Bandits: Part II
Adversarial Bandits

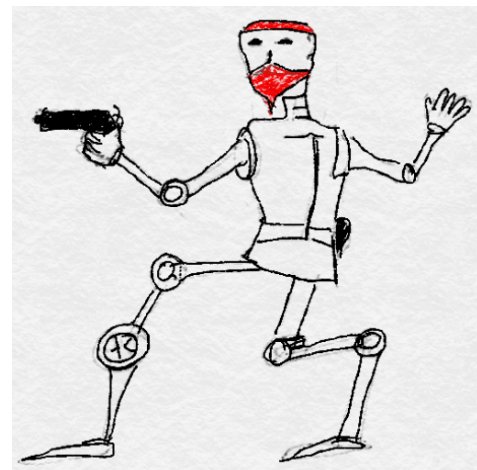
Csaba Szepesvári

Department of Computing Science & AICML
University of Alberta & (@ Deepmind since August)
csaba.szepesvari@ualberta.ca

UNIVERSITY OF ALBERTA

September 1, 2017

Summer School @ DS3



Outline

Outline

Overview of Talks

- Talk 1: Basics
  - What, why, applications, Q&A
  - Bandits: Problem definition
  - Stochastic finite-armed bandits: Basics
  - Measure concentration. Subgaussianity
  - Explore-then-commit
  - UCB, Optimism, Optimality
- **Talk 2**: Adversarial bandits and lower bounds
  - Adversarial finite-armed bandits
  - Lower bounds
  - Contextual bandits
  - Exp4: Prediction with Expert Advice
- Talk 3: Linear bandits

Outline

Problem Definition

- Stochastic bandits: Reward at time t is $X_t \sim P_{A_t}$, where $A_t \in [K] = \{1, \dots, K\}$ is the action and $(P_1, \dots, P_K)$ are some fixed distributions over the reals.
- Why is this a good model?
- Will this assumption be ever met in practice?
- What if it is not met?

- Extreme viewpoint:
  - Rewards are assigned *arbitrarily* to the arms ahead of time.
  - Learner picks one of the arms, sees the corresponding reward, but not the others.
  - Goal is to compete with best fixed action in hindsight.

Adversarial Bandits: Formal Definition

Definition (Adversarial Environment)

An environment ν is a point in $([0,1]^K)^n$: $\nu = (x_1, \ldots, x_n)$ where $x_t \in [0,1]^K$.

Interaction

Round $t = 0$: Environment chooses $x_1, \ldots, x_n \in [0,1]^K$.
For rounds $t = 1, 2, 3, \ldots$

1. Based on its past observations (if any), the learner chooses an action $A_t \in [K]$. The chosen action is sent to the environment.
2. The environment sends the reward $X_t = x_{t,A_t}$ to the learner.

Regret: $R_n = \mathbb{E}\left[\max_i \sum_{t=1}^n x_{t,i} - \sum_{t=1}^n X_t\right]$.

Notes

- $x_{t,i} \in [0,1]$ is the reward gained by a learner/policy if action i is chosen in round t.
- Why adversarial? We want small regret in the "worst-case".
  - Choose policy (=algorithm) π.
  - Then choose ν that maximizes $R_n(\nu, \pi)$ ("fitted" to the spec. π).

Regret: $R_n = \max_i \sum_{t=1}^{n} x_{t,i} - \mathbb{E}\left[\sum_{t=1}^{n} X_t\right]$.

Notes

- $x_{t,i}$ is lower case: Not random. Does this matter? Are we losing generality?
- A_t is upper case: In general, random. Does this matter?
- How about reactive opponents?

Exercise 1

Show that any deterministic policy has a worst-case regret of at least $R_n^*(\pi) \geq n(1 - 1/K)$.

Hint: Given π, choose ν so that $x_{t,A_t} = 0$ for all t and $x_{t,i} = 1$ for $i \neq A_t$. Valid?

Central questions

Let $R_n^*(\pi)$ be the worst-case regret of policy π:

$$R_n^*(\pi) = \sup_{\nu \in [0,1]^{nK}} R_n(\pi, \nu)\,.$$

Some important questions:

- Does there exists a policy π s.t. $R_n^*(\pi) = o(n)$? (Yes)
- How small can we make $R_n^*(\pi)$? ($\Theta(\sqrt{Kn})$)
- Which algorithm achieves the optimal regret growth rate? ($\approx$ Exp3)

Stochastic vs. Adversarial Regret

Let $X_{t,i} \sim P_i$, $i \in [K]$, $X_t \doteq X_{t,A_t}$ ("skipping reward model").

