RL Homework 1

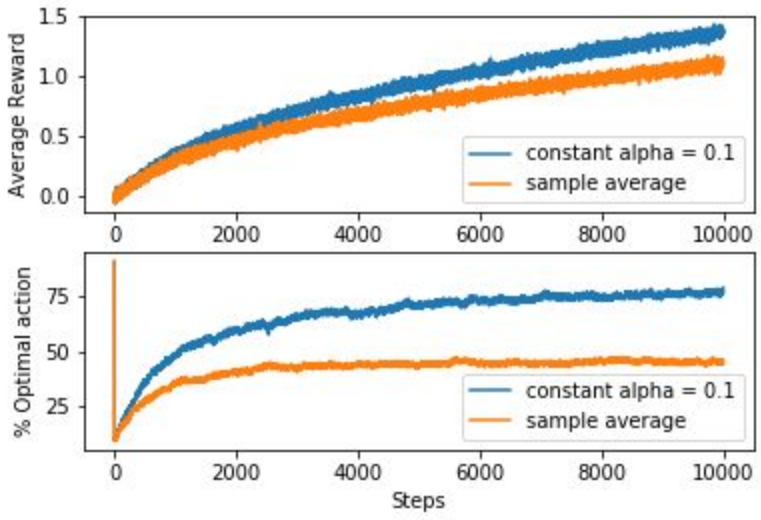
Submitted By:
Snehal Gupta
2016201

Q1

Experiment:

- For sample average method, Q*(a) was initialized with random numbers sampled from the standard normal distribution of mean 0 and variance 1.
- For constant alpha method, Q*(a) was initialized with 0.
- Q(a) was initialized to zero for both the cases.
- Epsilon e = 0.1 and alpha = 0.1
- A random number is picked from a uniform distribution (0,1) in order to decide whether to explore or choose action greedily. If the value is above epsilon, the action is chosen greedily, else, randomly.
- In the case of greedy selection, action with the highest value of Q(a) is chosen.
- After the action is selected, a reward is randomly selected from a normal distribution of mean = Q*(At) and standard deviation 1 where At denotes the action selected at time t.
- Q(At) is updated using the incremental implementation.
- Since its a non-stationary problem, after every time step, Gaussian noise of mean 0 and standard deviation 0.01 is added to Q*(a) of every action.
- This process is repeated for 2000 runs of 10000 time-steps.

Plot:

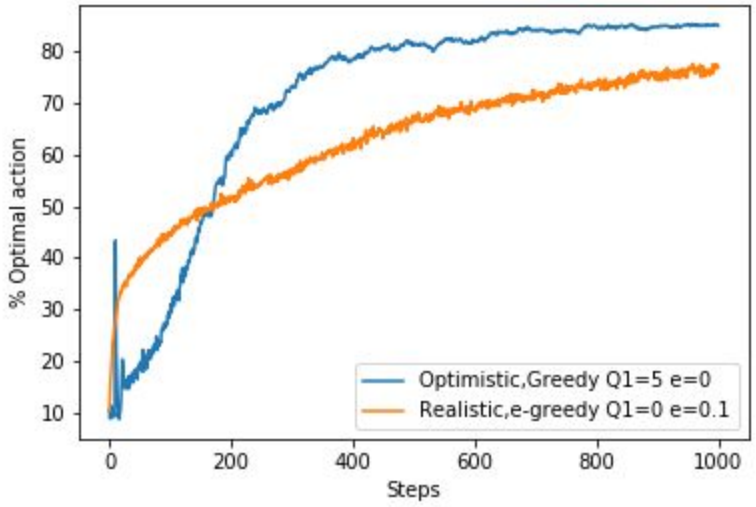


Observations:

- The constant alpha method performs better than the sample-average method. This is because, in the constant alpha method, more weight is given to recent rewards than to long-past awards.
- A spike at t=0 is observed in the optimal action % graph. In case of greedy selection, the action is selected using np.argmax() which gives the first occurrence of optimal action. This is the reason behind the spike in optimal action graph at time step=0. Since Q*(a) is initialized to 0 for all actions at t=0, any action selected would be the optimal action.

