Page :
Date :

RL

★ Exploration & Exploitation :-

Example :- Tic-Tac-Toe.

X		O
	X	X
		O

Player 1 & 2
↓ ↓
X O

Assume :-
Player 2 plays as per some strategy.

Player 1 is the RL agent.

★ State :- $S = \{ S_1, S_2 ---- S_9, S_{10} \}$

$S_i \in \{X, O, e\}$, i = 1, 2, ... 9

$S_{10} \in \{X, O\}$

$A_i \triangleq$ change an empty slot by putting X or O

$S_t = (X, e, O, e, X, X, e, e, O, X)$

$S_{t+1} = (X, e, O, O, X, X, e, e, O, X)$

Reward :- $R_i = \begin{cases} +1 & \text{if win} \\ 0 & \text{if draw} \\ -1 & \text{if lose.} \end{cases}$

Reward :- is 0 in between (i.e. before the game ends).

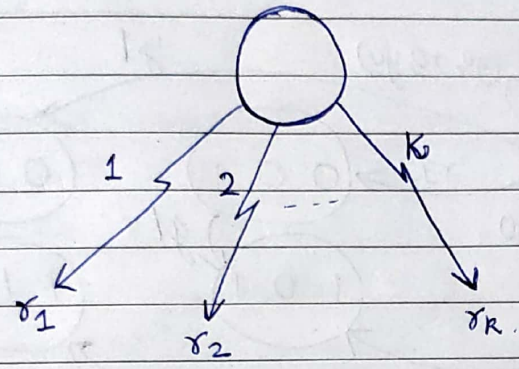
Chapter-1 → Sutton & Barto.

Page :
Date :

Chapter-2

Multi-armed Bandits

No. of states = 1.
Multiple actions (K).



$r_1, r_2 \dots r_K$: rewards obtained.
by playing arm-1, arm2, ... armk.

Assumption :- Rewards of each arm are random & come as per a certain distribution that is a function of the arm being pulled.

Problem :- Find the best arm given that we don't know the reward distribution for any arm.

No. of states = 1.
No. of actions = K

Let
$q^*(a) = E[R_{t+1} \mid A_t = a]$

$\Downarrow$

expected reward obtained by pulling arm a.

Note :- We don't know a priori what $q^*(a)$ is $\forall a = 1 \dots K$

We shall estimate $q^*(a)$ using sample average rewards

$Q_t(a)$ = Sum of rewards obtained by pulling arm a untill time t

No. of time arm a is pulled untill t.

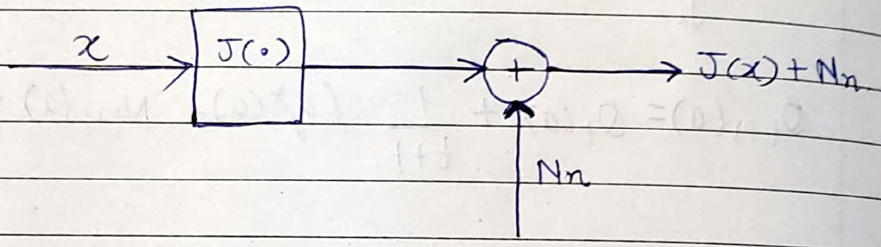
Camlin

New estimate = old estimate + Step-size (or learning rate) (target − old estimate)