We have

$$\mathbb{E}\left[\max_i \sum_{t=1}^n X_{t,i} - X_{t,A_t}\right] \geq \max_i \mathbb{E}\left[\sum_{t=1}^n X_{t,i} - X_{t,A_t}\right]. \qquad (1)$$

- LHS = expected adversarial regret.
- RHS = expected regret as defined for stochastic bandits.

What can be concluded?

$$R_n^* = \inf_\pi \sup_{\nu \in [0,1]^{nK}} R_n(\pi, \nu).$$

Also: $\inf_\pi \sup_{\nu\in[0,1]^{nK}} R_n(\pi,\nu) \geq c\sqrt{nK}$, where $c > 0$ is a universal constant.

Outline

The Exp3 Algorithm

"**E**xponential-weight algorithm for **E**xploration and **E**xploitation"

$\equiv$

Exp3.

Two ingredients:

1. Estimating unseen rewards.
2. Using reward estimates to produce a distribution over the actions (good algorithms must randomize).

Estimating Rewards

Problem: Only $X_t = X_{t,A_t}$ is seen. It would be nice to have some estimates of $x_{t,i}$ for all $i \in [K]$.

Let

$$P_{ti} = \mathbb{P}\left(A_t = i | X_1, \ldots, X_{t-1}, A_1, \ldots, A_{t-1}\right) .$$

Two observations:

- This is random. Why?
- This is what we choose when we design an algorithm!

Define:

$$\hat{X}_{ti} = \frac{\mathbb{I}_{\{A_t = i\}} X_t}{P_{ti}} . \tag{2}$$

Lemma

We have $\mathbb{E}_t\left[\hat{X}_{ti}\right] = x_{ti}$, *where* $\mathbb{E}_t[\cdot]$ *is defined as* $\mathbb{E}_t[Z] \doteq \mathbb{E}\left[Z | A_1, X_1, \ldots, A_{t-1}, X_{t-1}\right]$.

Proof of Unbiasedness

Claim: $\mathbb{E}_t\left[\hat{X}_{ti}\right] = x_{ti}$. Proof: Write $A_{ti} \doteq \mathbb{I}_{\{A_t=i\}}$. Then, $X_t A_{ti} = x_{t,i} A_{ti}$ and

$$\hat{X}_{ti} = \frac{A_{ti}}{P_{ti}} x_{ti} \, .$$

Now $\mathbb{E}_t[A_{ti}] = P_{ti}$ and since P_{ti} is a function of $A_1, X_1, \ldots, A_{t-1}, X_{t-1}$ we get

$$\mathbb{E}_t\left[\hat{X}_{ti}\right] = \mathbb{E}_t\left[\frac{A_{ti}}{P_{ti}} x_{ti}\right] = \frac{x_{ti}}{P_{ti}} \mathbb{E}_t[A_{ti}] = \frac{x_{ti}}{P_{ti}} P_{ti} = x_{ti} \, .$$

Qu.e.d. Corollary: We have $\mathbb{E}[\hat{X}_{ti}] = x_{ti}$.

Variance

Let $\mathbb{V}_t[\cdot]$ be: $\mathbb{V}_t[U] \doteq \mathbb{E}_t[(U - \mathbb{E}_t[U])^2]$.

Question: How big is $\mathbb{V}_t[\hat{X}_{ti}]$?

Why care? Ultimately, the variance determines learning speed.

Conditional vs. unconditional variance:

Exercise 2: $\mathbb{E}\left[\mathbb{V}_t\left[\hat{X}_{ti}\right]\right] = \mathrm{Var}\left[\hat{X}_{ti}\right]$.

Hint: Use $\mathbb{E}_t[\hat{X}_{ti}] = \mathbb{E}[\hat{X}_{ti}] = x_{ti}$ and $\mathbb{V}[U] = \mathbb{E}[U^2] - \mathbb{E}[U]^2$ (which also holds for $\mathbb{V}_t[\cdot]$).

On Variance

Lemma

$$\mathbb{V}_t\left[\hat{X}_{ti}\right] = \frac{x_{ti}^2(1-P_{ti})}{P_{ti}}\,.$$

Proof: Recall

$$\hat{X}_{ti} = \frac{A_{ti}}{P_{ti}}\,x_{ti}\,, \qquad A_{ti} = \mathbb{I}_{\{A_t=i\}}\,.$$

Hence,

$$\mathbb{V}_t\left[\hat{X}_{ti}\right] = \mathbb{E}_t\left[\hat{X}_{ti}^2\right] - x_{ti}^2 = \mathbb{E}_t\left[\frac{A_{ti}x_{ti}^2}{P_{ti}^2}\right] - x_{ti}^2 = \frac{x_{ti}^2(1-P_{ti})}{P_{ti}}\,. \tag{3}$$

Note: The variance can be quite large if P_{ti} is small! Trouble!?

