Multi-armed Bandits

1. A k-armed Bandit Problem

Consider the following learning problem. You are faced repeatedly with a choice among k different options, or actions. After each choice you receive a numerical reward chosen from a stationary probability distribution that depends on the action you selected. Your objective is to maximize the expected total reward over some time period, for example, over 1000 action selections, or time steps.

- Each of the k actions has an expected or mean reward given that that action is selected (value of the action)
- Action selected on time step t as A_t
- Reward of time step t as R_t
- Value of an arbitrary action a, $q_*(a)$, is the expected reward given that a is selected:

$$q_*(a) \doteq \mathbb{E}[R_t \mid A_t = a] \quad \forall a \in \{1, \dots, K\}$$

$$= \sum_r p(r|a) \cdot r$$

- Estimated value of action a at time step t as $Q_t(a)$
- Would like $Q_t(a)$ to be close to $q_*(a)$
- Greedy actions: When you select the action with the greatest estimated value. We say that you are exploiting your current knowledge of the values of the actions
- Nongreedy actions: When you select the action with the greatest estimated value. We say that you are exploring, because this enables you to improve your estimate of the nongreedy action's value.

2. Action-value Methods

$$Q_t(a) \doteq \frac{\text{sum of rewards when a taken prior to t}}{\text{number of times a taken prior to t}}$$

$$\therefore Q_t(a) = \frac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i=a}}{\sum_{i=1}^{t-1} \mathbb{1}_{A_i=a}}$$

where $\mathbb{1}_{A_i=a}$ denotes the random variable that if $A_i=a$ is equal to 1 and than if not is 0.

* If the denominator is zero, then we instead define $Q_t(a)$ as some default value, such as 0.

- As the denominator goes to infinity, by the law of large numbers, $Q_t(a)$ converges to $q_*(a)$:

$$\lim_{t \to \infty} Q_t(a) = \frac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i = a}}{\sum_{i=1}^{t-1} \mathbb{1}_{A_i = a}} = \mathbb{E}[R_t \mid A_t = a] = q_*(a)$$

- The greedy action will be described as:

$$A_t \doteq \arg\max_a Q_t(a)$$

- ε-greedy: Behave greedily most of the time, but every once in a while, say with small probability ε, instead select randomly from among all the actions with equal probability.

- Probability of selecting the optimal action converges to greater than $1-\varepsilon$ (asymptotic).

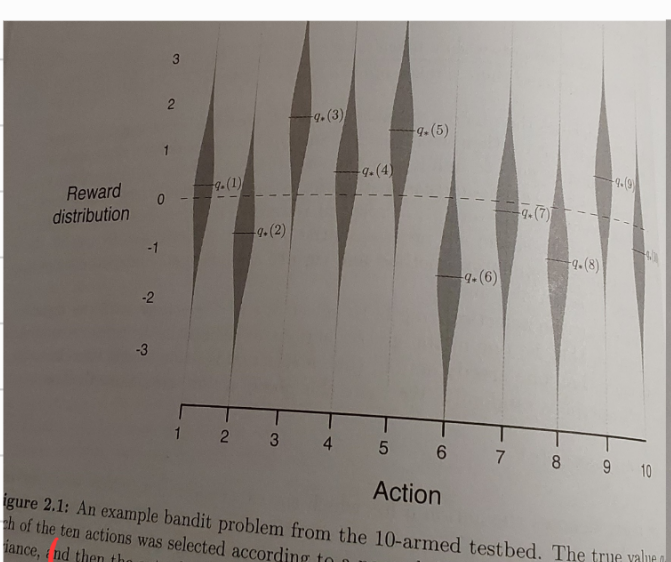
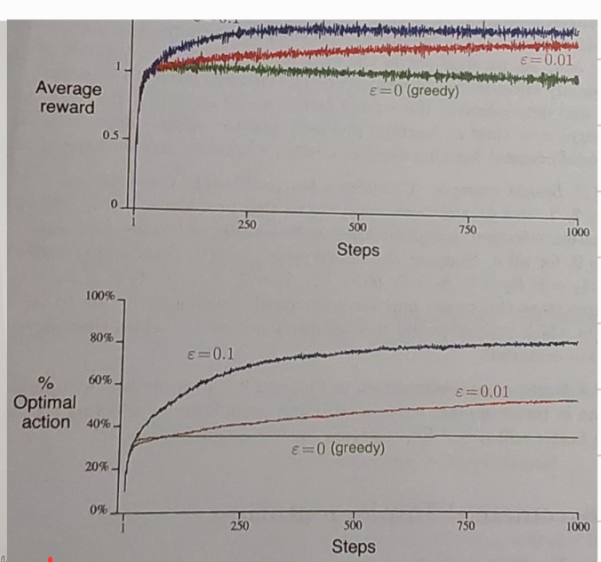