Q2

The output of generating Fig 2.3 (Stationary case):

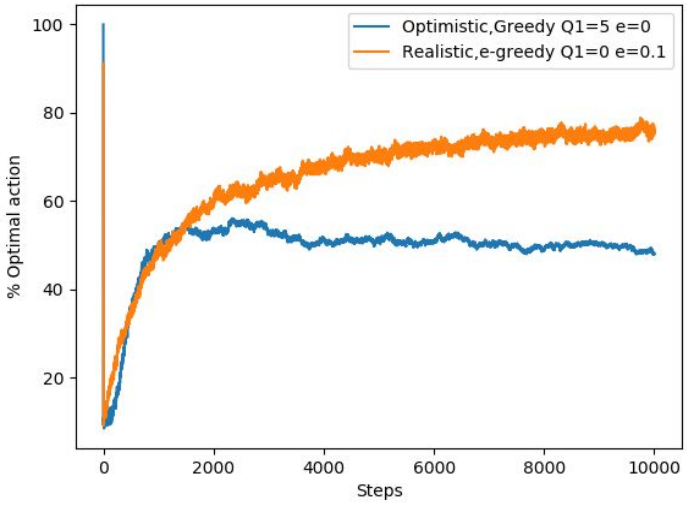


Exercise 2.6

In the optimistic case, Q(a) is initialized to 5 for all actions. Since Q*(a) is initialized with a normal distribution of mean 0 and variance 1, initialization of Q(a) with 5 is highly optimistic. That is why in early steps, the selection of any action leads to a decrease in Q(a) of the selected action. This is because the reward is less than the initial estimate of Q(a). So, the method tends to explore more than realistic epsilon-greedy method. Hence, the optimistic method performs worse in the beginning but performs better in the long run since all actions are selected several times before the estimates converge.

The reason for spikes in early steps is that the optimal action is selected exactly at the same time step in every run. All actions are tried once until t=11 since Q(a) decreases after the action gets selected. After trying out all actions once, an optimal action is selected greedily at the same time step t=11 for every run. However, since the reward is still less than the estimated Q(a), different actions start getting selected for every run after t=11.

For non-stationary case:



The spike at t=0 is because we have initialized Q*(a) with 0 and Q(a) with the same value (either 5 or 0). Hence, any action selected at t=0 would be an optimal action when we are taking the first occurrence of optimal action into account.

In non-stationary case, the realistic e-greedy method performs better than optimistic method. This is because, in non-stationary problem, the agent will stop exploring once the value estimates have started to converge to their true values. As mentioned in the book, “the drive for exploration is inherently temporary”.

Q3

Q3

Yes, it is possible to avoid the bias of constant step sizes while retaining their advantages on non-stationary problems.

Proposed Method (As mentioned in book)

Use step size $\beta_n = \frac{\alpha}{\bar{o}_n}$ where $\bar{o}_n = \bar{o}_{n-1} + \alpha(1 - \bar{o}_{n-1})$ for $n \geq 0$ with $\bar{o}_0 = 0$

