Introduction to reinforcement learning

Urtzi Ayesta, Matthieu Jonckheere

Chapter III : Bandits

Model

Model

- In each time slot, a player choose to play one of K bandits (machines).
- She observes the reward. (No other observable state).
- We assume that each bandit produces i.i.d. rewards with fixed **unknown** distribution.
- We are dealing with a very simple MDP, with $\mathcal{S} = \{1, \ldots, K\}$ and every possible transition from i to j is allowed.
- We can actually in that case associate actions with states (i.e. bandits).

Exploration vs exploitation

We aim at estimating

$$\mu_a = E(R_t|a),$$

without loosing too much reward in the process.

Simple estimation of μ_a:

$$\hat{\mu}_a(t) = \frac{\sum_{s=1}^{t} R_s 1_{A_s=a}}{\sum_{s=1}^{t} 1_{A_s=a}}.$$

If $\sum_{s=1}^{t} 1_{A_s=a}$ diverges, LLN (applied to $R_i 1_{A_i=a}$) implies that

$$\hat{\mu}_a(t) \to \mu_a$$

Sequential updates

Note that

$$\hat{\mu}_a(t) = \hat{\mu}_a(t-1) + \frac{1}{t(a)}\big(r_t(a) - \hat{\mu}_a(t-1)\big).$$

which has the shape:

$$\mathit{NewEstimate} \leftarrow \mathit{OldEstimate} + \mathit{StepSize}.(\mathit{Target} - \mathit{OldEstimate})$$

Target = innovation = new data of interest

Sequential updates

More general proposal (step n):

$$\mathit{NewEstimate} \leftarrow \mathit{OldEstimate} + \alpha(n)(\mathit{Target} - \mathit{OldEstimate})$$

Target = innovation = new data of interest

Exploration vs exploitation

Taking this as a generic estimation scheme, what conditions do we need to impose on the step size?

- Case $\alpha(n) = 1/n$: LLN
- Case α constant,

Exploration vs exploitation

Taking this as a generic estimation scheme, what conditions do we need to impose on the step size?

- Case $\alpha(n) = 1/n$: LLN
- Case α constant,
  then it does not converge to some deterministic limit.
- General case: Theorem of stochastic approximation

Stochastic approximation theorem

Theorem

CNS of convergence:

- *Convergence:*

$$\sum_n \alpha_n = \infty.$$

- *Make the noise small:*

$$\sum_n \alpha_n^2 < \infty.$$

What policy would you propose?

A simple proposal: ϵ-greedy

Naive exploration/exploitation tradeoff:

- With probability ϵ, choose an arm at random,
- With probability $1 - \epsilon$, choose

$$A_t = \arg\max \hat{\mu}_t(a)$$

Performance measure

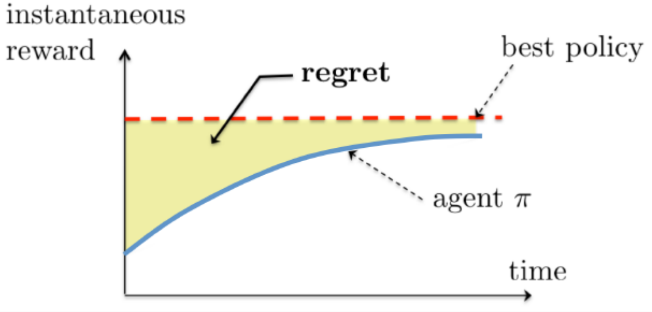
"Cumulative regret":

$$Regret_t = t\mu^* - \sum_{i=1}^{t} R_i$$

where μ^* is the best mean:

$$\mu^* = \max \mu_k.$$

Regret



Objective is to minimize regret.