Figure 2.1: An example bandit problem from the 10-armed testbed. The true value of each of the ten actions was selected according to a normal distribution, and then the actual

→ The 10-armed Tesbed



→ Results

3. Incremental Implementation

↳ chapter 2.4

- Let R_i denote the reward received after the i^{th} selection of a single action and let Q_n denote the estimate of its action value after it has been selected $n-1$ times:

$$Q_n = \frac{R_1 + R_2 + \cdots + R_{n-1}}{n-1}$$

- The obvious implementation would be to maintain a record of all the rewards and then perform this computation whenever the estimated value was needed. But this is costly (memory) and inefficient.

- Given Q_n and the n^{th} reward, R_n, the new average of all n rewards can be computed by:

$$Q_{n+1} = \frac{1}{n} \sum_{i=1}^{n} R_i$$

$$= \frac{1}{n} \left(R_n + \sum_{i=1}^{n-1} R_i \right)$$

$$= \frac{1}{n} \left[R_n + (n-1) \frac{1}{(n-1)} \sum_{i=1}^{n-1} R_i \right]$$

$$= \frac{1}{n} \left[R_n + (n-1) Q_n \right]$$

$$Q_{n+1} = Q_n + \frac{1}{n} \left[R_n - Q_n \right]$$

So this expression stands for:

$$\text{NewEstimate} \leftarrow \text{OldEstimate} + \text{StepSize}\,[\text{Target} - \text{OldEstimate}]$$

This part represents an error in the estimate

- Pseudocode for a complete bandit algorithm using incrementally computed sample averages and ε-greedy action selection:
  * The function bandit(a) is assumed to take an action and return a corresponding reward.

Initialize, for $a = 1$ to K:

$Q(a) \leftarrow 0$

$N(a) \leftarrow 0$

Loop forever:

$$A \leftarrow \begin{cases} \text{argmax}_a Q(a), & \text{with probability } 1-\varepsilon \\ \text{a random action}, & \text{with probability } \varepsilon \end{cases}$$

$R \leftarrow \text{bandit}(A)$

$N(A) \leftarrow N(A) + 1$

$$Q(A) \leftarrow Q(A) + \frac{1}{N(A)}\,[R - Q(A)]$$

4. Tracking a Nonstationary Problem

- The most RL problems are nonstationary
- In such cases it makes sense to give more weight to recent rewards than to long-past rewards.
- One of the most popular ways of doing this is to use a constant step-size parameter:

$$Q_{n+1} \doteq Q_n + \alpha [R_n - Q_n]$$

where $\alpha \in (0,1]$ is constant.

→ step-size parameter

So we have that:

$$\begin{aligned} Q_{n+1} &= Q_n + \alpha [R_n - Q_n] \\ &= \alpha R_n + (1-\alpha) Q_n \\ &= \alpha R_n + (1-\alpha) [\alpha R_{n-1} + (1-\alpha) Q_{n-1}] \\ &= \alpha R_n + (1-\alpha)\alpha R_{n-1} + (1-\alpha)^2 \alpha R_{n-2} + \\ &\quad \dots + (1-\alpha)^{n-1} \alpha R_1 + (1-\alpha)^n Q_1 \end{aligned}$$

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

- The sum $S = (1-\alpha)^n + \underbrace{\sum_{i=1}^{n} \alpha(1-\alpha)^{n-i}}$ is equal:

Geometric Progression

$$S = (1-\alpha)^n + \frac{\alpha\left[(1-\alpha)^n - 1\right]}{(1-\alpha) - 1}$$

$$S = \frac{(1-\alpha)^{n+1} - (1-\alpha)^n + \alpha(1-\alpha)^n - \alpha}{-\alpha}$$

$$S = \frac{(1-\alpha)^n\left[(1-\alpha) - 1 + \alpha\right] - \alpha}{-\alpha}$$

$$\therefore \quad S = 1$$

Thus the expression above for Q_{n+1} is a weighted average, because the sum of the weights (S) is equal to one.

