

10 runs testbed

A way to evaluate the effectiveness of the learning methods (greedy, ϵ -greedy...) comparing them numerically.

$y_{ox} \rightarrow R$ distribution

$x_{ox} \rightarrow$ Action

- Each R is sampled from a Normal distribution with same mean $q_*(a)$ and $VAR=1$
- Each $q_*(a)$ is drawn from a Normal distribution with mean=0 and variance=1
- The rewards are randomly sampled on $q_*(a)$
- Actions randomly taken on exploration steps

Experiments on 2000 steps

- Different values of ϵ
 - $\epsilon = 0$ (GREEDY)
 - $\epsilon = 0.01$
 - $\epsilon = 0.1$

GREEDY : performs worse in the long run
↳ it gets stuck in sub-optimal actions

$\epsilon = 0.1$: Reach best performances

OPTIMISTIC INITIAL VALUES METHOD

- Useful to encourage exploration
- We set $q_1(a) = +s \quad \forall a$

Optimism / Uncertainty

We can have **uncertainty** in the accuracy of our estimate.

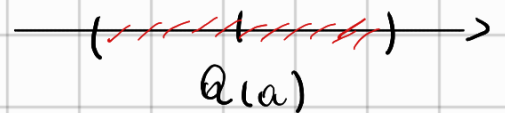
"Don't take the arm you believe is the best, take the one that has the most potential to be the best."

How to estimate uncertainty \rightarrow **confidence interval**

- SMALL \rightarrow very certain that $Q(a)$ is near our estimate.

region between ()

- BIG \rightarrow not certain



Upper Confidence Band

It's an action selection strategy that follows this principle:

"if you are uncertain about something, we should assume that it is good"

Pick the action with the highest UCB, because we have 2 possibilities:

1. We get a good reward (\checkmark SUM)
2. We learn more infos (\checkmark SUM to store)

Formally the UCB formula is:

parameter that control how much exploration

$$A_t = \operatorname{argmax}_a \left(\underbrace{Q_t(a) + c}_{\text{exploitation}} \sqrt{\underbrace{\frac{\ln t}{N_t(a)}}_{\text{exploration}}} \right)$$

- $Q_t(a)$ = current estimation of Q at time t

exploitation
↙

What I found so far

exploration
↓

How much I have to explore

- $N_t(a)$ = # time I choose a at time t

UCB exploit the fact that "the more I exploit in the beginning, the more the performances in long term"

The UCB obtains greater rewards than greedy, ϵ , ecc... but has difficulty to deal with NON-STATION problem.