Reinforcement Learning

Assignment 1 Multi-Armed Bandits

Prashant Pathak (MT19051)

INTRODUCTION

Consider a situation where we are given k different options (arm). Based on our choice we will receive a reward from the distribution of that particular arm. Distribution of the arm can be stationary as well as non-stationary. Our objective is to maximize the expected total reward over some period of time, say over 1000 time steps. This is called the Multi-armed bandit problem.

Each of the arms has expected reward given that action is chosen denoted by $q_*(a)$.

$$\mathbf{q_*(a) = E[R_t \mid A_t = a]}$$

Where R_t is the reward at time t and A_t is the action chosen at time t.We try to estimate the q_* in different ways for ex: with the help of sample mean and based on these estimates we try to choose the arm that maximizes our objective.

Question 1: k arm bandit with var = 1

In this question, we are asked to implement a 10 arm bandit problem.

- For the estimate of the action values (denoted by $Q_t(a)$) of the arm, we choose the **Sample mean.** The sample mean is implemented in an incremental way:

  $$Q_{t+1} = Q_t + (1/t)\ [R_t - Q_t]\ \text{(where t denote the timestep)}$$

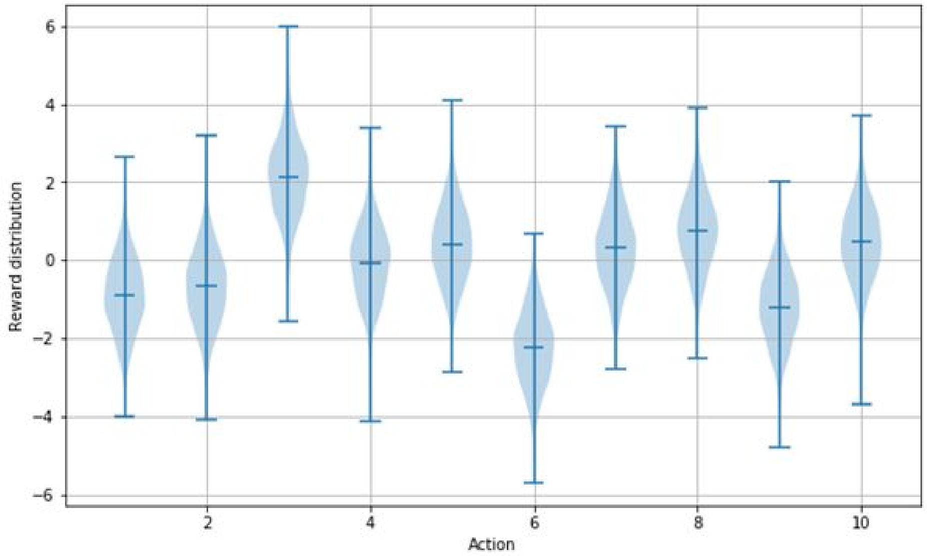
- For selecting the arm we have a different option
  - **Greedy**: In which we choose the arm with the maximum estimate i.e $\mathbf{A_t = argmax_{\ a} (Q_t)}$
  - **∈ Greedy Policy:** In this, with (1- ∈) we choose the best arm and with ∈ we choose any arm at random.

    At = ∈/m + (1-∈) , if a = $\text{argmax}_{(b \in A)} Q_t(b)$

    ∈/m, otherwise

Experiment: To see which selection policy perform well

- Number of arms: 10
- Each arm has its own distribution. **Mean of each arm is sampled from a standard normal distribution and variance is set to 1**. Below figure the arms' distribution.



- For values of ∈ we have chosen **0.0, 0.01 and 0.1 and epsilon decay.**
- In Epsilon decay we have taken values two values:
  - Decayed with (1/n) with follow equation 2.7
  - Decayed using following formula