- As $(1-\alpha)$ is less than one, the contribution of the reward given on the i^{th} step, given by the expression $\alpha(1-\alpha)^{n-i} R_i$, increases over time/step.

- This is called an exponential recency-weighted average.

- If we have $\alpha = \alpha_n(n)$, i.e., α variable, in order to guarantee the convergence of the sample-average we most choose $\{\alpha_n(a)\}$ in a way that

$$\sum_{n=1}^{\infty} \alpha_n(a) = \infty \quad \text{and} \quad \sum_{n=1}^{\infty} \alpha_n^2(a) < \infty$$

Proof :

This is the problem of convergence to the Robbins-Monro algorithm.

i) From the definition we have:

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

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

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

$$= \dots$$

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

ii) Now, let a sthocastic variable R in a way that $\mathbb{E}[R_n] = R$ and $\mathbb{E}[\|R_n - R\|^2] = \sigma^2 > 0$, which follows the general assumption noises are zero-mean and independent and identically distributed, so we have:

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

deterministic error — random error

$$\mathbb{E}[\|Q_{n+1} - R\|^2] = \prod_{i=1}^{n}(1-\alpha_i)^2\|Q_1 - R\|^2 + \sum_{i=1}^{n}\alpha_i^2\prod_{j=i+1}^{n}(1-\alpha_j)^2\sigma^2$$

If we hope Q_{n+1} converges to R in quadratic mean, it's sufficient to have

$$\underbrace{\lim_{n\to\infty} \prod_{i=1}^{n} (1-\alpha_i)^2 = 0}_{\text{Condition 1}} \; ; \; \underbrace{\lim_{n\to\infty} \sum_{i=1}^{n} \alpha_i^2 \prod_{j=i+1}^{n} (1-\alpha_j)^2 < \infty}_{\text{Condition 2}}$$

So, from the first condition:

$$\lim_{n\to\infty} \prod_{i=1}^{n} (1-\alpha_i)^2 = 0 \rightarrow \lim_{n\to\infty} \log \prod_{i=1}^{n} (1-\alpha_i)^2 = -\infty$$

$$\rightarrow \lim_{n\to\infty} 2 \sum_{i=1}^{n} \log(1-\alpha_i) = -\infty \sim \boxed{\lim_{n\to\infty} \sum_{i=1}^{n} \alpha_i = \infty}$$

Finally, from the second condition

$$\lim_{n\to\infty} \sum_{i=1}^{n} \alpha_i^2 \prod_{j=i+1}^{n} (1-\alpha_j)^2 < \infty$$

But we have that

$$\lim_{n\to\infty} \sum_{i=1}^{n} \alpha_i^2 \prod_{j=i+1}^{n} (1-\alpha_j)^2 < \lim_{n\to\infty} \sum_{i=1}^{n} \alpha_i^2$$

So is sufficient that

$$\boxed{\lim_{n\to\infty} \sum_{i=1}^{n} \alpha_i^2 < \infty}$$

- The first condition is required to guarantee that the steps are large enough to eventually overcome any initial conditions or random fluctuations.
- The second one guarantees that eventually the steps become small enough to assure convergence.
- Both conditions are met for $\alpha_n(a) = \frac{1}{n}$, but not for $\alpha_n(a) = \alpha$, indicating that the estimates never completely converge but continue to vary in response to the most recently received rewards. This is desirable in a nonstationary environment.

5. Optimistic Initial Values

- Initial action values can be used as a simple way to encourage exploration.
- $q_*(a)$ in this problem are selected from a normal distribution with mean 0 and variance 1.
- Optimistic initial values encourages action-value methods to explore.
- The reward is less than the starting estimates; the learner switches to other actions, being "disappointed" with the rewards it is receiving.
- The system does a fair amount of exploration even if greedy actions are selected all the time.

- It is a simple trick that can be quite effective on stationary problems.

