays x).
1. Assume both bidders use their dominant strategies for bidding in a second-price auction. What is the sellers expected revenue in this auction? Please explain your answer.
We need to examine all possible payoffs by taking the average of each outcome where the expectation value is

$$E[X] = \sum_{i=1}^{k} x_i p_i$$

p_i represents the given probability for a given value x_i We therefore have the expected revenue of

$$\begin{aligned} E[X] &= \frac{5}{9} * 2 + \frac{3}{9} * 4 + \frac{1}{9} * 8 \\ &= \frac{10}{3} \end{aligned}$$

2. Now the seller decides to set a reserve price of r. This means that if the highest bid is below r, the seller will not sell the item. If the highest bid is at least r, then the painting will go to the highest bidder, and the winner will pay the maximum of r and the second-highest bid. Suppose the reserve price is set to $r = 4$. Assume both bidders use their dominant strategies. What is the sellers expected revenue in this auction? Please explain your answer. If the expected revenue increases or decreases relative to your answer in part (1), give a qualitative explanation for why this change occurs.
We do the same thing as part 1 but set a minimum bid of 4.

$$\begin{aligned} E[X] &= \frac{1}{9} * 0 + \frac{7}{9} * 4 + \frac{1}{9} * 8 \\ &= 4 \end{aligned}$$

3. Is there a better reserve price than $r = 4$ (i.e., that will provide more revenue for the seller)? Give a brief justification for your response.
By using a reserve price of 8 we can get a better revenue for the seller

$$\begin{aligned} E[X] &= \frac{4}{9} * 0 + \frac{5}{9} * 8 \\ &= \frac{40}{9} \end{aligned}$$

Part 2)

Consider an IPV setting where three bidders are risk-neutral and each bidders valuation is drawn from a uniform distribution over the range $[a, b]$. Describe the optimal auction using virtual valuation and bidder-specific reserve price for this setting. What is the sellers expected revenue using this optimal auction mechanism, given bidders play the dominant strategy.
For this we know that there is a probability density function (PDF) of $\frac{1}{a-b}$ and a cumulative distribution function (CDF) of $\frac{v_i - a}{b-a}$ which gives us the virtual validation function

$$\begin{aligned} \psi_i(v_i) &= v_i - \frac{1 - F_i(v_i)}{f_i(v_i)} \\ &= v_i - \frac{\frac{v_i - a}{b-a}}{\frac{1}{b-a}} \\ &= v_i - (b - v_i - 2a) \\ &= 2a - b \end{aligned}$$

We then can calculate the reserve price by setting $\psi = 0$ which gives us $r = \frac{b}{2}$. We note that the reserve price is the same for all players.

We then need to consider the following cases.
1) $v_1, v_2 < r$: In this case neither player wins so both of their expected revenues is 0. This happens with probability $(r-a)^2$ (we just take the area from between a and r)
2) $v_i > r, v_j < r$: In this case the second highest bid is the reserve price, r. The probability that this happens is $(b-r)(r-a)$
3) $v_i, v_j > r$: In this case we have the probability that both are above as $(b-r)^2$ and need to find the minimum

$$\begin{aligned}&\min\{E[v_1], E[v_j]\}\\ &= \frac{1+2r}{3}\end{aligned}$$

Now we need to sum these to get the total expected revenue

$$\begin{aligned}=&(r-a)^2 * 0 + (b-r)(r-a) * r + (b-r)^2 * \frac{1+2r}{3}\\ =&\frac{1}{3}\left(-br^2 - r^3 - 3abr + 3ar^2 + b^2 - 2br + 2rb^2 + r^2\right)\end{aligned}$$

Part 3)

Consider a first price sealed-bid auction with n risk-neutral agents whose valuations, $v_1, \cdots, v_n$, are independently drawn from a uniform distribution on the interval $[0, b]$. Prove that $\left(\frac{n-1}{n}v_1, \cdots, \frac{n-1}{n}v_n\right)$ is a Bayes-Nash equilibrium.
We will follow a similar formula to the two player game that we did in class. We will let $\mathcal{V}$ be the space of players.

$$\begin{aligned}&\int_{\mathcal{V},0}^{b} u_1(s_1)dv\\ =&\int_{\mathcal{V},j}^{s_1} u_1(s_1)dv + \int_{\mathcal{V},s_1}^{b} u_1(s_1)dv\\ =&\int_{\mathcal{V},0}^{s_1} u_1(s_1)dv + 0\\ =&\int_{\mathcal{V},0}^{s_1} (v_1 - s_1)dv\\ =&(v_1 - s_1)\int_{\mathcal{V},0}^{s_1} dv\\ =&(v_1 - s_1)(s_1^{n-1} - a)\\ =&s_1^{n-1}v_1 - s_1^n - av_1 + as_1\end{aligned}$$

$$\begin{aligned}&\left(\frac{d}{ds_1}\{s_1^{n-1} - s_1^n\}\right)\\ &= (n-1)v_1 - ns_1\\ ns_1 &= (n-1)v_1 + a\\ s_1 &= \frac{n-1}{n}v_1\end{aligned}$$

This method can similarly be used for each player following the same pattern. We should see that v_1, s_1 can be replaced with v_i, s_i and we will get a similar result.
Original problem had bounds $[a, b]$ which creates an offset by a. This results in the profit smaller than a being obtained.