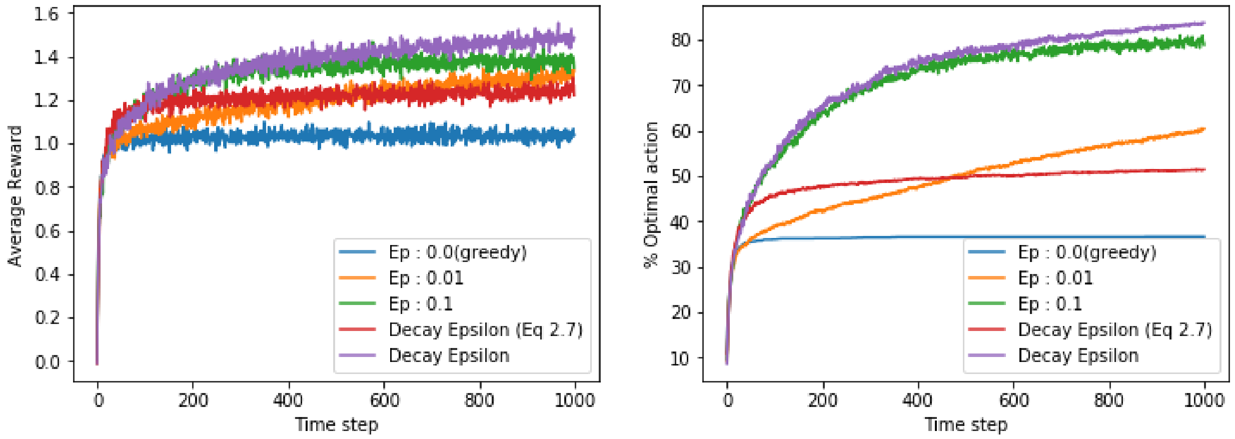
    Choose e_{init} and e_{end}

    r = max((N - step taken)/N , 0)

    Ep = r * (e_{init} - $e_{end)}$) + e_{init}
- 2K independent bandit problem experiment is performed to see the average effect.

Results :

a) Graph for Average reward and % Optimal action.



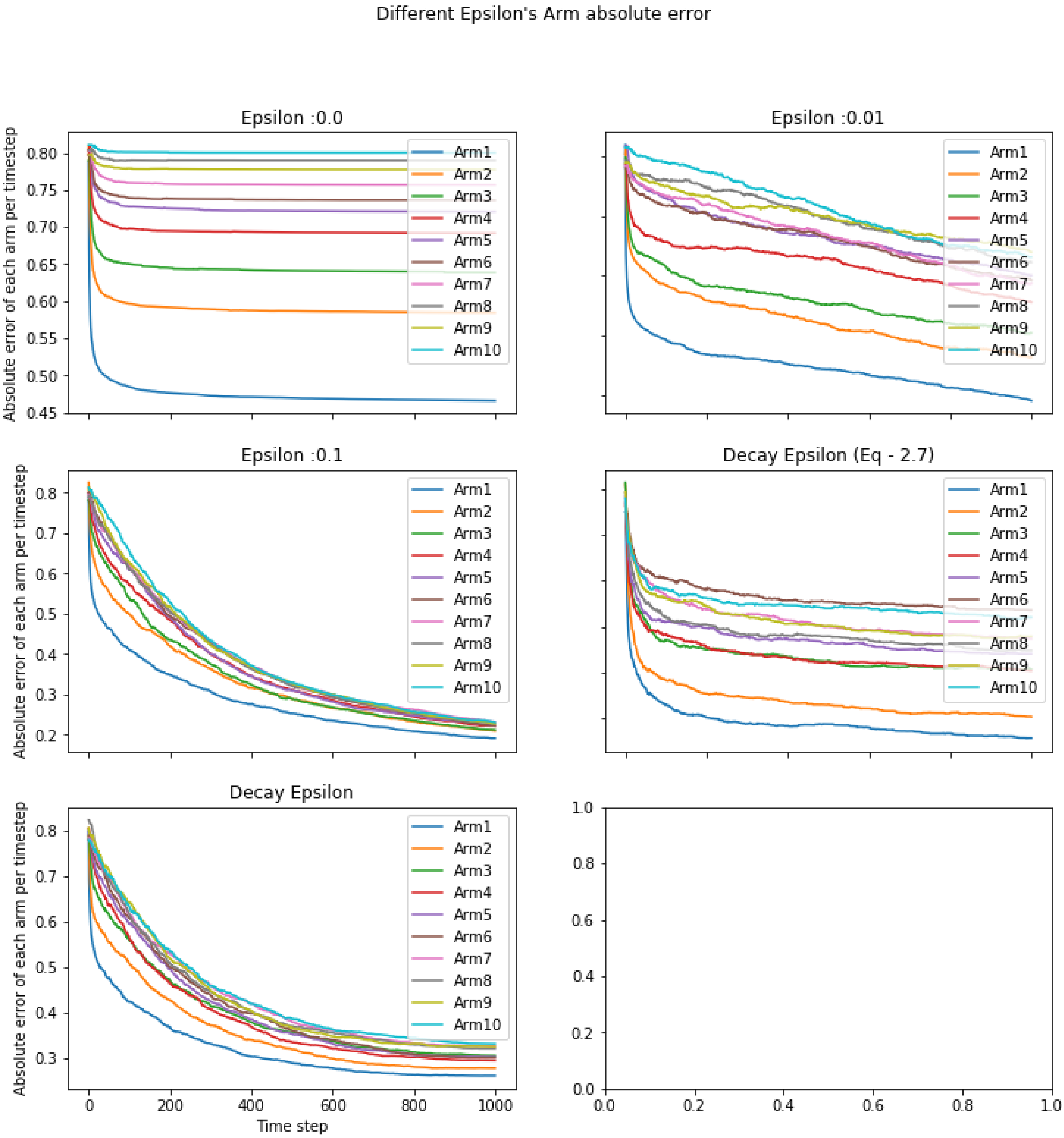
Conclusion:

In the RL problem, there is **Exploitation** and **Exploration.** In **Exploitation**, we always choose the best arm so as to maximize the cumulative reward. In **Exploration,** we explore a new arm so that we could get an arm that gives more reward than the current.

There is a trade-off between **Exploration** and **Exploitation. We can balance it with** ∈.

- When ∈ = 0, then we always exploit and do not explore any new arm so our average reward is not that good.
- When comparing ∈ = 0.01 and ∈ = 0.1 we see that ∈ = 0.1 performs better when we have timestep = 1000. ∈ = 0.1 explores more compared to 0.01 and finds the estimate that is close to the true value. But even after finding the best arm, it does not exploit it as ∈ is fixed. In the case of ∈ = 0.01, it explores with a slower rate but when it finds a closer estimate to true it exploits more compared to ∈ = 0.1. So in the longer run 0.01 will perform better.
- After the previous explanation, it is obvious that we need a way which first explores and then exploits, so decay epsilon comes into the picture. So initially we can have high-value epsilon and then we can slowly decay it so that it exploits more. (1/n) epsilon decays exponentially so it performs not that good.
- With selecting the decay value with precaution we can have it work better than constant epsilon values.

b) Average absolute error in the estimate for each action (arm) as a function of time steps:



Conclusion

As stated earlier for estimating the q values we have taken sample means of each arm's reward.

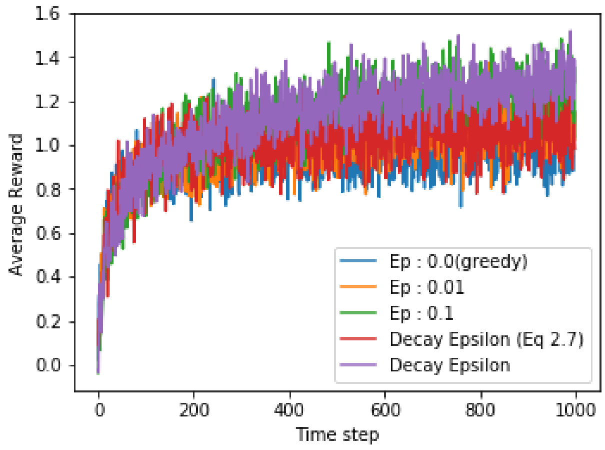
$$Q_t(a) = (r1 + r2 + ... r_{t-1})/t-1$$