Regret

- The action-value is the mean reward for action a

$$Q(a) = \mathbb{E}(r|a)$$

- The optimal value V^* is

$$q^* = Q(a^*) = \max_a Q(a)$$

- The regret is the opportunity loss per step $q^* - Q(a_t)$

Regret

- The action-value is the mean reward for action a

$$Q(a) = \mathbb{E}(r|a)$$

- The optimal value V^* is

$$q^* = Q(a^*) = \max_a Q(a)$$

- The regret is the opportunity loss per step $q^* - Q(a_t)$
- The total regret is the total opportunity loss

$$L_T = \mathbb{E}\left(\sum_{t=1}^{T} q^* - Q(a_t)\right)$$

Regret

- The action-value is the mean reward for action a

$$Q(a) = \mathbb{E}(r|a)$$

- The optimal value V^* is

$$q^* = Q(a^*) = \max_a Q(a)$$

- The regret is the opportunity loss per step $q^* - Q(a_t)$
- The total regret is the total opportunity loss

$$L_T = \mathbb{E}\left(\sum_{t=1}^{T} q^* - Q(a_t)\right)$$

Counting Regret

- Let $N_t(a)$ be the number of times a has been selected
- The *gap* be $\Delta_a = V^* - Q(a)$

Counting Regret

- Let $N_t(a)$ be the number of times a has been selected
- The *gap* be $\Delta_a = V^* - Q(a)$

$$\begin{aligned} L &= \mathbb{E}\left(\sum_{t=1}^{T} q^* - Q(a_t)\right) \\ &= \sum_a \mathbb{E}(N_T(a))(q^* - Q(a)) \\ &= \sum_a \mathbb{E}(N_T(a))\Delta_a \end{aligned}$$

$$\sum_a \mathbb{E}(N_T(a)) = T$$

Counting Regret

- Let $N_t(a)$ be the number of times a has been selected
- The *gap* be $\Delta_a = V^* - Q(a)$

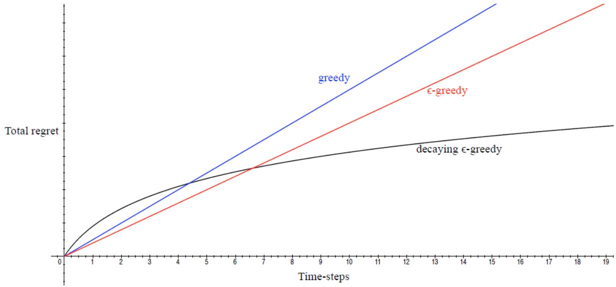
$$\begin{aligned} L &= \mathbb{E}\left(\sum_{t=1}^{T} q^* - Q(a_t)\right) \\ &= \sum_a \mathbb{E}(N_T(a))(q^* - Q(a)) \\ &= \sum_a \mathbb{E}(N_T(a))\Delta_a \end{aligned}$$

$$\sum_a \mathbb{E}(N_T(a)) = T$$

A good algorithm ensures small *counts* for large *gaps*

Problem: Gaps are not known!

Linear or sublinear regret



If an algorithm forever explores it will have linear total regret
If an algorithm never explores it will have linear total regret
Greedy policy, and ϵ-greedy have linear regret
Is it possible to achieve sublinear total regret?

Greedy Algorithm

- Estimate the reward $\hat{Q}_t(a) \approx Q(a)$
- Estimate the value of each action by MC evaluation

$$\hat{Q}_t(a) = \frac{1}{N_t(a)} \sum_{t=1}^{T} r_t 1_{a_t = a}$$

Greedy Algorithm

- Estimate the reward $\hat{Q}_t(a) \approx Q(a)$
- Estimate the value of each action by MC evaluation

$$\hat{Q}_t(a) = \frac{1}{N_t(a)} \sum_{t=1}^{T} r_t 1_{a_t=a}$$

- The *greedy* algorithm selects action with highest value

$$a_t^* = \text{argmax}_a \hat{Q}_t(a)$$

- Greedy can lock onto a suboptimal action forever

$\Longrightarrow$ Greedy has linear total regret

$\epsilon -$ *Greedy* Algorithm

- With probability $1 - \epsilon$ select $a = \text{argmax}_a \hat{Q}(a)$
- With probability ϵ select a random action

$\epsilon - \textit{Greedy}$ Algorithm

- With probability $1 - \epsilon$ select $a = \text{argmax}_a \hat{Q}(a)$
- With probability ϵ select a random action
- $\epsilon - \textit{Greedy}$ algorithm continues to explore forever
- Define the *gaps* $\Delta_a = V^* - Q(a)$
- The regret per step is

$$l_t \geq \frac{\epsilon}{\mathcal{A}} \sum_a \Delta_a$$

- $\implies \epsilon - \textit{Greedy}$ has linear total regret

Optimistic Initialization

- Simple and practical: Initialize $Q(a)$ to high value
- Update by incremental MC evaluation

$$\hat{Q}_t(a_t) = \hat{Q}_t(a_{t-1}) + \frac{1}{N_t(a_t)}(r_t - \hat{Q}_{t-1})$$