Problem Q2

Image an unknown game which has three states $\{A, B, C\}$ and in each state the agent has two actions to choose from $\{Up, Down\}$. Suppose a game agent chooses actions according to some policy π and generates the following sequence of actions and rewards in the unknown game:

t	s_t	a_t	s_{t+1}	r_t
0	A	Down	B	2
1	B	Down	C	3
2	C	Up	B	-2
3	B	Down	B	0
4	B	UP	A	1
5	A	Down	C	-3
6	C	Down	A	2
7	A	Up	C	1
8	C	Down	B	2
9	B	Down	A	2
10	A	Up	B	3

Table 1: $\gamma = 0.5$ and $\alpha = 0.5$

Part a)

Assume that all Q-values are initialized to 0. What are the Q-values learned by running Q-learning with the above experience sequence?
We have the algorithm for updating values

$$Q(s,a) = T(s,a,s')[R(s,a,s') + \gamma V(s')$$

and

$$V^{\pi}(s) = (1-\alpha)V(s) + \alpha[R(s,\pi(s),s') + \gamma V^{\pi}(s')$$

Using these we will iterate over the values

$$Q_{init}(s,a) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$V = (1-0.5)0 + 0.5(2 + 0.5 * 0) = 1$$

$$Q_0(s,a) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$V = (1-0.5)0 + 0.5(3 + 0.5 * 0) = 1.5$$

$$Q_1(s,a) = \begin{bmatrix} 0 & 1 \\ 0 & \frac{3}{2} \\ 0 & 0 \end{bmatrix}$$

$$V = (1-0.5)0 + 0.5(-2 + 0.5 * \frac{3}{2}) = -\frac{5}{8}$$

$$Q_2(s,a) = \begin{bmatrix} 0 & 1 \\ 0 & \frac{3}{2} \\ -\frac{5}{8} & 0 \end{bmatrix}$$

$$V = (1-0.5)\frac{3}{2} + 0.5(0 + 0.5 * \frac{3}{2}) = \frac{9}{8}$$

$$Q_3(s,a) = \begin{bmatrix} 0 & 1 \\ 0 & \frac{9}{8} \\ -\frac{5}{8} & 0 \end{bmatrix}$$

$$V = (1-0.5)0 + 0.5(1 + 0.5 * 1) = \frac{3}{4}$$

$$Q_4(s,a) = \begin{bmatrix} 0 & 1 \\ \frac{3}{4} & \frac{9}{8} \\ -\frac{5}{8} & 0 \end{bmatrix}$$

$$V = (1-0.5)1 + 0.5(-3 + 0.5 * 0) = -1$$

$$Q_5(s,a) = \begin{bmatrix} 0 & -1 \\ \frac{3}{4} & \frac{9}{8} \\ -\frac{5}{8} & 0 \end{bmatrix}$$

$$V = (1-0.5)0 + 0.5(2 + 0.5 * 0) = 1$$

$$Q_6(s,a) = \begin{bmatrix} 0 & -1 \\ \frac{3}{4} & \frac{9}{8} \\ -\frac{5}{8} & 1 \end{bmatrix}$$

$$V = (1-0.5)0 + 0.5(1 + 0.5 * 1) = \frac{3}{4}$$

$$Q_7(s,a) = \begin{bmatrix} \frac{3}{4} & -1 \\ \frac{3}{4} & \frac{9}{8} \\ -\frac{5}{8} & 1 \end{bmatrix}$$

$$V = (1-0.5)1 + 0.5(2 + 0.5 * \frac{9}{8}) = \frac{57}{32}$$

$$Q_8(s,a) = \begin{bmatrix} \frac{3}{4} & -1 \\ \frac{3}{4} & \frac{9}{8} \\ -\frac{5}{8} & \frac{57}{32} \end{bmatrix}$$

$$V = (1-0.5)\frac{9}{8} + 0.5(2 + 0.5 * \frac{3}{4}) = \frac{28}{16}$$

$$Q_9(s,a) = \begin{bmatrix} \frac{3}{4} & -1 \\ \frac{3}{4} & \frac{28}{16} \\ -\frac{5}{8} & \frac{57}{32} \end{bmatrix}$$

$$V = (1-0.5)\frac{3}{4} + 0.5(3 + 0.5 * \frac{28}{16}) = \frac{37}{16}$$

$$Q_9(s,a) = \begin{bmatrix} \frac{37}{16} & -1 \\ \frac{3}{4} & \frac{28}{16} \\ -\frac{5}{8} & \frac{57}{32} \end{bmatrix}$$

