1. Exploration vs. exploitation

According to $Q_t = \dfrac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i=a}}{\sum_{i=1}^{t-1} \mathbb{1}_{A_i=a}}$, get the Table as below

		$Q(1)$	$Q(2)$	$Q(3)$	$Q(4)$	$\arg\max Q_t(a)$
Initial state	$Q_1(a)$	0	0	0	0	1, 2, 3, 4
$A_1 = 1$	$Q_2(a)$	-1	0	0	0	2, 3, 4
$A_2 = 2$	$Q_3(a)$	-1	1	0	0	2
$A_3 = 2$	$Q_4(a)$	-1	$-\frac{1}{2}$	0	0	3, 4
$A_4 = 2$	$Q_5(a)$	-1	$\frac{1}{3}$	0	0	2
$A_5 = 3$	$Q_6(a)$	-1	$\frac{1}{3}$	0	0	

ε case will definitely occur on the time steps that $A_t \neq \arg\max_a Q_t(a)$

Hence, set T for time steps:

This possibly had occurred. when $T = 1, 2, 3$

This definitely had occurred when $T = 4, 5$

2. Varing step-size weights

Replace α by α_n in Eq. [2.5] :

$$Q_{n+1} = Q_n + \alpha_n [R_n - Q_n] = \alpha_n R_n + (1-\alpha_n) Q_n$$

$$= \alpha_n R_n + (1-\alpha_n) \cdot [\alpha_{n-1} R_{n-1} + (1-\alpha_{n-1}) Q_{n-1}]$$

$$= \alpha_n R_n + (1-\alpha_n)\alpha_{n-1} R_{n-1} + (1-\alpha_n)(1-\alpha_{n-1}) Q_{n-1}$$

$$= \alpha_n R_n + (1-\alpha_n)\alpha_{n-1} R_{n-1} + (1-\alpha_n)(1-\alpha_{n-1})\alpha_{n-2} \cdot R_{n-2}$$
$$+ (1-\alpha_n)(1-\alpha_{n-1})(1-\alpha_{n-2})\alpha_{n-3} R_{n-3} + \cdots + \prod_{i=1} (1-\alpha_i) Q_1$$

Hence, we can turn the equation to:

$$Q_{n+1} = \alpha_n R_n + \sum_{j=1}^{n-1} \alpha_j R_j \prod_{i=j+1}^{n-1} (1-\alpha_i) + \prod_{i=1}^{n} (1-\alpha_i) Q_1$$

The weighting on reward R_j is: $\prod_{i=j+1}^{n-1} (1-\alpha_i) \cdot \alpha_j$

3. Bias in Q-value estimate

(a) $Q_n = \dfrac{R_1 + R_2 + \cdots + R_{n-1}}{n-1}$. $E(Q_n) = \dfrac{1}{n-1} E[R_1 + R_2 + \cdots + R_{n-1}]$

$$= \frac{1}{n-1} \cdot (E[R_1] + E[R_2] + \cdots + E[R_{n-1}])$$

$$= \frac{1}{n-1} \cdot (n-1) q_*$$

$$= q_*$$

Hence, Eq. 2.1 is unbiased.

(b) Eq. 2.5 $\longrightarrow$ $Q_{n+1} = Q_n + \alpha [R_n - Q_n]$

If $Q_1 = 0$, then $Q_{n+1} = \sum_{i=1}^{n} \alpha (1-\alpha)^{n-i} R_i$

$$E(Q_{n+1}) = \sum_{i=1}^{n} \alpha (1-\alpha)^{n-i} \cdot E(R_i) = \sum_{i=1}^{n} \alpha (1-\alpha)^{n-i} \cdot q_*$$

$E(Q_{n+1}) = q_*$. It is unbiased, when $\sum_{i=1}^{n} \alpha (1-\alpha)^{n-i} = 1$

Otherwise, it is biased.

(c) As the result of (b), the condition for

Q_n being unbiased is: $\begin{cases} Q_1 = 0, \\ \sum_{i=1}^{n} \alpha (1-\alpha)^{n-i} = 1 \end{cases}$

(d) $Q_{n+1} = (1-\alpha)^n Q_1 + \sum_{i=1}^{n} \alpha(1-\alpha)^{n-i} R_i$

$$\lim_{n\to\infty} Q_{n+1} = \lim_{n\to\infty} (1-\alpha)^n Q_1 + \lim_{n\to\infty} \sum_{i=1}^{n} \alpha(1-\alpha)^{n-i} q_*$$

$\because$ $0<\alpha<1$, $0<(1-\alpha)<1$

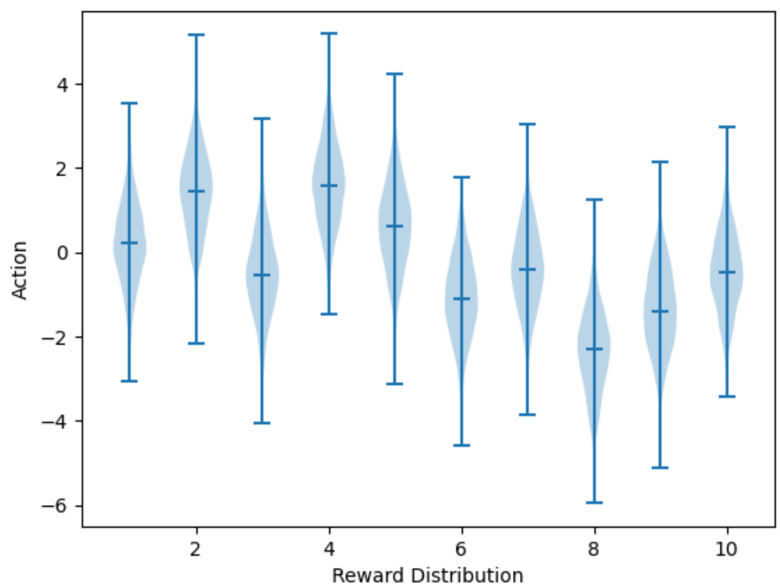
$$\therefore \lim_{n\to\infty} Q_{n+1} = 0 + \lim_{n\to\infty} \alpha \frac{(1-(1-\alpha)^n)}{1-(1-\alpha)} q_*$$

