Reinforcement Learning - TP3
Bandit Algorithm for Game Theory and RL

Chia-Man Hung

December 19, 2016

1 ADVERSARIAL MABS AND NASH EQUILIBRIA

The EXP3 algorithm:

Initialize the weight $w_{i,0} = 1$.

At every step t

- Compute ($W_{t-1} = \sum_{i=1}^{N} w_{i,t-1}$)
  $\hat{p}_{i,t} = (1-\beta)\frac{w_{i,t-1}}{W_{t-1}} + \frac{\beta}{K}$, where β is a parameter and K is the number of arms.
- Choose an arm at random $l_t \sim \hat{p}_t$
- Receive a reward $X_{l_t,t}$
- Update $w_{i,t} = w_{i,t-1} exp(\eta \hat{X}_{i_t,t})$
  where η is a parameter and the importance weight $\hat{X}_{i,t} = \frac{X_{i,t}}{\hat{p}_{i,t}}$ if $i = l_t$, 0 otherwise

Q1.
Illustrate the convergence towards the Nash equilibrium and the value of the game, for specified values of the parameters for EXP3.

A1.

Denote p_a as the mixed strategy used by A (the probability of A playing the action 1) and p_b as the mixed strategy used by B. The expectation of A's reward

$$E_{(p_a,p_b)}[R_A] = 2p_a p_b - p_a(1-p_b) + (1-p_a)(1-p_b) \tag{1.1}$$

The expectation of B's reward $E_{(p_a,p_b)}[R_B] = -E_{(p_a,p_b)}[R_A]$.

Figure 1.1: Convergence of the average number of choosing action 1 for A and B. $\eta = 0.01$, $\beta = 0.01$.

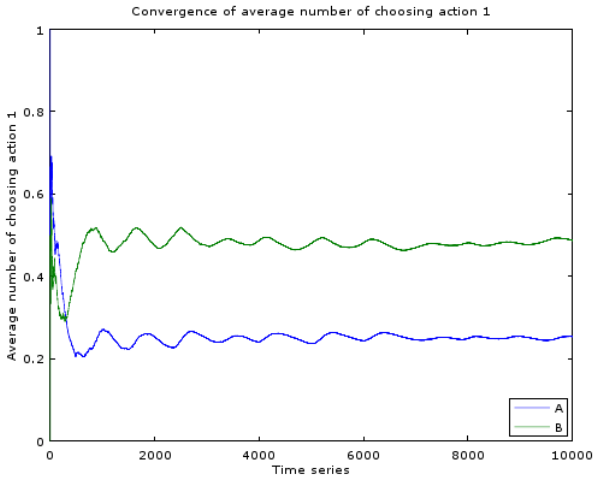
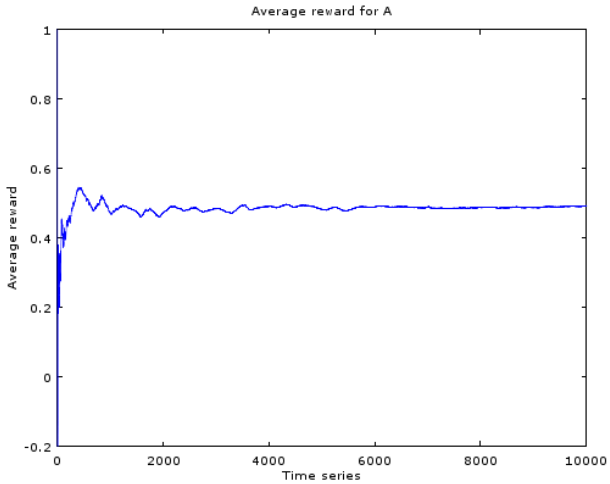


Figure 1.2: Convergence of the average reward for A. $\eta = 0.01$, $\beta = 0.01$.



At a Nash equilibrium, The derivative w.r.t. p_a and p_b is zero. This yields to

$$\begin{aligned} 2p_b^* - 2(1 - p_b^*) &= 0 \\ 2p_a^* + p_a^* - (1 - p_a^*) &= 0 \end{aligned} \tag{1.2}$$

$$\begin{aligned} p_a^* &= \frac{1}{4} \\ p_b^* &= \frac{1}{2} \end{aligned} \tag{1.3}$$