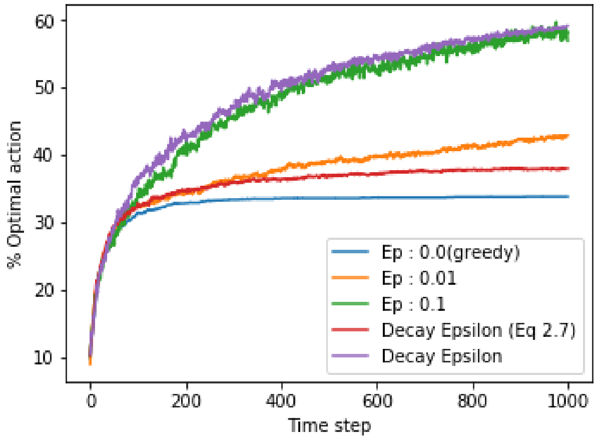
By the law of large numbers, Sample mean reaches the true value when we have a lot of samples. When we will try an arm many times then its absolute error will reduce.

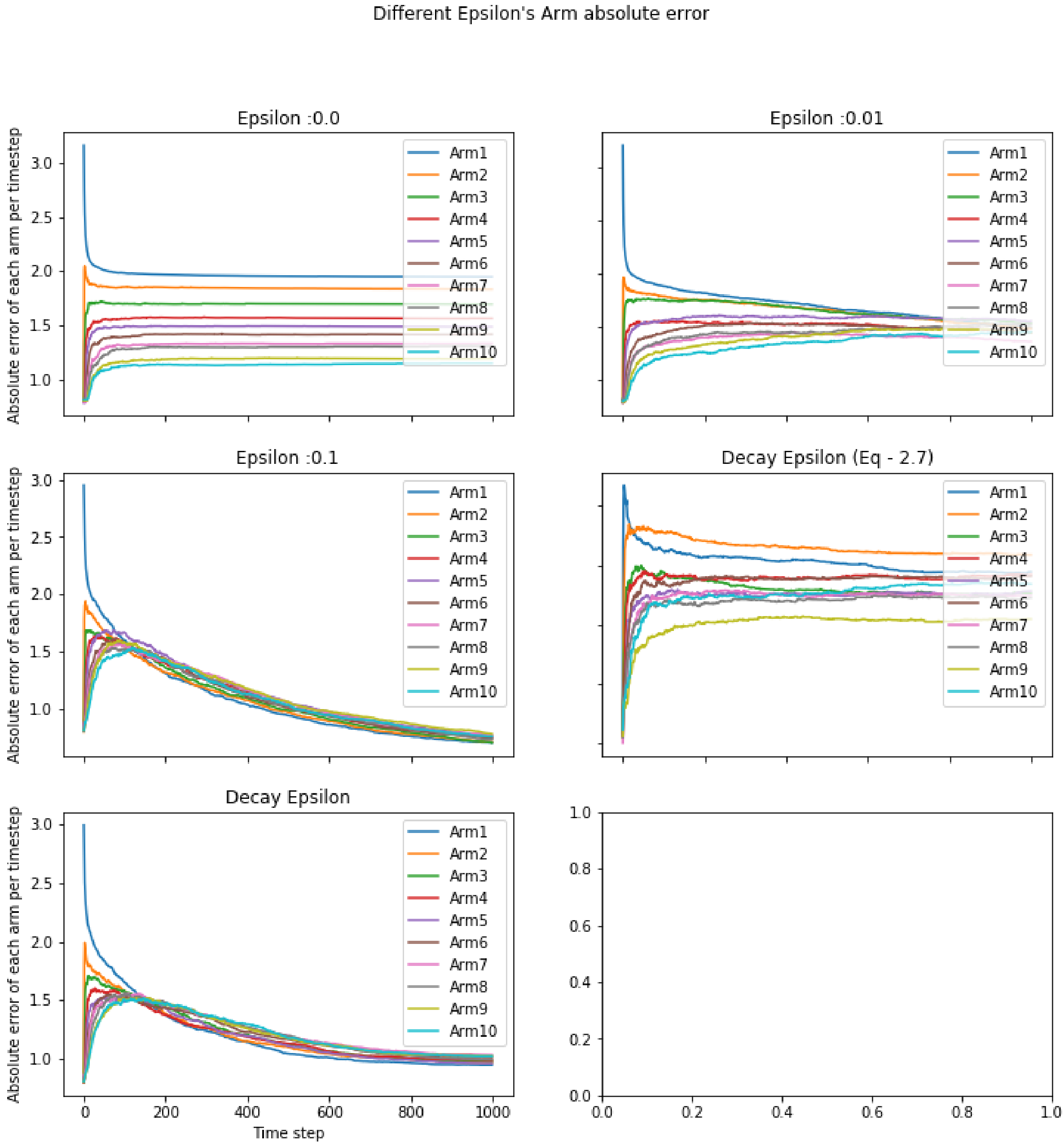
- In the case of greedy only one arm is tried many times so only one value reduces and that can be seen from the graph.
- Between ∈ = 0.01 and ∈ = 0.1, the ∈ = 0.1 explores more so all the arm’s absolute error is low in the latter case.
- In the decaying case also all the arms are initially tried so all arm error is reduced.