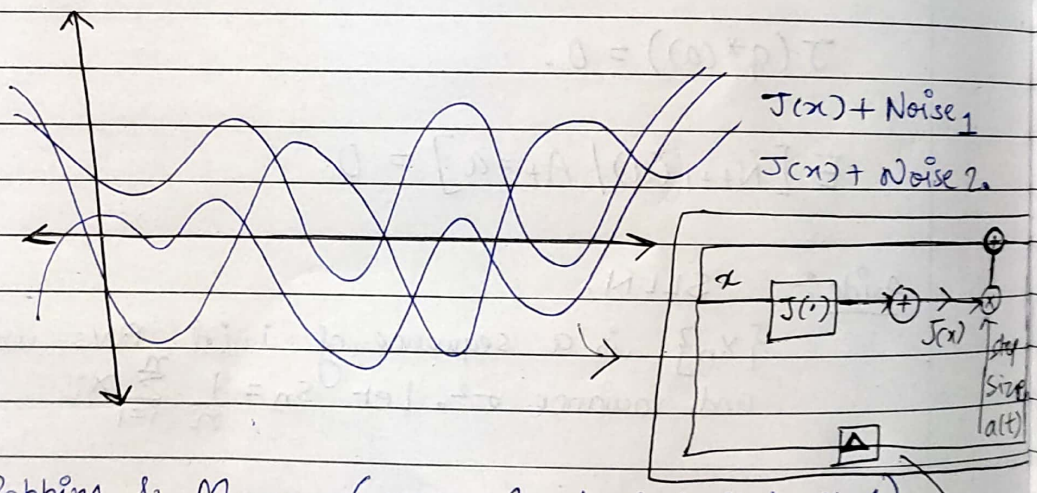
☆ Incremental update algos

★ Stochastic Approximation algorithms:-

Suppose $J: \mathbb{R}^d \rightarrow \mathbb{R}$ is some objective function, whose analytical form we don't know. We can obtain noisy estimates of this function.



Q) Find x^* s.t. $J(x^*) = 0$



Robbins & Munro (1951 - Annals of Applied Stat).

Algo closes the loop

Under some assumptions, it can be shown that:-

$x_n \rightarrow x^*$ w.p. 1 s.t. $J(x^*) = 0$

$x_{n+1} = x_n + a(n) [J(x) + N]$

Noise → zero mean $E[N_{n+1}] = 0$

Page :
Date :

Goal :- Find $A^* = \arg\max_a q^*(a)$.

$$Q_t(a) = \frac{\sum_{i=1}^{t} R_i \cdot I_{\{A_i=a\}}}{\sum_{i=1}^{t} I_{\{A_i=a\}}} \qquad I_{\{A_i=a\}} = \begin{cases} 1 & \text{if } A_i=a \\ 0 & \text{o.w.} \end{cases}$$

Policies :-

(1) $A_{t+1} = \arg\max_a Q_t(a)$

(Greedy Policy) Exploitation

A good policy should incorporate some exploration

(2) ϵ - greedy exploration

Here, $0 < \epsilon < 1$ is a small parameter.

$$A_{t+1} = \begin{cases} \arg\max_a Q_t(a) & ; \text{ w.p. } (1-\epsilon). \\ \text{random action} & ; \text{ w.p. } \epsilon \end{cases}$$

(3) UCB (upper confidence bound) exploration :-

Suppose we pull a certain arm a all the time.

$$Q_t(a) = \frac{1}{t}\sum_{i=1}^{t} R_i.$$

$$Q_{t+1}(a) = \frac{1}{(t+1)}\sum_{i=1}^{t+1} R_i = \frac{1}{(t+1)}\left(\sum_{i=1}^{t} R_i + R_{t+1}\right) = \frac{t}{(t+1)}\left[\frac{1}{t}\sum_{i=1}^{t} R_i\right] + \frac{R_{t+1}}{t+1}$$

$$= \frac{t}{t+1} Q_t(a) + \frac{1}{t+1} R_{t+1}$$

Stochastic approx. algo.

$$\Rightarrow \boxed{Q_{t+1}(a) = Q_t(a) + \frac{1}{t+1}\left(R_{t+1} - Q_t(a)\right)}$$

Camlin

Page :
Date :

Recall the bandit algorithm

$$Q_{t+1}(a) = Q_t(a) + \frac{1}{t+1}\left(R_{t+1} - Q_t(a)\right)$$

$$Q_{t+1}(a) = Q_t(a) + \frac{1}{t+1}\left(E[R_{t+1} \mid A_t = a] + N_{t+1}(a) - Q_t(a)\right)$$

where $N_{t+1}(a) = \left(R_{t+1} - E[R_{t+1} \mid A_t = a]\right)$

$$Q_{t+1}(a) = Q_t(a) + \frac{1}{t+1}\left(q^*(a) + N_{t+1}(a) - Q_t(a)\right)$$

$J(Q_t(a)) = \left(q^*(a) - Q_t(a)\right)$ is not known.

$J(q^*(a)) = 0.$

$E[N_{t+1}(a) \mid A_t = a] = 0.$

Aside :- SLLN.

$\{X_n\}$ is a sequence of i.i.d RVs with mean μ and variance σ^2. Let $S_n = \frac{1}{n}\sum_{i=1}^{n} X_i$.

$$S_n \to \mu \quad a.s.$$

$$S_{n+1} = S_n + \frac{1}{n+1}\left(X_{n+1} - S_n\right) \Rightarrow S_n \to \mu \ a.s.$$

using Stoc. approx.