Loss-based Estimates

An alternate estimator:

$$\hat{X}_{ti} = 1 - \frac{\mathbb{I}_{\{A_t=i\}}}{P_{ti}} (1 - X_t) . \tag{4}$$

Claim: $\mathbb{E}_t \left[\hat{X}_{ti} \right] = x_{ti}$.

Let $y_{ti} = 1 - x_{ti}$, $Y_t = 1 - X_t$, $\hat{Y}_{ti} = 1 - \hat{X}_{ti}$, then

$$\hat{Y}_{ti} = \frac{\mathbb{I}_{\{A_t=i\}}}{P_{ti}} Y_t .$$

Same formula as before just $X_t \leftrightarrow Y_t$, $X_{ti} \leftrightarrow Y_{ti}$!

y_{ti}, Y_t: "losses"

Claim: With $\hat{X}_{ti}$ as in (4), $\mathbb{V}_t \left[\hat{X}_{ti} \right] = \mathbb{V}_t \left[\hat{Y}_{ti} \right] = y_{ti}^2 \frac{1-P_{ti}}{P_{ti}}$.

Question: Is this better than the previous bound? When?

Loss vs. Reward-based Estimators: II

Reward-based estimator

$$\hat{X}_{ti} = \frac{\mathbb{I}_{\{A_t=i\}} X_t}{P_{ti}}.$$

Variance:

$$\mathbb{V}_t\left[\hat{X}_{ti}\right] = x_{ti}^2 \frac{1-P_{ti}}{P_{ti}}.$$

Range: $\hat{X}_{ti} \in [0, \infty)$.

Loss-based estimator

$$\hat{X}_{ti} = 1 - \frac{\mathbb{I}_{\{A_t=i\}}}{P_{ti}} (1 - X_t).$$

Variance:

$$\mathbb{V}_t\left[\hat{Y}_{ti}\right] = y_{ti}^2 \frac{1-P_{ti}}{P_{ti}}.$$

Range: $\hat{X}_{ti} \in (-\infty, 1]$.

Promise: Overestimating $\hat{X}_{ti}$ is pricey, loss-based estimators will have an edge!

Probability Computation

Let

$$\hat{S}_{ti} \doteq \sum_{s=1}^{t} \hat{X}_{si} \,.$$

Define

$$P_{ti} \doteq \frac{\exp(\eta \hat{S}_{t-1,i})}{\sum_j \exp(\eta \hat{S}_{t-1,j})} \,. \tag{5}$$

Alternative names:

- Exponential weighting, Hedge
- "Boltzmann exploration" / "Gibbs exploration"
- Mirror descent with negentropy regularized on the simplex
- Etc.

Exp3: The Algorithm

Input: n, K, η

Set $\hat{S}_{0i} = 0$ for all i

For $t \in \{1, \ldots, n\}$:

Calculate sampling distribution P_t:

$$P_{ti} = \frac{\exp\left(\eta \hat{S}_{t-1,i}\right)}{\sum_{j=1}^{K} \exp\left(\eta \hat{S}_{t-1,j}\right)}$$

Sample $A_t \sim P_t$ and observe reward X_t

Calculate $\hat{S}_{ti}$:

$$\hat{S}_{ti} = \hat{S}_{t-1,i} + 1 - \frac{\mathbb{I}_{\{A_t = i\}}(1 - X_t)}{P_{ti}}\,.$$

Outline

Regret Upper Bound for Exp3

Theorem

For an arbitrary assignment $(x_{ti})_{ti} \in [0,1]^{nK}$ *of rewards, the expected regret of Exp3 with an appropriate choice of* η*, satisfies*

$$R_n \leq 2\sqrt{nK \log(K)}\,.$$

Note: This is nearly optimal (apart from $\log K$ and a constant factor).

Proof Start

Fix $i \in [K]$.

$$R_{ni} = \sum_{t=1}^{n} x_{ti} - \mathbb{E}\left[\sum_{t=1}^{n} X_t\right]$$
$$= \mathbb{E}\Big[\underbrace{\sum_{t} \hat{X}_{ti}}_{\hat{S}_{ni} :=}\Big] - \mathbb{E}\Big[\underbrace{\sum_{t,i} P_{ti}\hat{X}_{ti}}_{\hat{S}_{n} :=}\Big],$$

because $\mathbb{E}[X_t] = \mathbb{E}\left[\sum_i P_{ti}\hat{X}_{ti}\right]$ thanks to

$$\mathbb{E}_t[X_t] = \mathbb{E}_t\left[\sum_i A_{ti}x_{ti}\right] = \sum_i P_{ti}x_{ti} = \sum_i P_{ti}\mathbb{E}_t\left[\hat{X}_{ti}\right].$$

Let's bound $\exp(R_{ni})$, or $\exp(\eta\hat{S}_{ni})$.