Question 2: k arm bandit with variance 4

The above experiment repeated with each arm’s variance = 4 and below is the results.







Conclusion:

- ❏ Even with a higher variance of the arms distribution, the trend is the same for all different action selection ways but the chances are that as the variance is high the reward variance is also high.
- ❏ For a few initial time steps we see that error increases because as the variance is high it takes few time steps to reach close to mean. For the first few steps as the variance is high the values that are samples varies a lot so error is high.

Question 3 Exercise 2.3

Question : In the comparison shown in Figure 2.2, which method will perform best in the long run in terms of cumulative reward and probability of selecting the best action? How much better will it be? Express your answer quantitatively.

Solution: If an epsilon is decayed in a proper way it can perform better than constant epsilons. But the best epsilon decay does not follow eq 2.7 so we will not consider that in this question. So we evaluate among the remaining values of epsilon we i.e 0.0, 0.01,0.1 and decay epsilon according to 1/n. Among the rest, epsilon 0.01 will perform better in the longer run because after the estimate reaches near the true values it does not explore much in later stages compared to others.

For quantitative analysis we will consider arm true values i.e form arm distribution graph shown above.

$m = \max_{(a \in A)} q_*(a)$ => $m = 2$ (from the graph)

For greedy : we can see from the graph that the expected reward it obtains is approx 1. Greedy action does not explore it, just exploit it. From the absolute error graph we can observe that in ep = 0, arm 1 error is reduced very much . so arm1 is tried many times. And so expected reward from greedy policy is around 1. From the graph of (% optimal action) we can see the optimal arm is selected in initial phases only and later only the greedy arm is selected.

For epsilon 0.01 :
For Expected reward in the long run: with probability of .99 we will select the best arm and with probability of 0.01 we will select any arm at random.

As the mean of the arm is taken from a standard Normal distribution the random selection of any arm will give an expectation close to 0 approx. So only the first term will constitute the final expectation.
Expected reward in longer run :
$E[R] = q_*(\text{best arm}) * \text{prob of selecting best arm}$
$q_*(\text{best arm}) = 2.$
$E[R] = 2 * .99 = 1.98$

Probability of selecting best arm = $(1- \in) + \in/\text{no of arm} = (1 - 0.01) + (0.01/10) = .99 + .001 = .991$ i,e **99.1% (best choice)**

For epsilon 0.1
For Expected reward in the long run: with probability of (1-0.1) = .9 we will select the best arm and with probability of 0.01 we will select any arm at random .Again the latter term is zero because of reasons told in the previous section.

Expected reward in longer run :
E[R] = q_*(best arm) * prob of selecting best arm
q_*(best arm) = 2.
E[R] = 2 * .9 = 1.8

Probability of selecting best arm = (1- ∈) + ∈/no of arm = (1 - 0.1) + (0.1/10) = .9 + .01 = .91 i,e **91%**

For decay epsilon following eq 2.7:
We can see that it do not perform well in the graph but is decay factor is chosen properly it works the best

Question 4 Initial choice of estimate and alpha

Question: Show that the sample mean is not influenced by the initial choice of Q1(a), ∀a, where as when using a constant step-size α (see Equation (2.5)) the estimate Qt(a) is a function of Q1(a). Also, show that the dependence is larger for a smaller α. Propose a method such that we can have a constant step-size but no dependence of Qt(a) on Q1(a).