- Encourages systematic exploration
- But can lock onto suboptimal action

$\Longrightarrow$ $\epsilon - \mathit{Greedy} + \mathit{optimistic} \quad \mathit{exploration}$ has linear total regret

Decaying $\epsilon_t - \textit{Greedy}$ algorithm

- Pick a decaying schedule for $\epsilon_1, \epsilon_2, \ldots,$

Decaying $\epsilon_t - \textit{Greedy}$ algorithm

- Pick a decaying schedule for $\epsilon_1, \epsilon_2, \ldots,$
- Consider the following schedule:

$$c > 0, \quad \delta = \min_{a|\Delta_a>0} \Delta_i$$

$$\epsilon_t = \min\left\{1, \frac{c|\mathcal{A}|}{\delta^2 t}\right\}$$

Decaying $\epsilon_t - \textit{Greedy}$ algorithm

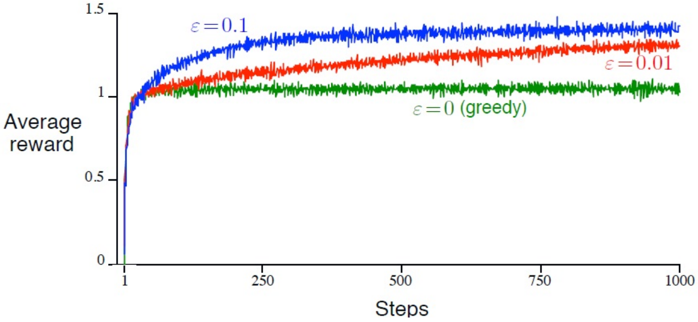
- Pick a decaying schedule for $\epsilon_1, \epsilon_2, \dots,$
- Consider the following schedule:

$$c > 0, \quad \delta = \min_{a|\Delta_a > 0} \Delta_i$$

$$\epsilon_t = \min \left\{ 1, \frac{c|\mathcal{A}|}{\delta^2 t} \right\}$$

- Decaying $\epsilon_t - \textit{greedy}$ has logarithmic asymptotic regret
- It requires advance knowledge of gaps
- Find an algorithm with sublinear regret without knowledge

Some performance simulations



Optimistic start

Put arbitrary large values for the initial estimates to force to explore more at the beginning.

Upper confidence interval: UCB

Can we do better by measuring the quality of the estimation as time goes by?

Proposal:
we use an upper bound on a confidence interval for $\hat{\mu}_a$ instead of looking at $\hat{\mu}_a$

Intermezzo: Large deviations for random walks

(X_i) iid with mean μ and finite exponential moment.

Lemma (Chernoff-Hoeffding)

$$P\Big(\sum_{i=1}^{n} X_i - \mu n \geq \epsilon n\Big) \leq e^{-2\epsilon^2 n}.$$

$$P\Big(\sum_{i=1}^{n} X_i - \mu n \leq -\epsilon n\Big) \leq e^{-2\epsilon^2 n}.$$

Classical proof of such results: exponential Markov inequality and optimisation of parameters

Upper confidence interval

Proposal: we use an upper bound on a confidence interval for $\hat{\mu}_a$ instead of looking at $\hat{\mu}_a$
Recall

$$N_t(a) = \sum_{s=1}^{t} 1_{A_s = a}$$

Taking $\epsilon_{a,t} = \sqrt{\frac{2\log(t)}{N_t(a)}}$, and using C-H, we obtain:

$$P\Big(\hat{\mu}_t(a) + \epsilon_{a,t} \leq \mu_a\Big) \leq O(t^{-4}).$$

Then we define the policy:

$$A_t = \arg\max\Big[\hat{q}_t(a) + \epsilon_{a,t}\Big].$$

UCB paradigm

- "optimism against uncertainty": if you don't know which action is best then choose the one that currently looks to be the best.
- $\epsilon_{a,t}$ quantifies the current uncertainty on a given action. Choose uncertain actions is good for exploration.