Bringing in the Algorithm

Define $W_t \doteq \sum_j \exp(\eta \hat{S}_{tj})$. For $t = 0$, $\hat{S}_{0i} = 0$, so $W_0 = K$.
Thus,

$$\exp(\eta \hat{S}_{ni}) \leq \sum_j \exp(\eta \hat{S}_{ni}) = W_n = W_0 \cdot \frac{W_1}{W_0} \cdot \cdots \cdot \frac{W_n}{W_{n-1}} .$$

Now,

$$\frac{W_t}{W_{t-1}} = \sum_j \frac{\exp(\eta \hat{S}_{t-1,j})}{W_{t-1}} \exp(\eta \hat{X}_{tj}) = \sum_j P_{tj} \exp(\eta \hat{X}_{tj}) .$$

Since $\hat{X}_{tj} \leq 1$ and $\exp(x) \leq 1 + x + x^2$ for $x \leq 1$, and also
$1 + x \leq \exp(x)$, $x \in \mathbb{R}$:

$$\frac{W_t}{W_{t-1}} \leq 1 + \eta \sum_j P_{tj} \hat{X}_{tj} + \eta^2 \sum_j P_{tj} \hat{X}_{tj}^2 \leq e^{\eta \sum_j P_{tj} \hat{X}_{tj} + \eta^2 \sum_j P_{tj} \hat{X}_{tj}^2} .$$

Bringing in the Algorithm - II.

Combine $\exp(\eta \hat{S}_{ni}) \leq W_0 \cdot \frac{W_1}{W_0} \cdot \cdots \cdot \frac{W_n}{W_{n-1}}$, $W_0 = K$, and

$$\frac{W_t}{W_{t-1}} \leq e^{\eta \sum_j P_{tj} \hat{x}_{tj} + \eta^2 \sum_j P_{tj} \hat{x}_{tj}^2},$$

to get

$$\exp\left(\eta \hat{S}_{ni}\right) \leq K \exp\left(\eta \hat{S}_n + \eta^2 \sum_{t,j} P_{tj} \hat{X}_{tj}^2\right).$$

Algebra:

$$\hat{S}_{ni} - \hat{S}_n \leq \frac{\log(K)}{\eta} + \eta \sum_{t,j} P_{tj} \hat{X}_{tj}^2.$$

Finishing Steps

We got:

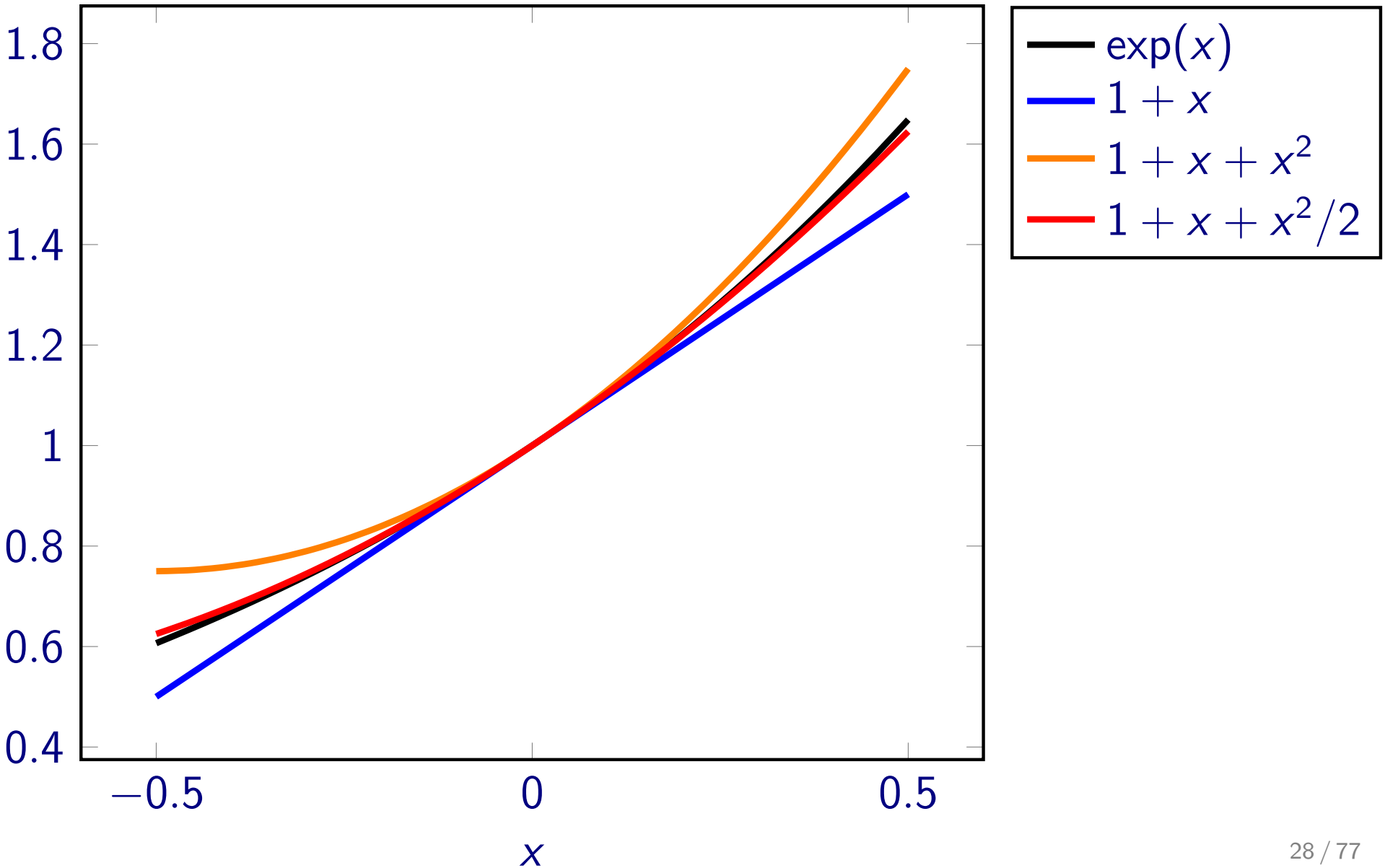
$$\hat{S}_{ni} - \hat{S}_n \leq \frac{\log(K)}{\eta} + \eta \sum_{t,j} P_{tj} \hat{X}_{tj}^2 \,. \tag{6}$$