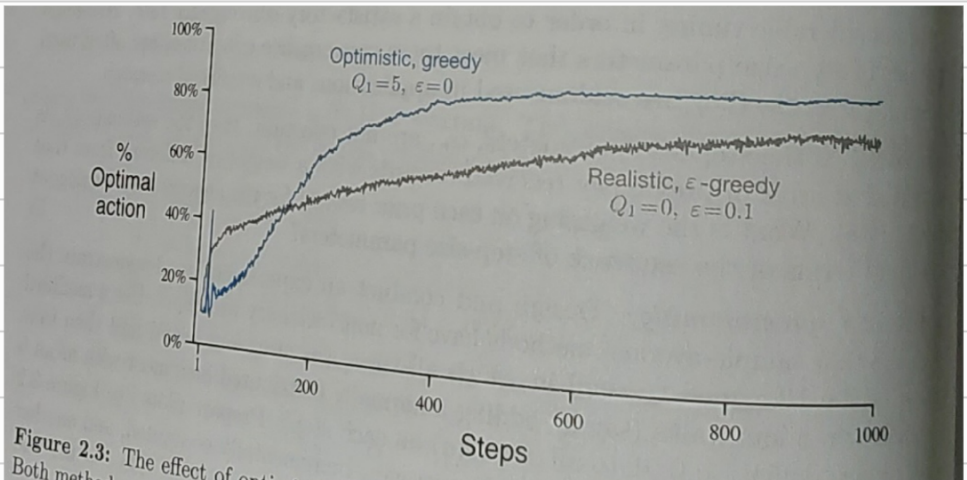


Figure 2.3: The effect of optimistic initial action-value estimates on the 10-armed testbed. Both methods used a constant step-size parameter, $\alpha = 0.1$.

6. Upper-Confidence-Bound Action Selection

- Exploration is needed because there is always uncertainty about the accuracy of the action-value estimates. The greedy actions are those that look best at present, but some of the other actions may actually be better.

- ε-greedy action selection forces the non-greedy actions to be tried, but indiscriminately.

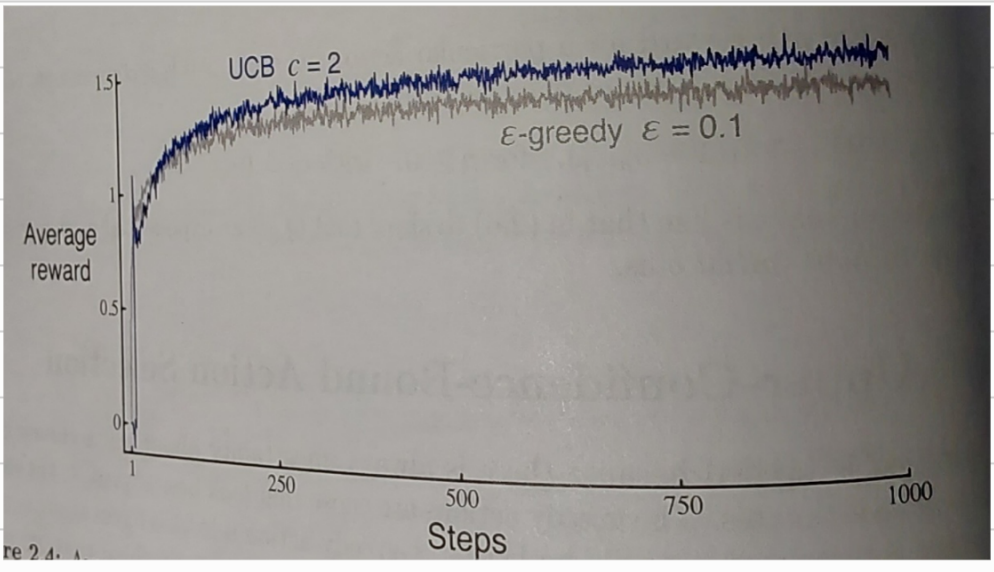
- Would be better to select among the non-greedy actions taking into account both how close their estimates are to being maximal and the uncertainties in those estimates.

- One effective way of doing this is to select actions according to:

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

where $N_t(a)$ denotes the number of times that action a has been selected prior to time t, and the number $c > 0$ controls the degree of exploration.

- The idea of this upper confidence bound (UCB) action selection is that the square-root term is a measure of the uncertainty or variance in the estimate of a's value.



7. Gradient Bandit Algorithms

- Consider learning a numerical preference for each action a, denoted $H_t(a) \in \mathbb{R}$.

