2.1 K-Armed Bandit Problem

A_t = action

R_t = reward

a = arbitrary action

Expected reward that a is selected:
$q_*(a) = E[R_t | A_t = a]$ ← real value

we want the $Q_t(a)$ = Estimated value of action to be close to $q_*(a)$

2.2 Action Value Methods

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

$$= \frac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i = a}}{\sum_{i=1}^{t-1} \mathbb{1}_{A_i = a}} \qquad \mathbb{1} = \begin{cases} 1 & \text{if predicate is true} \\ 0 & \text{if predicate is false} \end{cases}$$

As denominator goes to ∞, $Q_t(a)$ converges to $q_*(a)$

One way to choose an action is greedy action:

$$A_t = \underset{a}{\arg\max}\, Q_t(a)$$

1. Exploits current knowledge to maximize reward
2. Spends 0 time sampling other actions

ε-greedy method takes small probability ε chance to select other actions

- As action sampled ∞ times, $Q_t(a)$ converges to

2.6(a)

2.3 The 10-Armed Testbed

greedy better when variance is high.

Ex 2.2 $A = \{1, 2, 3, 4, 3\}$

$Q_2(A) = 0 \quad \forall a$

$A_1 = 1$	$A_2 = 2$	$A_3 = 2$	$A_4 = 2$	$A_5 = 3$
$R_1 = -1$	$R_2 = 1$	$R_3 = -2$	$R_4 = 2$	$R_5 = 0$

$Q_1(1) = \frac{-1}{1} = [0, 0, 0]$ $Q_4(2) = \frac{1}{2} = [-1, -0.5, 0]$

$Q_2(2) = \frac{1}{1} = [-1, 0, 0]$ $Q_5(3) = [-1, 0.33, 0]$

$Q_3(2) = \frac{1 + (-2)}{2} = [-1, 1, 0]$

Possible $t = 1, 2, 3$

Definitely $t = 4, 5$

Ex 2.3

$\varepsilon = 0.01$

$= (0.99 + 0.01 \cdot 0.1)$

$= 0.991$

$\varepsilon = 0.1$

$0.9 + 0.1 \cdot 0.1$

$= 0.91$

2.4 Incremental Implementation

R_i = reward after ith selection

Q_n = Estimate of action after selected $n-1$ times

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

An issue with this is that to recompute takes even longer. So we propose a new solution.

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

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

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

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

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

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

New Estimate ← Old + Step Size [Target − Old Estimate]

α

Simple Bandit Algorithm

Initialize for $a = 1$ to k

$Q(a) = 0$

$N(a) = 0$ # of times action a taken

Loop

$$A = \begin{cases} \text{argmax}_a Q(a) & p = 1 - \varepsilon \\ \text{random action} & p = \varepsilon \end{cases}$$

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

$N(A) = N(A) + 1$

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

2.5 Tracking a non-stationary problem

Previous solution only good for stationary bandit problems.

Solution to add weight to recent rewards than long past rewards.

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

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

This is called weighted average because sum of weights is 1.

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

also called the exponential recency-weighted average.

$\alpha_n(a)$ = step-size parameter to process reward after nth selection of action a

$\alpha_n(a) = \frac{1}{n}$ is sample-average method which converges by law of large numbers.

Ex 2.4

$$\prod_{i=1}^{n} (1-\alpha_i) \cdot Q_1 + \sum_{i=1}^{n} \left[\alpha_i \cdot \prod_{j=1}^{i} [1-\alpha_j] \cdot R_j \right]$$

2.6 Optimistic Initial Values

All methods depend on some $Q_1(a)$. This is bias. Sampling Average Methods: Bias disappears after all actions selected at least once.

Constant α: Bias decreases over time.

Instead of setting initial Q to 0, setting to 5 encourages exploration, good for starting problems but not good for many.

2.7 Upper-Confidence Bound Action Selection

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

c = confidence level

uncertainty of a's value