Solution:

$$Q_{t+1}(a) = Q_t(a) + (1/t)\ [R_t - Q_t(a)]\ \text{ a belong to all action}$$

Lets see ~~for~~ when we calculate

$$Q_2(a) = Q_1(a) + \frac{1}{1}[R_1 - Q_1(a)]$$

'a $\in A$

$$\Rightarrow Q_2(a) = \cancel{Q_1(a)} + R_1 - \cancel{Q_1(a)}$$

$$= R_1$$

So we can see our $Q_2(a)$ $a \in A$ estimate is independent of initial value.

Now lets take it α

$$Q_{t+1}(a) = Q_t(a) + \alpha[R_t - Q_t(a)]$$

$a \in A$

Rewriting this

$$Q_{t+1}(a) = \alpha R_t + (1-\alpha) Q_t$$

On Expanding we get,

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

From the last equation we can see that if Q1 is multiplied by (1-alpha)^n so that the dependence is larger for a smaller α .

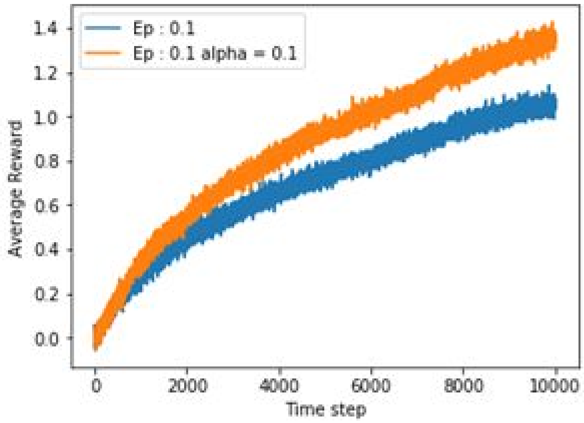
A method such that we can have a constant step-size but no dependence of $Q_t(a)$ on $Q_1(a)$ is to **set $Q_1(a)$ to 0 (zero)**. If it is set to zero then the first term in the last equation comes to zero and $Q_t(a)$ independent of $Q_1(a)$.

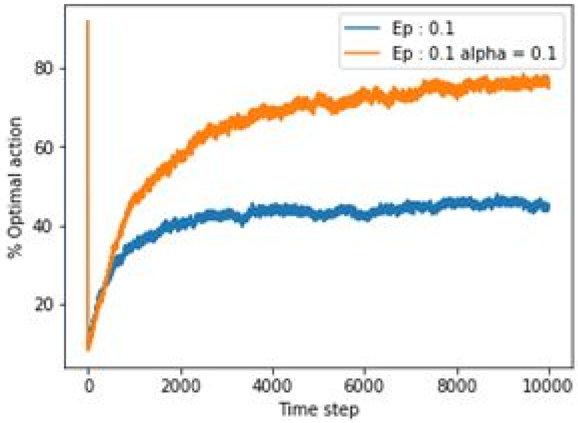
Question 5 Non-stationary k arm bandit problem.

In this question, we have to implement a non-stationary k arm bandit problem.

Experiment and Result

- Initially, we have kept the mean of all the arms to be the same which is drawn from a standard normal distribution.
- After that, we change the mean of each arm by adding a small random number to each arm's mean. A small number is taken from a normal distribution with 0 mean and 0.01 variance





Conclusion:

- ❑ When computing the sample mean we use below formula

$$Q_{t+1} = Q_t + (1/t)\,[R_t - Q_t] \text{ (where t denote the timestep)}$$

We can see that this formula has a factor of (1/t) multiplied with error term, so when computing the estimate it is giving some weight to all the previous rewards sampled till now. In case of non stationary settings as the distribution is continuously changing so sample means do not work better.

- In the above equation if we use constant alpha in place of (1/t) it is beneficial for non stationary setting

$$\begin{aligned} Q_{n+1} &= Q_n + \alpha\Big[R_n - Q_n\Big] \\ &= \alpha R_n + (1-\alpha)Q_n \end{aligned}$$

As we can see alpha can be used to decide how much weight to be given to current reward and how much to previous mean.

- The above points are clear with graphs as constant alpha performs better.