- The larger the preference, the more often that action is taken.

- The preference, action probabilities, are determined according to a soft-max distribution (Gibbs or Boltzmann distribution):

$$\Pr\{A_t = a\} \doteq \frac{e^{H_t(a)}}{\sum_{b=1}^{K} e^{H_t(b)}} \doteq \pi_t(a)$$

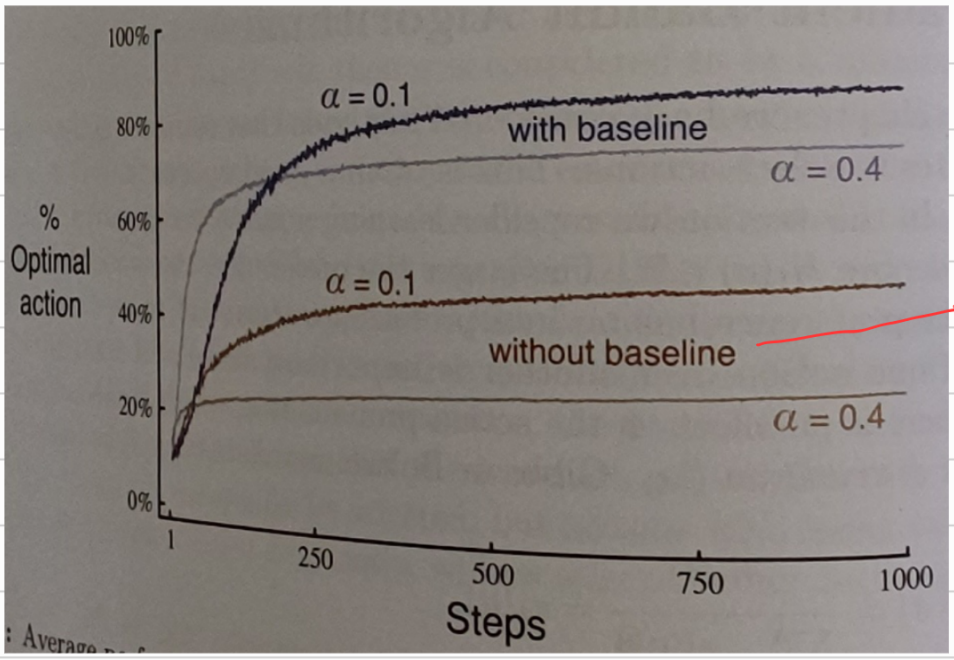
- Initially all action preferences are the same (e.g., $H_1(a) = 0, \forall a$).

- There is a natural learning algorithm for soft-max action preferences based on the idea of sthocastic gradient ascent:

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

where $\alpha > 0$ is a step-size parameter, $\bar{R}_t \in \mathbb{R}$ is the average of the rewards up to but not including time t (with $\bar{R}_1 \doteq R_1$).

→ Proof on the next page!



$\bar{R}_t = 0$

Proof of the algorithm!

(*) The Bandit Gradient Algorithm as a Sthocastic Gradient Ascent

- We can understand the bandit gradient algorithm as a sthocastic approximation to gradient ascent.
- In exact gradient ascent, each action preference $H_t(a)$ would be incremented in proportion to the increment's effect on performance:

$$\begin{cases} H_{t+1}(a) \doteq H_t(a) + \alpha \dfrac{\partial \mathbb{E}[R_t]}{\partial H_t(a)} \\ \mathbb{E}[R_t] = \sum_x \pi_t(x)\, q_*(x) \end{cases}$$

From this equations:

$$\frac{\partial \mathbb{E}[R_t]}{\partial H_t(a)} = \frac{\partial \left[\sum_x \pi_t(x)\, q_*(x)\right]}{\partial H_t(a)} = \sum_x q_*(x) \frac{\partial \pi_t(x)}{\partial H_t(a)}$$

We can include a baseline without changing the equality because the gradient sums to zero over all the actions, $\sum_x \frac{\partial \pi_t(x)}{\partial H_t(a)} = 0$. So:

$$\frac{\partial \mathbb{E}[R_t]}{\partial H_t(a)} = \sum_x (q_*(x) - B_t) \frac{\partial \pi_t(x)}{\partial H_t(a)}$$