Tedious, but simple calculation gives $\mathbb{E}\left[\sum_{t,j} P_{tj} \hat{X}_{tj}^2\right] \leq nK$. Thus,

$$R_{ni} \leq \frac{\log(K)}{\eta} + \eta n K \,.$$

Choosing $\eta = \sqrt{\log(K)/(nK)}$ gives the claimed result. Qu.e.d.

Going second order



Improved Bound

Theorem

For an arbitrary assignment $(x_{ti})_{ti} \in [0,1]^{nK}$ *of rewards, with* $\eta = \sqrt{\log(K)/(2nK)}$*, the expected regret of Exp3 satisfies*

$$R_n \leq \sqrt{2nK \log(K)}\,.$$

Compare with: $R_n \leq 2\sqrt{nK \log(K)}$.
The above bound predicts that the regret is 70% of this latter bound (*multiplicative* improvement!)

Note: We cannot use $\exp(x) \leq 1 + x + \frac{1}{2}x^2$ *unless* we use the loss-based estimator which ensures $\hat{X}_{ti} \leq 1$ (an *upper* bound).

Smaller learning rate than before.

Exercise 3: Investigate whether this translates into an actual advantage.

Outline

The Goal and the Problem

Goal: Controlling the upper tail of

$$\hat{R}_{ni} = \sum_{t=1}^{n} x_{ti} - \sum_{t=1}^{n} X_t \,.$$

Problem: Exp3 controls this in a poor manner.

Why? The algorithm uses $\sum_t \hat{X}_{ti}$. This has a huge variance.

In particular, we have $\mathbb{V}_t[\hat{X}_{ti}] \sim 1/P_{ti}$, so perhaps $\mathbb{V}[\sum_t \hat{X}_{ti}]$ is close $\mathbb{E}[\sum_t 1/P_{ti}]$.

Problematic when P_{ti} is small!

Exp3-IX: Exp3 with Implicit Exploration

Input: n, K, η, γ

Set $\hat{L}_{0i} = 0$ for all i

For $t \in \{1, \ldots, n\}$:

Calculate sampling distribution P_t:

$$P_{ti} = \frac{\exp\left(-\eta \hat{L}_{t-1,i}\right)}{\sum_{j=1}^{K} \exp\left(-\eta \hat{L}_{t-1,j}\right)}$$

Sample $A_t \sim P_t$ and observe reward X_t

Calculate $\hat{L}_{ti}$:

$$\hat{L}_{ti} = \hat{L}_{t-1,i} + \frac{\mathbb{I}_{\{A_t = i\}}(1 - X_t)}{P_{t-1,i} + \gamma}$$

Optimism!