Question 6 : Upper Confidence Bound (UCB) Action Selection

- Idea behind UCB is that we should select that arm more frequently which is more uncertain. Uncertainty means we are not sure about the reward we get from them.
- Naturally the arm that is less tired will have more uncertainty.
- So with the estimate of the arm we also add an uncertainty value that tells which arm has high uncertainty and then we choose that arm.
- Mathematically,

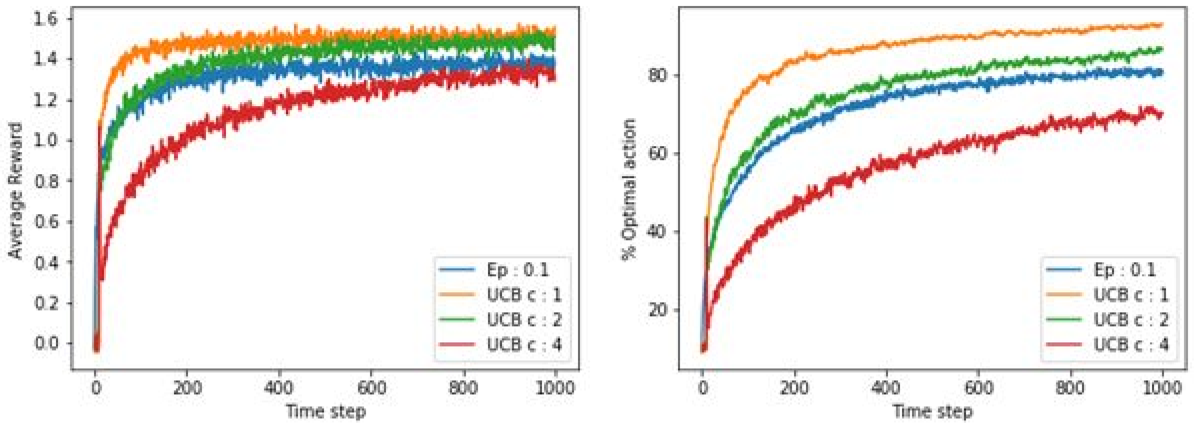
$$A_t \doteq \operatorname*{argmax}_a \left[Q_t(a) + c\sqrt{\frac{\ln t}{N_t(a)}} \right]$$

First term is the estimate and second term is the uncertainty.

- We can see that uncertainty is inversely proportional to the number of times an arm is selected.

Experiment and result :

We tries 10 arm bandit with UCB with different c to observes the pattern



Conclusion

- We can see that when proper c values are chosen then the UCB method works better than ∈- greedy method.
- C > 0 control the degree exploration.There must be a trade off between Exploration and exploitation so in our case c = 1 work better.

Exercise 2.8 of Sutton and Barto

Question : Suppose you face a 2-armed bandit task whose true action values change randomly from time step to time step. Specifically, suppose that, for any time step, the true values of actions 1 and 2 are respectively 0.1 and 0.2 with probability 0.5 (case A), and 0.9 and 0.8 with probability 0.5 (case B). If you are not able to tell which case you face at any step, what is the best expectation of success you can achieve and how should you behave to achieve it? Now suppose that on each step you are told whether you are facing case A or case B (although you still don't know the true action values). This is an associative search task. What is the best expectation of success you can achieve in this task, and how should you behave to achieve it?

Solution :
Case 1
When we do not the situation we are in we can only act according to overall return. So first we have to find the expectation of reward for both arms.

$$E(X) = \sum_{x \in X} (p(x) * x)$$

For arm 1

x	0.1	0.9
p(x)	0.5	0.5

So E(arm 1) = 0.5 * 0.1 + 0.9*0.5 = 0.5

For arm 2 :

x	0.2	0.8
p(x)	0.5	0.5

So E(arm 2) = 0.5* 0.2 + 0.5 * 0.8 = 0.5
As both are the same the best expectation of reward we can get is 0.5 and to get it we can choose any.