Part b)

In a model-based reinforcement learning, we first estimate the transition function $T(s, a, s')$ and the reward function $R(s, a, s')$. Write down the estimates of T and R, estimated from the experience above. Write "n/a" if not applicable or undefined.
To figure this out we're going to reorder the above table for more clarity Here we can start calculating

s_t	a_t	s_{t+1}	r_t
A	Down	B	2
A	Down	C	-3
A	Up	B	3
A	Up	C	1
B	Down	A	2
B	Down	B	0
B	Down	C	3
B	UP	A	1
C	Down	A	2
C	Down	B	2
C	Up	B	-2

Table 2: Sorted by states and actions

the transition states by taking a given (s, a) pair and determining the probability of going to another state, s_{t+1}. We can determine the reward by normalizing.

$$\begin{aligned}
T(A, Down, B) &= \frac{1}{2} \\
T(A, Down, C) &= \frac{1}{2} \\
R(A, Down, B) &= 1 \\
R(A, Down, C) &= -\frac{3}{2} \\
T(A, Up, B) &= \frac{1}{2} \\
T(A, Up, C) &= \frac{1}{2} \\
R(A, Up, B) &= \frac{3}{4} \\
R(A, Up, C) &= \frac{1}{4} \\
T(B, Down, A) &= \frac{1}{3}
\end{aligned}$$

$$\begin{aligned}
T(B, Down, B) &= \frac{1}{3} \\
T(B, Down, C) &= \frac{1}{3} \\
R(B, Down, A) &= \frac{2}{5} \\
R(B, Down, B) &= 0 \\
R(B, Down, C) &= \frac{3}{5} \\
T(C, Down, A) &= \frac{1}{2} \\
T(C, Down, B) &= \frac{1}{2} \\
R(C, Down, A) &= \frac{1}{2} \\
R(C, Down, B) &= \frac{1}{2} \\
T(C, Up, B) &= 1 \\
R(C, Up, B) &= -2
\end{aligned}$$

Part c)

Assume we had a different experience and ended up with the following estimates of the transition and reward functions

s	a	s'	$\hat{T}(s,a,s')$	$\hat{R}(s,a,s')$
A	Up	A	1	12
A	Down	B	0.5	2
A	Down	C	0.5	-3
B	Up	B	1	8
B	Down	C	1	-6
C	Down	C	1	12
C	Up	C	0.5	2
C	Up	B	0.5	-2

(i) Give the optimal policy $\hat{\pi}^*(s)$ and $\hat{V}^*(s)$ for the MDP with transition function $\hat{T}$ and reward function $\hat{R}$. Explain your answers.
Our two easiest policies are for being in states A and C where we already have the maximal reward in the MDP.
So given state A, $\hat{\pi}^*(A) =$ Up we always pick A and similarly in state C we have the policy $\hat{\pi}^*(C) =$ Down to stay in C. Where in A we will always pick Up and in state C we will always pick Down. Because they have the same reward we know that finding one will result in the other.

We have the infinite equation

$$\begin{aligned}V^* =&\hat{R}(s,a,s')(1+\gamma+\gamma^2+\cdots)\\ =&\hat{R}(s,a,s')\left(\frac{1}{1-\frac{1}{2}}\right)\\ =&\hat{R}(s,a,s')2\\ =&24\end{aligned}$$

This gives us the value for both A and C, where $\hat{R}(A,Up,s') = \hat{R}(C,Down,s')$.
B is a little more difficult to find, but we can see that once we get to A or C we will use the above values.
We can simply look at $\pi(B)$ = Up and see that we will always get a reward of 8, giving us $V(B,Up) = 16$, similarly to above. We need to also look at $\pi(B)$ = Down. We see that we get $-6+\gamma V^*(C) = 6$. Here we know that $16 > 8 \therefore \hat{\pi}^*(B)$ = Up with $V^*(B) = 16$.

(ii) If we repeatedly feed this new experience sequence through our Q-learning algorithm, what values will it converges to? Assume that convergence is guaranteed.
We can use the equation

$$V^{\pi}_{k+1}(s) \leftarrow \sum_{s'} T(s,\pi(s),s')\left[R(s,\pi(s),s') + \gamma V^{\pi}_k(s')\right]$$

If we start with our optimal policies above we have

Q=

	Down	Up
A	0	24
B	0	16
C	16	0

Following the above equation and the values from the table and the above Q matrix we get

Q=

	Down	Up
A	14.5	24
B	6	16
C	16	10