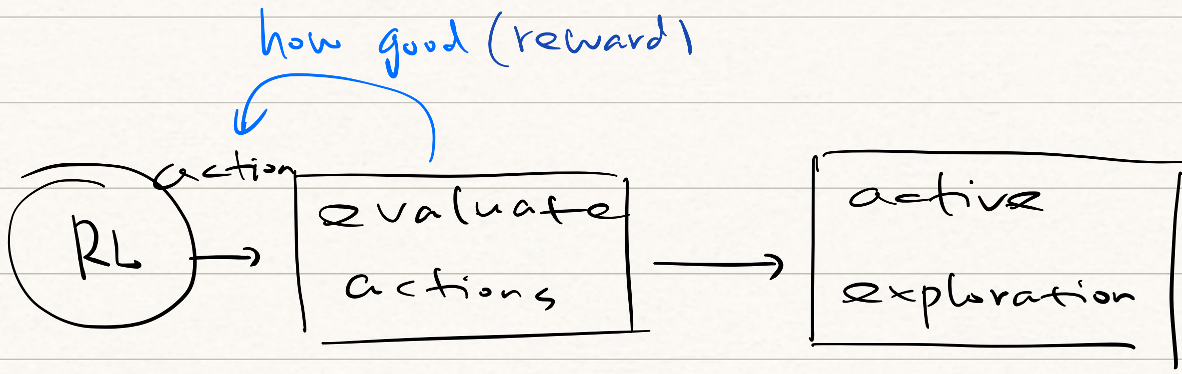
RL:

evaluative feedback: depends on action taken
(how good action)

Others:

instructive feedback: independent of the action taken (correct action)



k-armed bandit problem

k different options (actions)

action selections (time steps)

Value (of the action): expected/mean reward given that action

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

$$Q_t(a) \xrightarrow{\text{逼近}} q_*(a)$$

greedy actions

exploit current knowledge

Select one action : exploit current knowledge of the values of the actions

(on one step)

select non-greedy action : exploring

improve your estimate of the nongreedy action's value.

(in the long run)

Action-Value Method

Value :

$$\boxed{Q_t(a) \doteq \frac{\text{sum of rewards when } a \text{ taken prior to } t}{\text{number of times } a \text{ taken prior to } t} = \frac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i = a}}{\sum_{t=1}^{t-1} \mathbb{1}_{A_i = a}}}$$

$t \to \infty$, $\sum_{t=1}^{t-1} \mathbb{1}_{A_i = a} \to \infty$, $Q_t(a) \longrightarrow q_*(a)$

(sample-average method)

Greedy action selection method :

$$A_t \doteq \underset{a}{\operatorname{argmax}}\, Q_t(a)$$

exploit current knowledge to maximize immediate reward.

small probability ε; (ε-greedy)

Select randomly from among all the actions with $\boxed{\text{equal probability}}$, independently of the action-value estimates.

10-armed Testbed: $\varepsilon = 0.01$ slower but better

$\boxed{\text{"Get stuck performing suboptimal actions"}}$

Incremental Implementation

Action value:

$$Q_n = \frac{R_1 + R_2 + \cdots + R_{n-1}}{n-1} = \frac{1}{n} \sum_{i=1}^{n} R_i$$

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

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

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

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

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

New Estimate $\leftarrow$ Old Estimate + Step Size $\times$ [Target − Old Estimate]

error

Non-stationiary : (Exponential Recency - weighted Average)

$$Q_{n+1} = Q_n + \alpha \left(R_n - Q_n \right)$$

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

Conditions :

$$\begin{cases} \sum_{n=1}^{\infty} \alpha_n(a) = \infty \\ \sum_{n=1}^{\infty} \alpha_n^2(a) < \infty \end{cases}$$

Upper-confidence-bound Action Selection

(UCB)

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

$N_t(a)$: number of times that action a has been selected prior to time t

$c > 0$: control the degree of exploration

Gradient Bandit Algorithm

Estimate action values $\longleftrightarrow$ learning a numerical preference for each action a $H_t(a)$

$$\boxed{\text{select actions}}$$

soft-max distribution (Gibbs / Boltzmann distribution)

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

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

Associative Search

(Contextual Bandits)

Learn a policy : a mapping from situations to actions
that are best in those situations

trial-and-error learning to search for the best actions

association of these actions with the situations
in which they are best

parameter study :

perform best

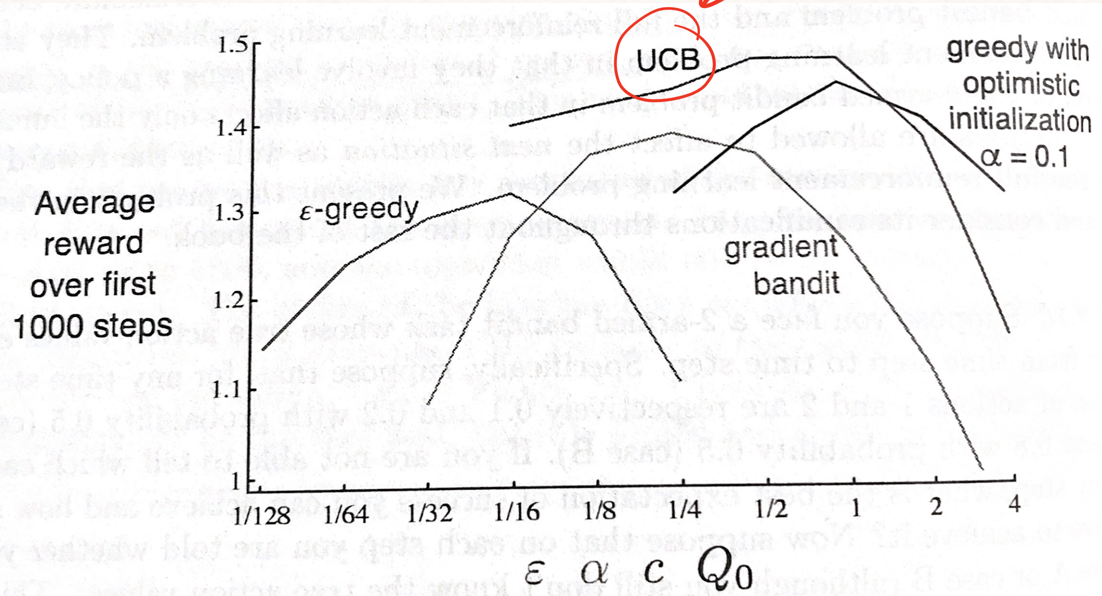


Figure 2.6: A parameter study of the various bandit algorithms presented in this chapt
Each point is the average reward obtained over 1000 steps with a particular algorithm a
particular setting of its parameter.