Theoretical upper performance bound for UCB

We control the error of sub-estimation. This corresponds to an optimistic policy.
The following performance bound can be proven:

Proposition

Under UCB

$$E(Regret_t) \leq c_1 + c_2 \log(t).$$

Sketch of the proof I

We observe that:

$$Regret_t = \sum_k (\mu^* - \mu_k) N_k(t) = \sum_k \Delta_k N_k(t),$$

We define the event:

$$B_{k,t-1} = \{\omega : \mu_k - \epsilon_{t,k} \leq \hat{\mu}_k(t) \leq \mu_k + \epsilon_{t,k}\}.$$

Let k^* the optimal bandit If action k is chosen then, in the event $B_{k,t-1} \cap B_{k^*,t-1}$:

$$\hat{\mu}_k(t) + \epsilon_{t,k} \geq \hat{\mu}_{k^*}(t) + \epsilon_{t,k^*} \geq \mu^*.$$

(First inequality because of the definition of the policy, second because of the UCB)
Then

$$\epsilon_{t,k} \geq \mu^* - \hat{\mu}_k(t) \geq \Delta_k - \epsilon_{t,k}.$$

and

$$2\epsilon_{t,k} \geq \Delta_k.$$

Sketch of the proof II

As $\epsilon_{t,k} = \sqrt{\frac{2\log(t)}{N_k(t-1)}}$, then

$$N_k(t-1) \leq \frac{8\log(t)}{\Delta_k^2}.$$

From there we deduce

$$Regret_t 1_{\cap_{s\leq t} B_{k,s-1}} = \sum_k (\mu_k^* - \mu_k) N_k(t) \leq \log(t) 8 \sum_k \frac{1}{\Delta_k},$$

From the other side, using large deviations:

$$E\Big(N_k(t-1) 1_{\left(\cup_{s\leq t} B_{k,s-1}\right)^c}\Big) \leq c_1 \sum_s s^{-4} \leq c_2.$$

Generic lower bound

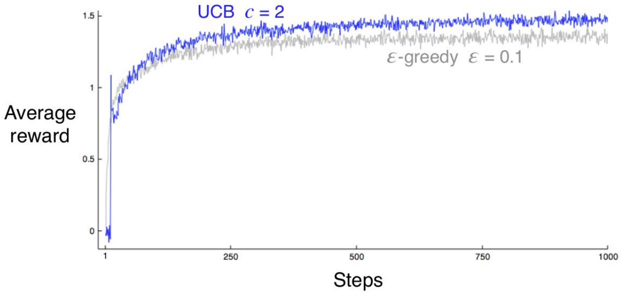
One can prove

Proposition

$$\limsup E(T_k(n)) \geq \log(n) KL(\nu_k, \nu^*)^{-1},$$

which allows to see that UCB is asymptotically optimal.

UCB Performance



Gradient bandits

Other proposal.
We define a preference function $H_t(a)$, and choose an action with the following probabilities (Boltzmann):

$$P(A_t = a) = \pi_a(t) = \frac{e^{H_t(a)}}{\sum_k e^{H_t(k)}}.$$

Gradient paradigm: How to update H?

Assume that we want to find $\mathbf{w}^*$ that minimizes $J(\mathbf{w})$.
Gradient Theorem shows that the iterations

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \bigtriangledown J(\mathbf{w})$$

converge, i.e., $\mathbf{w}_t \longrightarrow \mathbf{w}^*$

If $\bigtriangledown J(\mathbf{w}) = \mathbb{E}(f(s, \mathbf{w}^*))$, then Stochastic Gradient Theorem shows that the iterations

$$\mathbf{w}_{t+1} = \mathbf{w}_t + f(s, \mathbf{w}^*)$$

converge too.

Gradient paradigm

If the actual reward of a is better than the estimated mean reward, increase the preference of a and decrease the others.

Proposition

$$\frac{\partial E(R_t)}{\partial H_t(a)} = E\Big(R_t(1_{A_t=a} - \pi_a(t))\Big).$$

With baseline:

$$\frac{\partial E(R_t)}{\partial H_t(a)} = E\Big((R_t - \bar{R}_t)(1_{A_t=a} - \pi_a(t))\Big).$$

An exact gradient descent would be:

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

a stochastic gradient version:

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