Case 2:
As we know the situation we can store independent estimates of both case A and B. **We learn the best action in case A is Arm 2 and in case B is Arm 1.** Now for the overall estimate. With 0.5 prob we choose Arm 1 and get a reward of 0.9 and with 0.5 we choose Arm 2 and get 0.2. So expectation = 0.5*0.9 + 0.5 * 0.2 = **0.55**

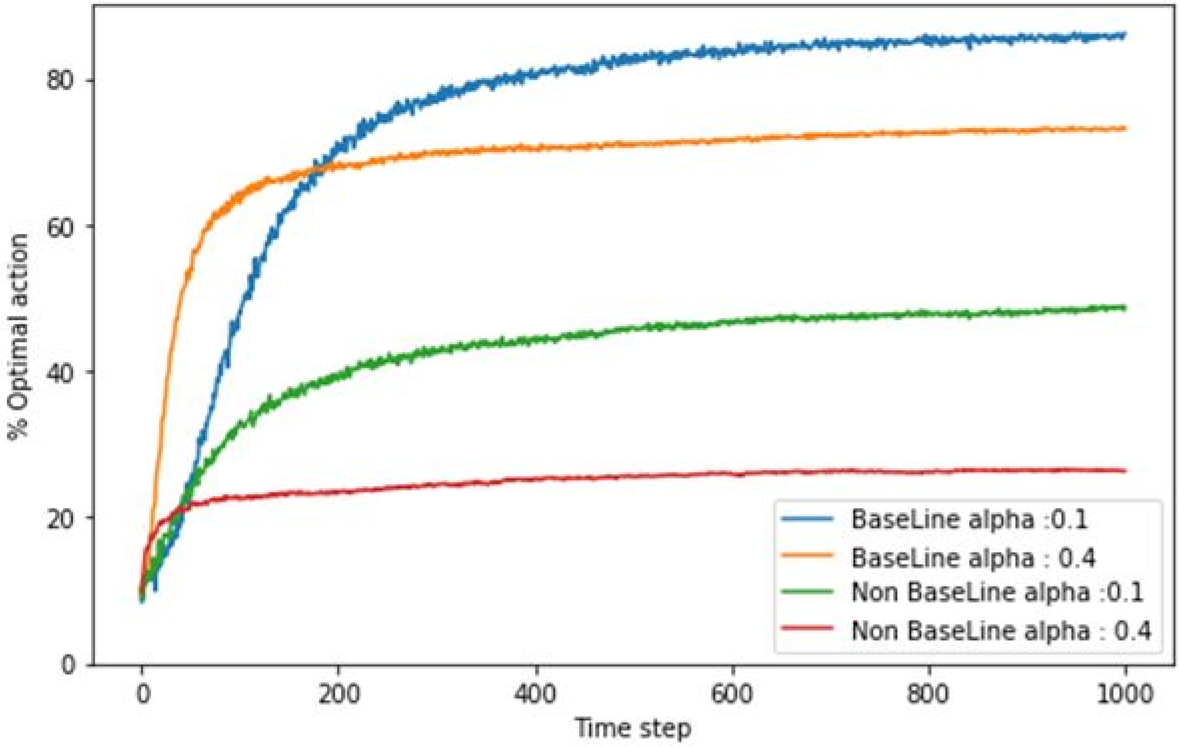
Question 7 Gradient based Bandit

- In gradient based methods we use stochastic gradient descent to solve the problem.
- Instead of estimating the action values we have an action preference number denoted by $H_t(a)$. We apply softmax on the action preference values to get a probability mass function over the action.
- We select action based on this PMF and then update the $H_t(a)$ by below formula:

$$H_{t+1}(A_t) \doteq H_t(A_t) + \alpha\big(R_t - \bar{R}_t\big)\big(1 - \pi_t(A_t)\big), \qquad \text{and}$$
$$H_{t+1}(a) \doteq H_t(a) - \alpha\big(R_t - \bar{R}_t\big)\pi_t(a), \qquad \text{for all } a \neq A_t,$$

Experiment and Result

- For initial distribution of the arm mean values are chosen from normal distribution of mean 4 and variance 1.
- Initially all the H values are set to 0.


% Optimal action
0
20
40
60
80
0
200
400
600
800
1000
Time step
BaseLine alpha :0.1
BaseLine alpha : 0.4
Non BaseLine alpha :0.1
Non BaseLine alpha : 0.4