* As $H_t(a)$ is changed, some actions probabilities go up and some go down, but the sum of the changes must be zero because the sum of the probabilities is always one.

$$\frac{\partial \mathbb{E}[R_t]}{\partial H_t(a)} = \sum_x \pi_t(x)(q_*(x) - B_t) \frac{\partial \pi_t(x)}{\partial H_t(a)} \cdot \frac{1}{\pi_t(x)}$$

$\mathbb{E}[R_t | A_t] = q_*(A_t)$

$$= \mathbb{E}\left[(q_*(A_t) - B_t) \frac{\partial \pi_t(x)}{\partial H_t(a)} \cdot \frac{1}{\pi_t(x)}\right]$$

$$= \mathbb{E}\left[(R_t - \bar{R}_t) \frac{\partial \pi_t(x)}{\partial H_t(a)} \cdot \frac{1}{\pi_t(x)}\right]$$

- Shortly we will establish that $\frac{\partial \pi_t(x)}{\partial H_t(a)} = \pi_t(x)(\mathbb{1}_{a=x} - \pi_t(a))$

where $\mathbb{1}_{a=x}$ is defined to be 1 if $a=x$, else 0.

Proof:
from the partial derivative of $\pi_t(x)$ math definition

Assuming that for now, we have

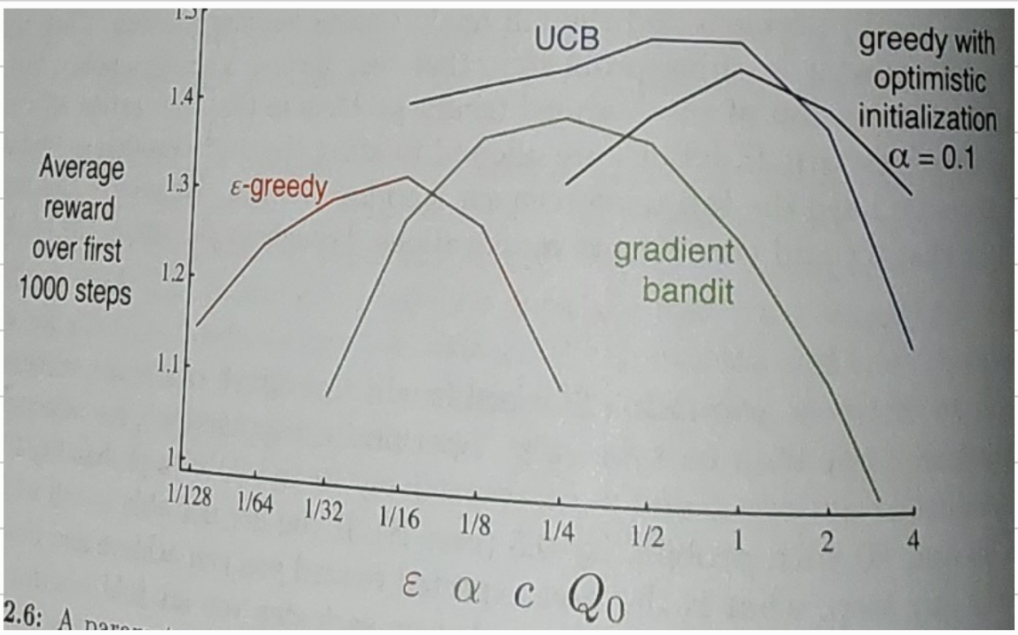
$$= \mathbb{E}\left[(R_t - \bar{R}_t)\, \pi_t(A_t) \left(\mathbb{1}_{a=A_t} - \pi_t(a) \right) / \pi_t(A_t) \right]$$

$$\frac{\partial \mathbb{E}[R_t]}{\partial H_t(a)} = \mathbb{E}\left[(R_t - \bar{R}_t) \cdot \left(\mathbb{1}_{a=A_t} - \pi_t(a) \right) \right]$$

$$\therefore \boxed{H_{t+1}(a) = H_t(a) + \alpha (R_t - \bar{R}_t) \cdot \left(\mathbb{1}_{a=A_t} - \pi_t(a) \right)}$$

↳ Incremental update rule of The Bandit Gradient Algorithm as stochastic Gradient Ascent

- Comparing all methods for k-armed bandits problem showed until here:



2.6: A para...