$$= q_*$$

Hence, it is unbiased, when $n\to\infty$

(e) First of all, we cannot implement $n\to\infty$ in practice, even if we can control Q_1 at the inial state. And the weighting on reward would be changed as more steps are taken. Therefore, in general, $Q_{n+1} \neq q_*$. the exponential recency-weighted average will be biased

4. Plot:



5. In the long run, $\varepsilon=0.01$ will perform best.

While $\varepsilon=0$, it is greedy method. which will never explore after it try each action once. It will always take action that has the highest estimated reward after trying the first time.

While $\varepsilon=0.1$ or $\varepsilon=0.01$, it is ε-greedy method. which will explore more.

As a result, this will eventually performed better because they continue to explore and improve their chance of recognizing the optimal action. The $\varepsilon=0.01$ method improved slowly but eventually will perform better than the $\varepsilon=0.1$ method.

$$\text{Optimal Action} = (1-0.1)\times 1 + 0.1\times\frac{1}{10} = 0.91 \text{ . when } \varepsilon=0.1 \text{ ,}$$

$$\text{Optimal Action} = (1-0.01)\times 1 + 0.01\times\frac{1}{10} = 0.991 \text{ , when } \varepsilon=0.01 \text{ ,}$$

Hence, the $\varepsilon=0.01$ method selected the optimal action more.

6. The predicted asymptotic level of average reward should be:

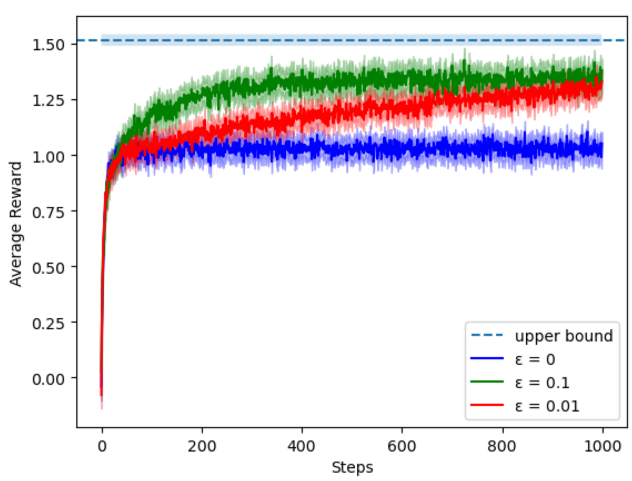
Upper bound $\times$ Optimal Action %

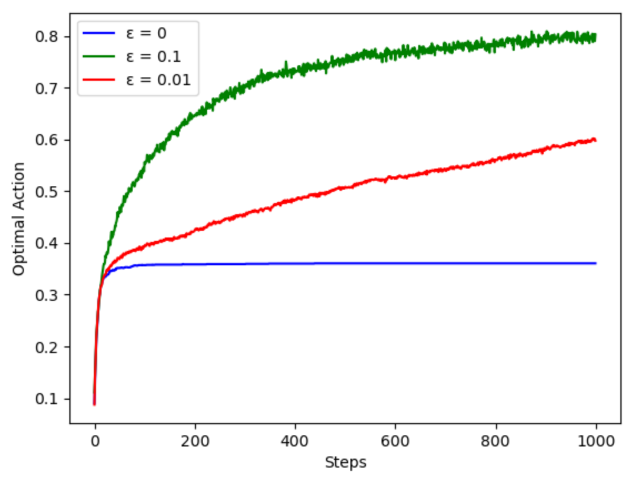
Therefore,

(a) the average reward of the $\varepsilon=0.1$ method reach the asymptotic level;

$1.5\times 0.91 = 1.35$

(b) But none of their optimal action percentage reach the asymptotic level as it didn't run long enough.





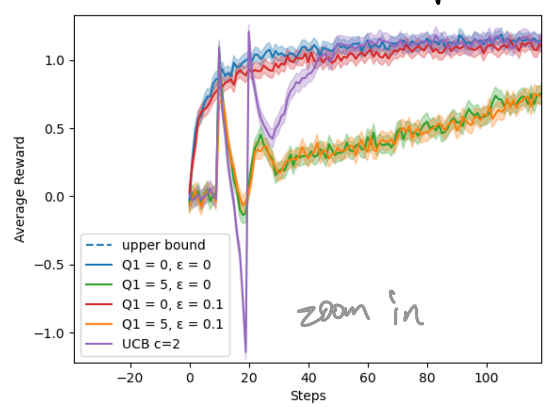
7. The spike in the very beginning is the result of initial exploration.

Analyze : (1) Sharp increase

For both optimistic initialization and UCB produce the first sharp increased spike at around the 10th step. Because the first 10 steps, the algorithm had taken all actions once and find out the estimated higher reward action.

(2) Sharp decrease

As we zoom in the first 40 steps, we noticed that the optimistic initializations tend to smooth out after the first spike. However, UCB method continue to drop until the 20th step.



That is because UCB will explore actions that have been selected less frequently relative to the total steps. The exploration term will dominate between the 10th and 20th step. Therefore, it will select the lower reward action that result in a drop.