Camlin

Page :
Date :

Step size or learning parameter

$$a(t) = \frac{1}{t+1}, \quad t \geq 0.$$

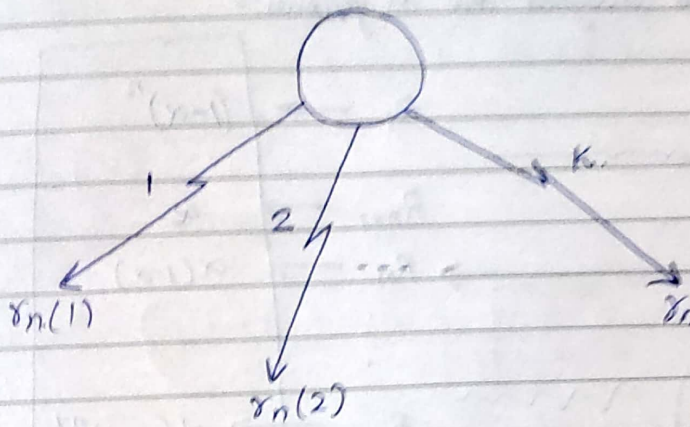
In general, conditions on step-sizes :-

(i) $a(t) > 0 \quad \forall t.$

(ii) $\sum_t a(t) = \infty \rightarrow$ Ensures progress

Page No.
Date / /

Multi armed bandits :



Assume $r_n(1), \ldots, r_n(K)$ are independent $\forall n \; \forall i = 1, \ldots K$

So, far we assumed reward distribution does not change with time, i.e a stationary setting

- what happens if the dist. is changing with time, i.e. non-stationary setting.
- Const. step sizes help instead of diminishing step sizes.

$Q_{n+1}(i) = Q_n(i) + \alpha(R_{n+1}(i) - Q_n(i))$. Here α does not change with time.

Consider the recursion :-

$$Q_{n+1}(i) = Q_n(i) + \alpha(R_{n+1} - Q_n)$$

$$= \alpha R_{n+1} + (1-\alpha) Q_n$$

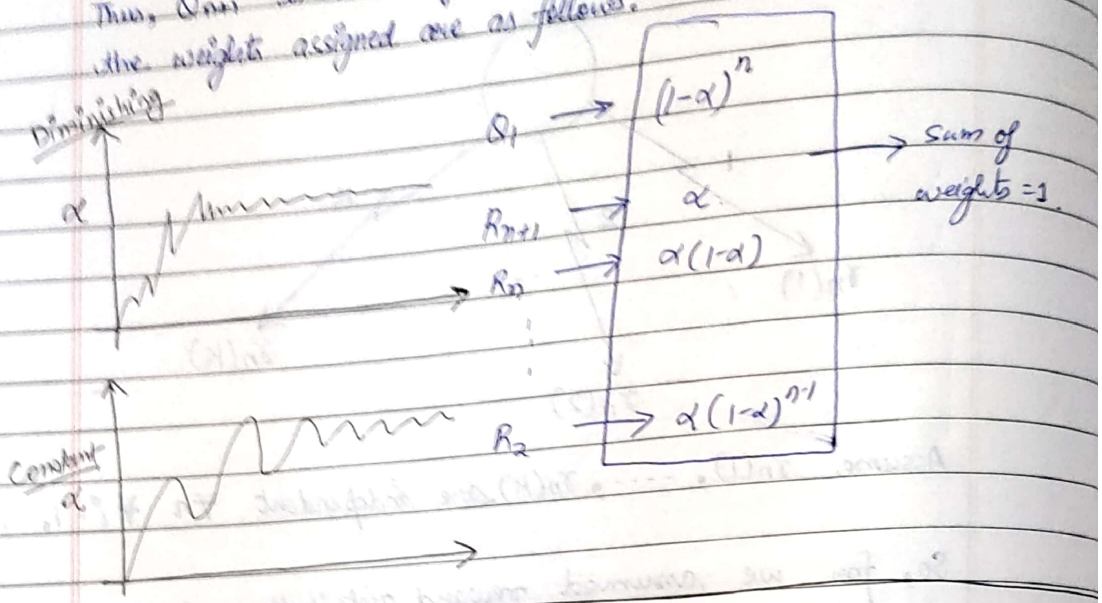
$$= \alpha R_{n+1} + (1-\alpha)(\alpha R_n + (1-\alpha) Q_{n-1})$$

$$= \alpha R_{n+1} + \alpha(1-\alpha) R_n + (1-\alpha)^2 Q_{n-1}$$

$$\vdots$$

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

Thus, Q_{n+1} is a weighted av. of $R_1, R_2, \ldots, R_{n+1}$, where the weights assigned are as follows:-



3. Exploration Strategy :- Upper Confidence Bound (UCB) :-

Let $R_1, R_2 \ldots R_n \ldots$ are independent rewards and let them be sub-gaussian.