At the Nash equilibrium,

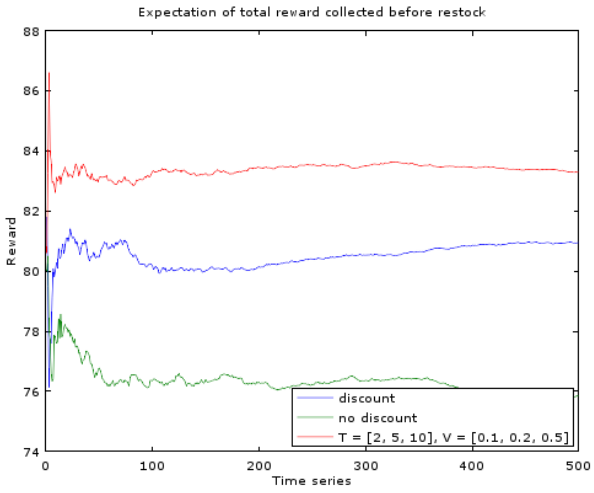
$$E_{(p_a^*, p_b^*)}[R_A] = \frac{1}{2} \tag{1.4}$$

This corresponds to what we found with the simulation.

2 Using MABs to solve an RL problem

For a specific choice of $T = (t_1, t_2, t_3)$ and $V = (v_1, v_2, v_3)$, compare the performance of π_{t_1,t_2,t_3} to the two baselines *soda_strategy_discount.m* ad *soda_strategy_nodiscount.m* that are given.

Figure 2.1: Convergence of the average reward for A. $\eta = 0.01$, $\beta = 0.01$.



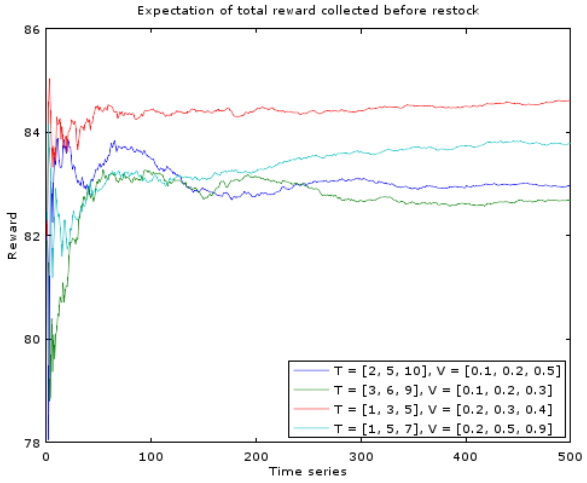
Q2.
Implement two out of these possible algorithms: UCB, KL-UCB, Thompson Sampling, EXP3. Notice that the normal implementation of bandit algorithms assumes that the performance of the bandit algorithm compared to the previous baselines and the best policy in your class.

A2.
Our MAB setting:

arm1: T = [2, 5, 10], V = [0.1, 0.2, 0.5]
arm2: T = [3, 6, 9], V = [0.1, 0.2, 0.3]
arm3: T = [1, 3, 5], V = [0.2, 0.3, 0.4]
arm4: T = [1, 5, 7], V = [0.2, 0.5, 0.9]

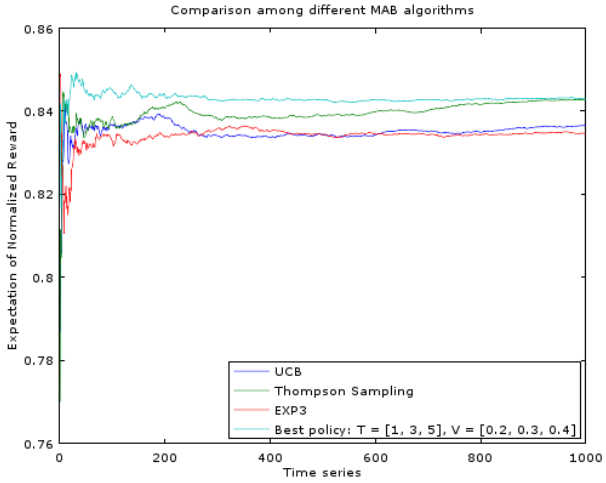
First, we determine the best arm in our MAB class. According to our simulation, arm3: T = [1, 3, 5], V = [0.2, 0.3, 0.4] seems to be the best one.

Figure 2.2: Finding the best policy.



Then, we compare different MAB algorithms: UCB, Thompson Sampling and EXP3.

Figure 2.3: Comparison among different MAB algorithms.



In conclusion, there is a small gap between the UCB, EXP3 and the best policy. UCB and EXP3 are comparable. Thompson Sampling performs better than UCB and EXP3 and converges towards the best policy.