Regret Upper Bounds for Exp3-IX

Theorem

Let $\hat{R}_n$ be the random regret of Exp3-IX run with $\gamma = \eta/2$. Then, choosing $\eta = \sqrt{\frac{2\log(K+1)}{nK}}$, for any $0 \leq \delta \leq 1$, the inequality

$$\hat{R}_n \leq \sqrt{8.5nK\log(K+1)} + \left(\sqrt{\frac{nK}{2\log(K+1)}} + 1\right)\log(1/\delta) \quad (7)$$

hold with probability at least $1-\delta$. Further, for any $0 \leq \delta \leq 1$, if $\eta = \sqrt{\frac{\log(K)+\log(\frac{K+1}{\delta})}{nK}}$, then

$$\hat{R}_n \leq 2\sqrt{(2\log(K+1)+\log(1/\delta))nK} + \log\left(\frac{K+1}{\delta}\right) \quad (8)$$

holds with probability at least $1-\delta$.

Notes – Summary

- An expected regret bound can also be given.
- We can use these algorithms with decreasing "learning rates" ($\eta \to (\eta_t)_t$, $\gamma \to (\gamma_t)_t$).
- How do these algorithms work in stochastic environments?
  - Best of both worlds: Bubeck and Slivkins (2012); Auer and Chiang (2016); Seldin and Lugosi (2017).

Bibliographic Remarks

- Adversarial bandits, Exp3: Auer et al. (1995)
- Exp3-IX: Neu (2015)
- Removing $\sqrt{\log(K)}$ from the upper bound; MOSS algorithm: Audibert and Bubeck (2009).
- Second order bounds: Hazan and Kale (2011)

Outline

Minimax Regret

- $\mathcal{A}_{n,K}$: Set of all possible policies on horizon n and action set $[K]$.
- $\mathcal{E}$: Set of environments.

Definition (Minimax Regret for Horizon n)

$$R_n^*(\mathcal{E}) = \inf_{A \in \mathcal{A}_{n,K}} \sup_{\nu \in \mathcal{E}} R_n(A, \nu)\,.$$

For any policy A is chosen, $R_n^*(\mathcal{E}) \leq R_n^*(A, \mathcal{E}) \doteq \sup_{\nu \in \mathcal{E}} R_n(A, \nu)$.

Thus, $R_n^*(\mathcal{E})$ captures the fundamental hardness of the problem.

Default Environment Class

Subgaussian environments:

$$\mathcal{E}_K = \Big\{ (P_i)_{i=1}^K : P_i \text{ has 1-subgaussian tails and } \max_{i,j} \mu_i - \mu_j \leq 1 \text{ where } \mu_i \text{ is the mean of } P_i \Big\}.$$

MOSS bound: $R_n^*(\mathcal{E}_K) \leq R_n(\mathrm{MOSS}, \mathcal{E}) \leq C\sqrt{nK}$.

Minimax Regret Lower Bound

Theorem

There exists a universal constant $c > 0$ such that for any number of arms $K \geq 2$ and horizon $n \geq 2$,

$$R_n^*(\mathcal{E}_K) \geq c \min(n, \sqrt{Kn}) \,. \tag{9}$$

Background: Hypothesis Testing I.

Let $X_1, X_2, \ldots, X_n \sim \mathcal{N}(\mu, 1)$, μ unknown.

Basic hypothesis testing question:
Given $X_1, \ldots, X_n$, decide between $\mu = 0$ and $\mu = \Delta$ for $\Delta > 0$, a known constant.

Let $\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} X_i$. By symmetry, we should choose $\mu = \Delta$ if $\hat{\mu} \geq \Delta/2$, otherwise choose $\mu = 0$.

What is the error probability, e_n, say, when $\mu = 0$?

Note: $e_n = \mathbb{P}(\hat{\mu} \geq \Delta/2)$.

Hypothesis Testing II.