Thus, $E[R_i] = 0 \ \forall i$, Suppose $Q_n(a) = \frac{1}{n}\sum_{i=1}^{n} R_i$, $n \geq 1$. [what is $Q_n(a)$?]

$$P(|Q_n(a)| \geq \varepsilon) \leq \underbrace{e^{-n\varepsilon^2/2}}_{=\delta}$$

$$\frac{-n\varepsilon^2}{2} = \log\delta \Rightarrow \varepsilon^2 = \frac{2}{n}\log\left(\frac{1}{\delta}\right)$$

$$\Rightarrow \varepsilon = \sqrt{\frac{2}{n}\log\left(\frac{1}{\delta}\right)}$$

Suppose $\delta = \frac{1}{n}$ (Why was this asm taken?)

$$\Rightarrow P(Q_n(a) \geq \varepsilon) \leq \delta.$$

Best plausible estimate for the maximizing asm :-

$$A_n = \underset{a}{\arg\max}\left(Q_n(a) + c\sqrt{\frac{\log n}{N_n(a)}}\right) \quad n \geq k$$

Page No.
Date / /20

Until $N_n(a) = K$, we pick each action one at a time.
(i.e., let $N_n(a) = n$, for $n < k$).

(★) Gradient Bandit Algorithms :-

Suppose $P(A_t = a) = \frac{e^{H_t(a)}}{\sum_{b=1}^{K} e^{H_t(b)}} \triangleq \pi_t(a)$.

(Softmax or Boltzmann distribution)

$H_t(a)$ = Preferred function.

Consider the gradient algorithm :-

$$H_{t+1}(a) = H_t(a) + \alpha \frac{\partial E[R_t]}{\partial H_t(a)} ; \quad q_*(a) = E[R_t | A_t = a]$$

Here, $E[R_t] = \sum_x \pi_t(x)\, q_*(x)$

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

[From where did this come].

$$= \sum_x \left(q_*(x) - B_t \right) \frac{\partial \pi_t(x)}{\partial H_t(a)}$$

Where B_t = Baseline function (does not depend on x)

Note :- Extra-term :- $\sum_x B_t \frac{\partial \pi_t(x)}{\partial H_t(a)}$

$$= B_t \sum_x \frac{\partial \pi_t(x)}{\partial H_t(a)}$$

$$= B_t \frac{\partial}{\partial H_t(a)} \left(\sum_x \pi_t(x) \right)$$

$$= 0 \qquad (\because \sum_x \pi_t(x) = 1)$$

Remark:- Even though both estimators are unbiased, a proper choice of baseline can reduce the estimator variance.

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

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

where $A_t \sim \pi_t(\cdot)$

Suppose we use $B_t = \bar{R}_t$ (av. of rewards by time t).

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

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

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

$$\Rightarrow \frac{\sum_{b=1}^{k} e^{H_t(b)} e^{H_t(A_t)} \mathbb{I}_{\{A_t=a\}} - e^{H_t(A_t)} \cdot e^{H_t(a)}}{\left(\sum_{b=1}^{k} e^{H_t(b)}\right)^2}$$

$$= \frac{e^{H_t(A_t)}}{\sum_{b=1}^{k} e^{H_t(b)}}\mathbb{I}_{\{A_t=a\}} - \frac{e^{H_t(A_t)}}{\sum_{b=1}^{k} e^{H_t(b)}} \cdot \frac{e^{H_t(a)}}{\sum_{b=1}^{k} e^{H_t(b)}}.$$

$\frac{\partial E[R_t]}{\partial H_t(a)}$

(★) Algorithm

H_{t+1}

For $a = A$

H

For $a \neq$

t

$$= \pi_t(A_t)\left(I_{\{A_t=a\}} - \pi_t(a)\right).$$

Revise → Gradient

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

Note: We dropped expectation. (bcoz we don't know).

(★) Algorithm :-

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

For $a = A_t$.

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

For $a \neq A_t$

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

★ Read Ch-1 & Ch-2 Sutton & Barto.

(★) Quiz on Jan 24.