Multi-Arm Bandits - Book

RL is different from other methods as it evaluates actions taken rather than instructs by giving correct actions. → this creates a need for active exploration, trial and error search for good behaviour

Evaluative feedback is the basis of methods for function optimisation.

[2.1] n-Armed Bandit Problem

↓

You are faced repeatedly with a choice among n different options.

↓

For each choice you receive a numerical reward chosen from a stationary probability distribution acc. to your action

- **Value**: Each action has an expected or mean reward for that action to be selected
- We don't know the values with certainty.
- If you maintain estimates then at any time step there is at least one action whose estimated value is greatest.

selecting this action is greedy approach also called Exploitation → Best to maximise reward at one step

selecting other actions is non-greedy also called exploration. → May produce a greater total reward.

Teacher's Signature.....

[2.2] Action Value Methods

$a \longrightarrow$ Action a
$q_*(a) \longrightarrow$ true (actual) value of action a
$Q_t(a) \longrightarrow$ estimated value of action a at t^{th} step

One way to estimate is by averaging the rewards actually received:

If by t^{th} time step action a has been chosen $N_t(a)$ times prior to t, yielding rewards $R_1, R_2 \cdots, R_{N_t(a)}$ then

$$Q_t(a) = \frac{R_1 + R_2 + \cdots + R_{N_t(a)}}{N_t(a)}$$

If $N_t(a) = 0$, then we define $Q_t(a)$ some default value, such as $Q_1(a) = 0$, As $N_t(a) \longrightarrow \infty$ by law of large numbers, $Q_t(a)$ converges to $q_*(a)$.

↓
sample-average method

The simplest action selection rule is to select the action with highest estimated action value i.e A_t^* for which

$$Q_t(A_t^*) = \max_a Q_t(a)$$
$$A_t^* = \text{argmax}_a Q_t(a)$$

But this again turns to exploitation. A simple solution is to select non-greedily once in a while, say with small probability ε.

Teacher's Signature.....

Instead of randomly selecting amongst all actions with equal probability → independent of the action value estimates. We calls methods using this near greedy action selection rule ε-greedy methods.

Advantage: in the limit as no. of plays increases, every action will be sampled an infinite number of times, guaranteeing $N_t(a) \to \infty \ \forall a \in A$. thus ensuring $Q_t(a)$ converges to $q_*(a) \ \forall a \in A$.

Probability of selecting optimal action converges to $1-\varepsilon$.

[2.3] Incremental Implementation

In the previous we need to store all the rewards received for recalculation which is inefficient and redundant.

we say $Q_k \to$ avg. of first $k-1$ rewards
$R_k \to$ reward for k^{th} step.

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

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

$$= \frac{1}{k} \left(R_k + (k-1) Q_k + Q_k - Q_k \right)$$

$$= \frac{1}{k} \left(R_k + k Q_k - Q_k \right)$$

$$\boxed{Q_{k+1} = Q_k + \frac{1}{k} \left[R_k - Q_k \right]}$$

Teacher's Signature.....

Himanshu

Page No.

Date

New Estimate ← Old estimate + Stepsize [Target − oldestimate]

Stepsize can be denoted by α → it is a parameter

[Target − oldestimate] → error in the estimate

for sample-average method

$$\alpha = \frac{1}{k}$$

[2.4] Tracking Non-stationary Problem

The averaging methods discussed until now works for stationary environment.

In this case we must weigh the recent rewards more heavily. One way → using a constant step-size parameter.

$$Q_{k+1} = Q_k + \alpha[R_k - Q_k]$$

where $\alpha \in (0, 1]$

this makes

$$Q_k = \alpha R_k + (1-\alpha) Q_k$$

$$= \alpha R_k + (1-\alpha)[\alpha R_{k-1} + (1-\alpha) Q_{k-1}]$$

$$= \alpha R_k + \alpha(1-\alpha) R_{k-1} + (1-\alpha)^2 Q_{k-1}$$

$$= \alpha R_k + (1-\alpha) R_{k-1}\alpha + (1-\alpha)^2 R_{k-2}\alpha + \dots + (1-\alpha)^{k-1}\alpha R_{k-1} + (1-\alpha)^k Q_1$$

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

↓

sometimes called exponential, recency-weighted average

Teacher's Signature .....

Sometimes its convenient to vary the step-size parameter from step to step.

$\alpha_k(a)$ → Step-size parameter to process the reward received after k^{th} selection of a

If $\alpha_k(a) = \frac{1}{k}$ → this becomes sample avg. method.
↳ guaranteed to converge

conditions for assured convergence

$$\sum_1^\infty \alpha_k(a) = \infty \quad \& \quad \sum_1^\infty \alpha^2_k(a) < \infty$$

guarantees steps are large enough to eventually overcome initial conditions or random fluctuations

eventually steps will become small enough to assure convergence.

Noting this we see a constant $\alpha_k(a) = \alpha$ doesn't meet the second condition and hence will not converge, thus will keep tracking the recent reward.

[2.5] Optimistic Initial Values

All methods discussed yet are somehow dependant on initial value estimates i.e $Q_1(a)$ i.e these methods are biased by initial estimates. For sample average method bias disappears when all actions are visited at least once but for constant α → the bias is permanent though decreasing over time.