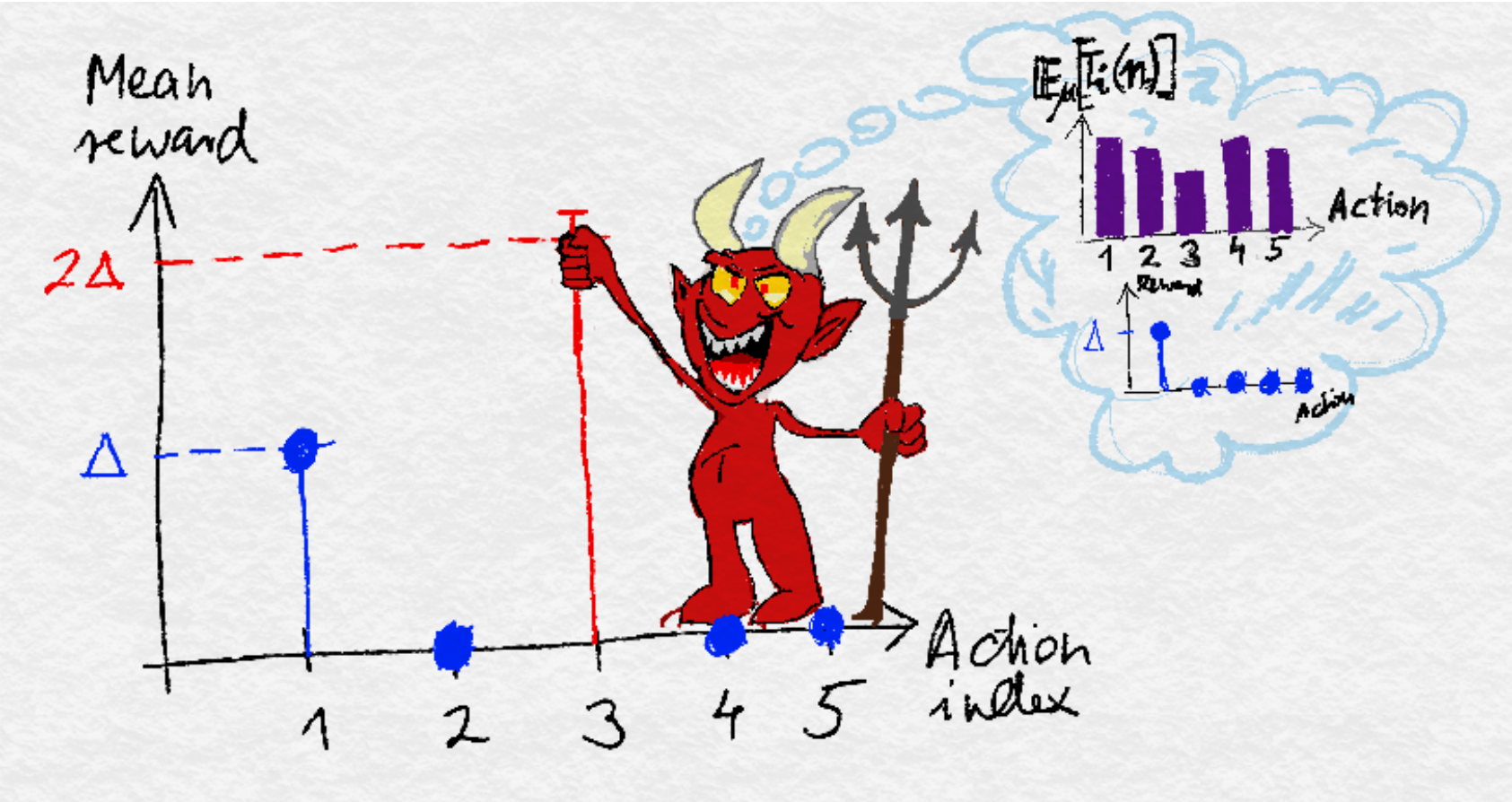
Note that $\hat{\mu} \sim \mathcal{N}(\mu, 1/n)$ ($\mathbb{V}[\hat{\mu}] = 1/n$).

Then

$$\frac{1}{\sqrt{n\Delta^2} + \sqrt{n\Delta^2 + 4}} \sqrt{\frac{2}{\pi}} \exp\left(-\frac{n\Delta^2}{8}\right) \leq \mathbb{P}\left(\hat{\mu} \geq \frac{\Delta}{2}\right)$$
$$\leq \frac{1}{\sqrt{n\Delta^2} + \sqrt{n\Delta^2 + 8/\pi}} \sqrt{\frac{2}{\pi}} \exp\left(-\frac{n\Delta^2}{8}\right).$$

Lesson: For $n \gg 8/\Delta^2$, e_n decays exponentially, while for $n \leq 8/\Delta^2$, $e_n \geq \text{const}$.