Analysis

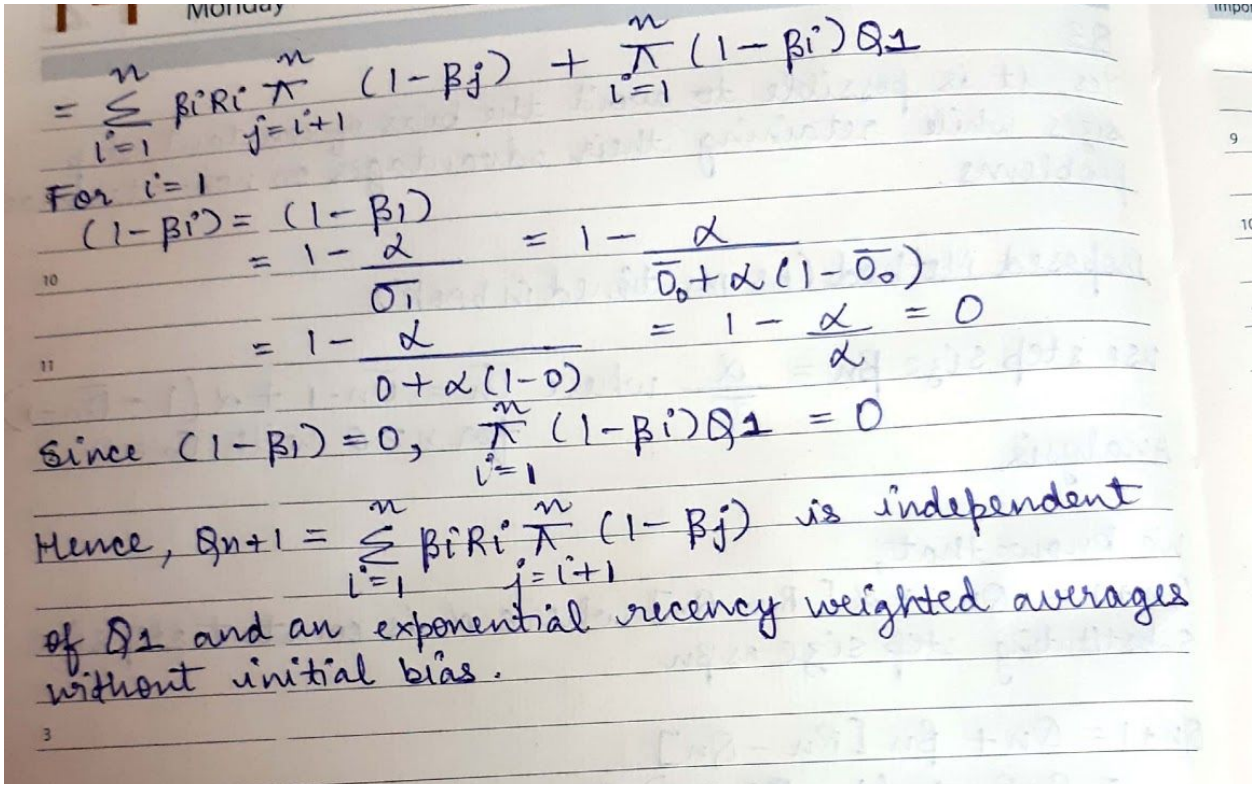
We know that,

$Q_{n+1} = Q_n + \alpha[R_n - Q_n]$ where α is constant step size

Substituting step size as β_n

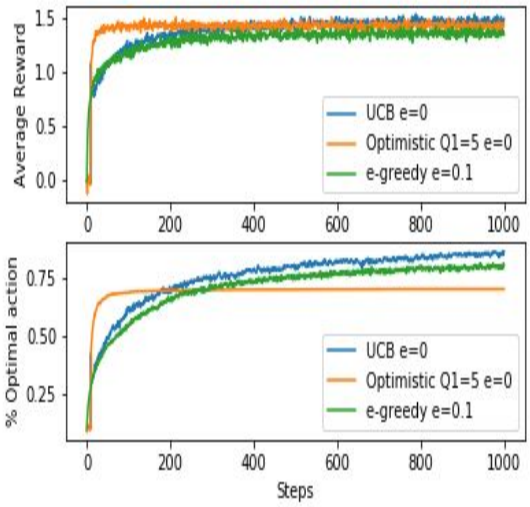
$$Q_{n+1} = Q_n + \beta_n [R_n - Q_n]$$
$$= \beta_n R_n + (1 - \beta_n) Q_n$$
$$= \beta_n R_n + (1-\beta_n)[Q_{n-1} + \beta_{n-1}[R_{n-1} - Q_{n-1}]]$$
$$= \beta_n R_n + (1-\beta_n)[R_{n-1}\beta_{n-1} + (1-\beta_{n-1})Q_{n-1}]$$
$$= \beta_n R_n + (1-\beta_n)[R_{n-1}\beta_{n-1} + (1-\beta_{n-1})(Q_{n-2} + \beta_{n-2}[R_{n-2} - Q_{n-2}]]$$
$$= \beta_n R_n + (1-\beta_n)[R_{n-1}\beta_{n-1} + (1-\beta_{n-1})[\beta_{n-2}R_{n-2} + (1-\beta_{n-2})Q_{n-2}]$$
$$= \beta_n R_n + (1-\beta_n)[R_{n-1}\beta_{n-1} + (1-\beta_{n-1})\beta_{n-2}R_{n-2} + (1-\beta_{n-1})(1-\beta_{n-2})Q_{n-2}]$$
$$= \beta_n R_n + (1-\beta_n)R_{n-1}\beta_{n-1} + (1-\beta_n)(1-\beta_{n-1})\beta_{n-2}R_{n-2} + (1-\beta_n)(1-\beta_{n-1})(1-\beta_{n-2})Q_{n-2}$$

