CS 4/5789 23 March 2022 Prof Sarah Dean

Lecture 17: Multi-armed Bandits

1) Interactive coding demo - jupyter notebook

2) Formal Setting

Simplified RL setting with no state and no transitions

$\mathcal{A}: 1, 2, \dots, K$ K discrete actions ("arms")

$r: \mathcal{A} \to \Delta(\mathbb{R})$ noisy reward $r_t \sim r(a_t)$
denote $\mathbb{E}[r(a)] = \mu_a$

$T: \mathbb{Z}_+$ integer time horizon

Goal: maximize cumulative expected reward

$$\mathbb{E}\left[\sum_{t=1}^{T} r(a_t)\right] = \sum_{t=1}^{T} \mu_{a_t}$$

What is the optimal action?

$$a^* = \arg\max_{a=1,\dots,K} \mu_a$$

This very simple MDP is easy to solve if rewards are known. When rewards are unknown, we must devise a strategy for balancing exploration (trying out different actions) against exploitation (selecting actions that perform well).

We measure the performance of a strategy, or algorithm, by comparing it against the optimal action.

Definition (Regret):
The regret of an algorithm which chooses actions $a_1, \dots, a_T$ is

$$R(T) = \mathbb{E}\left[\sum_{t=1}^{T} r(a^*) - r(a_t)\right] = \sum_{t=1}^{T} \mu^* - \mu_{a_t}$$

Our goal is to find algorithms with <u>sublinear</u> regret. That way, the average suboptimality converges to 0:

$$\lim_{T\to\infty} \frac{R(T)}{T} \to 0 \quad \text{if } R(T) \text{ sublinear} \quad \text{e.g. } R(T) \lesssim T^p \text{ for } p<1.$$

3) Balancing exploration & exploitation

Consider the following two algorithms:

Alg 1: Random
for $t=1,\dots,T$
 $a_t \sim \text{unif}(1,\dots,K)$

pure explore

Alg 2: Greedy
for $t=1,\dots,K$:
 $a_t = t$
 $r_t \sim r(a_t)$
for $t=K+1,\dots,T$
 $a_t = \operatorname{argmax}_{a\in 1,\dots,K} r_a$

pure exploit

Both of these suffer from <u>linear</u> regret.

Why?

$$R(T) = \sum_{t=1}^{T} \mathbb{E}[r(a^*) - r(a_t)] =$$

$$= \sum_{t=1}^{T} \mathbb{E}[\mathbb{1}\{a_t \neq a^*\}(r(a^*) - r(a_t))]$$

$$= \sum_{t=1}^{T} \mathbb{P}\{a_t \neq a^*\}(\mu^* - \mu_{a_t})$$

$$\geq \sum_{t=1}^{T} \underbrace{\mathbb{P}\{a_t \neq a^*\}}_{\substack{\text{probability of} \\ \text{not pulling } a^* \\ \text{(constant for alg 1\&2)}}} \cdot \underbrace{\min_{a\neq a^*} (\mu^* - \mu_a)}_{\substack{\text{smallest gap} \\ \text{(constant)}}} = C \cdot T$$

Exercise: what is $\mathbb{P}\{a_t \neq a^*\}$ for Alg 1 & 2?

Alg 3: Explore-then-commit:

For $t = 1, \dots, N \cdot K$

$\quad a_t = t \bmod K$ } pull each arm N times

$\hat{\mu}_a = \frac{1}{N} \sum_{i=1}^{N} r_{k \cdot i}$ } compute average reward

For $t = N \cdot K + 1, \dots, T$

$\quad a_t = \arg\max_a \hat{\mu}_a = \hat{a}^*$

This algorithm balances exploration & exploitation.

How to set N?

Let's do some analysis.

<u>Lemma</u> (Hoeffdings): Suppose $r_i \in [0, 1]$ and $\mathbb{E}[r_i] = \mu$. Then for $r_1, \dots, r_N$ iid, with probability $1-\delta$,

$$\left| \frac{1}{N} \sum_{i=1}^{N} r_i - \mu \right| \lesssim \sqrt{\frac{\log(1/\delta)}{N}}$$

Proof is out of scope.

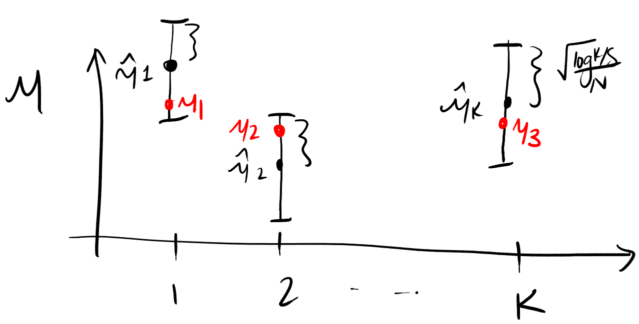
<u>Lemma (Explore)</u>: After exploration phase, for all arms $a = 1, \dots, K$,

$$|\hat{\mu}_a - \mu_a| \lesssim \sqrt{\frac{\log(K/\delta)}{N}} \quad \text{with probability } 1-\delta.$$

<u>Proof</u>: Hoeffding & union Bound $P(A \cap B) \leq P(A) + P(B)$.

This gives us $1-\delta$ confidence intervals:

$$\mu_a \in \left[\hat{\mu}_a \pm c\sqrt{\frac{\log(K/\delta)}{N}} \right]$$



The regret decomposes:

$$R(T) = \sum_{t=1}^{T} \mu^* - \mu_{a_t} = \underbrace{\sum_{t=1}^{NK} \mu^* - \mu_{a_t}}_{R_1} + \underbrace{\sum_{t=NK+1}^{T} \mu^* - \mu_{\hat{a}^*}}_{R_2}$$

for rewards bounded $[0, 1]$, $R_1 \leq NK$

We use confidence intervals to bound R_2.

$$R_2 = (T-NK)(\mu^* - \mu_{\hat{a}^*}) \leq (T-NK)\Big[\underbrace{\hat{\mu}_{a^*} + \sqrt{\tfrac{\log(K/\delta)}{N}}}_{\text{upper confidence bound}} - \Big(\underbrace{\hat{\mu}_{\hat{a}^*} - \sqrt{\tfrac{\log(K/\delta)}{N}}}_{\text{lower confidence bound}}\Big)\Big]$$

$$= (T-NK)\Big(\underbrace{\hat{\mu}_{a^*} - \hat{\mu}_{\hat{a}^*}}_{\leq 0 \text{ by definition of } \hat{a}^*} + 2\sqrt{\tfrac{\log(K/\delta)}{N}}\Big)$$

Combining everything, we have

$$R(T) = R_1 + R_2 \leq \underbrace{NK}_{\text{explore cost}} + \underbrace{2T\sqrt{\tfrac{\log K/\delta}{N}}}_{\text{exploit cost (if wrong)}} \qquad \text{w.p. } 1-\delta$$

Minimizing this upper bound with respect to N,

$$N = \left(\frac{T}{2K}\sqrt{\log(K/\delta)}\right)^{2/3} \quad \text{and w.p. } 1-\delta,$$

$$R(T) \lesssim T^{2/3} K^{1/3} \log^{1/3}\left(\frac{K}{\delta}\right) \quad \text{for explore-then-commit}$$

subLinear!

Next Lecture: consider confidence intervals directly in our algorithm.