Initial action-values encourage exploration as they have to try all methods to find the best one.

Initially optimistic method performs worse, but eventually its performs better as exploration decreases with time. → This called optimistic initial values

Good for stationary problems but not well suited for non stationary.

[2.6] Upper-Confidence-Bound Action selection

ε-greedy selection forces non-greedy actions to be tried but indiscriminately with no preference to those who are nearly-greedy actions according to their potential for being actually optimal. One way of doing that is

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

$c > 0$ controls the degree of exploration

If $N_t(a) = 0$, then a is considered to be a maximising action.

The square root term is a measure of uncertainty or variance in the estimate of a's value. The quantity being max'ed over is thus a sort of upper bound of the true value of action a.

Teacher's Signature.....

[2.7] Gradient Bandits

Not in terms of reward. ← $H_t(a)$ → Numerical preference for each action a.

↓

larger the preference more often the action is taken

It is relative and is determined using soft-max distribution.

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

↓

Probability of taking action a at time t.

Initially all preferences are same: For eg: $H_1(a) = 0 \quad \forall a$

Natural learning algorithm based on stochastic gradient ascent.

A_t → action taken on time step t

R_t → reward received.

$$H_{t+1} = H_t(A_t) + \alpha(R_t - \bar{R}_t)(1 - \pi_t(A_t))$$

and $$H_{t+1}(a) = H_t(a) - \alpha(R_t - \bar{R}_t)\pi_t(a) \quad \forall a \neq A_t$$

$\bar{R}_t \in R$ → avg. of all rewards upto and including time t.

↓

serves as a baseline. If reward is higher → $P(A_t)$ ↑ else $P(A_t)$ ↓

Teacher's Signature.....

$$H_{t+1}(a) = H_t(a) + \alpha \frac{\partial E[R_t]}{\partial H_t(a)}$$

$$E[R_t] = \sum_b \pi_t(b)\, q_t(b)$$

$$\frac{\partial E[R_t]}{\partial H_t(a)} = \frac{\partial}{\partial H_t(a)} \left[\sum_b \pi_t(b)\, q_t(b) \right]$$

$$= \sum_b q_t(b) \frac{\partial \pi_t(b)}{\partial H_t(a)}$$

$$= \sum_b (q_t(b) - X_t) \frac{\partial \pi_t(b)}{\partial H_t(a)}$$

any scalar that doesn't depend on b

Σ of this must be 0 as the sum of all probabilities will always remain 0.

$$= \sum_b \pi_t(b) (q_t(b) - X_t) \frac{\partial \pi_t(b)}{\partial H_t(a)} \Big/ \pi_t(b)$$

this can be written as —

$$= E\left[(q_t(A_t) - X_t) \frac{\partial \pi_t(A_t)}{\partial H_t(a)} \Big/ \pi_t(A_t) \right]$$

$$= E\left[(R_t - \bar{R}_t) \frac{\partial \pi_t(A_t)}{\partial H_t(a)} \Big/ \pi_t(A_t) \right]$$

Teacher's Signature.....

Himanshu

Page No.

Date

Shortly we establish that $\frac{\partial \pi_t(b)}{\partial H_t(a)} = \pi_t(b)\left(I_{a=b} - \pi_t(a)\right)$

Assuming this to be true

$$= E\left[(R_t - \bar{R}_t)\pi_t(A_t)\left(I_{a=A_t} - \pi(A_t)\right) \middle| \pi_t(A_t)\right]$$

$$= E\left[(R_t - \bar{R}_t)\left(I_{a=A_t} - \pi(A_t)\right)\right]$$

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

Proving $\frac{\partial \pi_t(b)}{\partial H_t(a)} = \pi_t(b)\left(I_{a=b} - \pi_t(a)\right)$

$$\frac{\partial\left[\frac{f(x)}{g(x)}\right]}{\partial x} = \frac{\frac{\partial f(x)}{\partial x} g(x) - \frac{\partial (g(x))}{\partial x} f(x)}{g(x)^2}$$

$$\frac{\partial \pi_t(b)}{\partial H_t(a)} = \frac{\partial\left[\frac{e^{H_t(b)}}{\sum_1^n e^{H_t(c)}}\right]}{\partial H_t(a)}$$

$$= \frac{\frac{\partial H_t(b)}{\partial H_t(a)}\left[\sum_1^n e^{H_t(c)}\right] - e^{H_t(b)} \frac{\partial \sum_1^n e^{H_t(c)}}{\partial H_t(a)}}{\left(\sum_1^n e^{H_t(c)}\right)^2}$$

$$= \frac{I_{a=b} e^{H_t(a)} \sum_1^n e^{H_t(c)} - e^{H_t(b)} e^{H_t(a)}}{\left(\sum_1^n e^{H_t(c)}\right)^2}$$

$$= \frac{I_{a=b} e^{H_t(b)}}{\sum_1^n e^{H_t(c)}} - \frac{e^{H_t(b)} e^{H_t(a)}}{\left(\sum_1^n e^{H_t(c)}\right)^2}$$

Teacher's Signature.....

$$= I_{a=b}\, \pi_t(b) - \pi_t(b)\,\pi_t(a)$$

$$= \pi_t(b)\left[I_{a=b} - \pi_t(a)\right]$$