256-109 Sunday 13

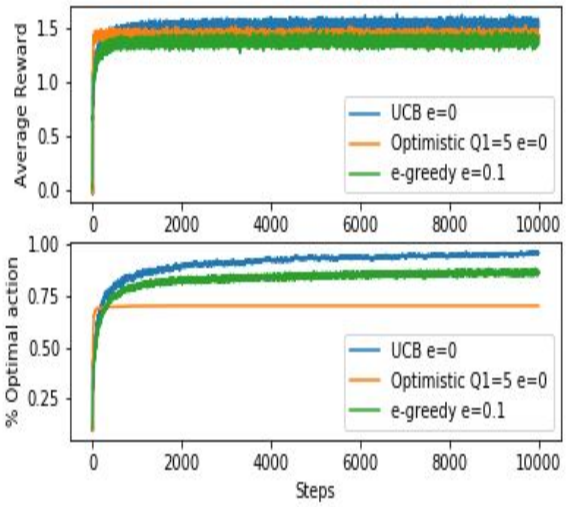


Q4

Stationary case

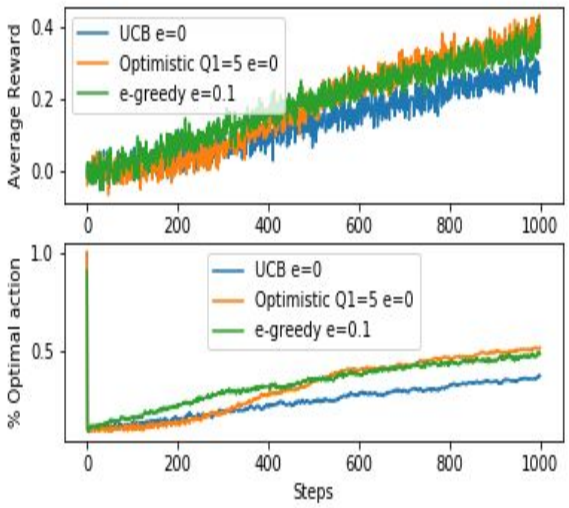


1000 steps

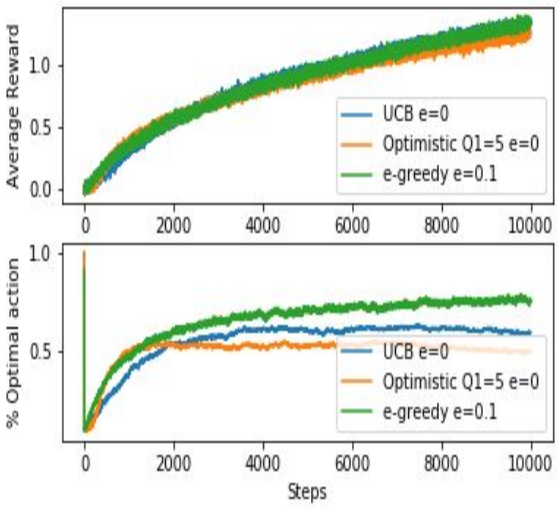


10000 steps

Non-stationary case



1000 steps



10000 steps

For stationary case, UCB performs better than Optimistic value and e-greedy. This is because, as mentioned in the book, it “selects among the non-greedy actions according to their potential for being optimal, taking into account both how close their estimates are to being maximal and the uncertainties in those estimates”. It also ensures that every action is selected once in a while, which leads to a better result.

For non-stationary case, in early steps, UCB performs worst. However, in later steps, it performs better than optimistic and poorer than e-greedy. This is because, in early steps, uncertainty is more since Nt(a) is close to 0. After some amount of steps, certainty is gathered. Also, since the term for uncertainty assumes stationarity, the actions are not selected in a legitimate manner. In the long run, all three methods perform almost in a similar manner, because of Q*(a) changing at each time-step.