Picture Proof



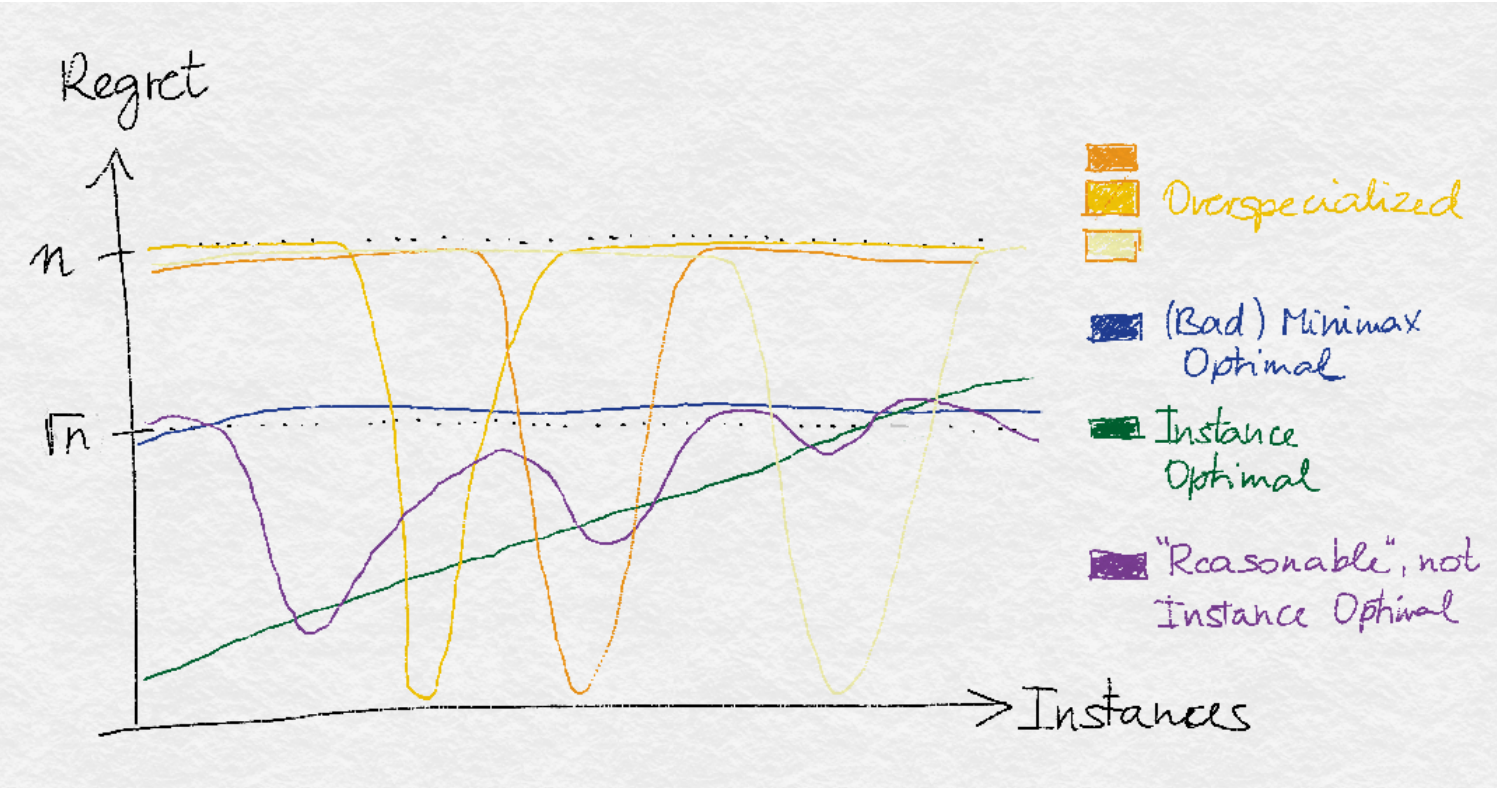
Is MOSS/UCB the "ultimate algorithm"?

We have $R_n^*(\mathcal{E}_K) \leq R_n(\mathrm{MOSS}, \mathcal{E}) \leq C\sqrt{nK}$.
How about UCB? $\sqrt{\log K}$ off in the upper bound..
Also:

$$\lim_{K\to\infty} \sup_{\nu\in\mathcal{E}_K} \frac{R_{K^3}(\mathrm{MOSS}, \nu)}{R_{K^3}(\mathrm{UCB}, \nu)} = \infty \text{ and } \lim_{n\to\infty} \sup_{\nu\in\mathcal{E}_K} \frac{R_n(\mathrm{UCB}, \nu)}{R_n(\mathrm{MOSS}, \nu)} = \infty$$

for all $K > 1$.

On Worst-Case Optimality



Outline

Instance-wise Asymptotic Bounds

What policies are reasonable??

Definition

A policy π is called consistent over a class of bandits $\mathcal{E}$ if for all $\nu \in \mathcal{E}$ and for all $p > 0$ it holds that

$$R_n(\pi, \nu) = O(n^p) \qquad \text{as } n \to \infty\,.$$

Denote the class of consistent policies over $\mathcal{E}$ by $\Pi_{\text{cons}}(\mathcal{E})$.

Instance-wise Asymptotic Bounds

Theorem

Let $\mathcal{E}$ be a class of bandits and $\pi \in \Pi_{cons}(\mathcal{E})$ be a consistent strategy over $\mathcal{E}$. Let $\nu \in \mathcal{E}$ be a bandit with mean vector μ and for each suboptimal arm i define

$$d_i(\nu, \mathcal{E}) = \inf_{\nu' \in \mathcal{E}} \left\{ \mathrm{D}(P_i(\nu), P_i(\nu')) : \mu_i(\nu) > \mu^*(\nu) \quad \text{and} \right.$$
$$\left. P_j(\nu) = P_j(\nu') \quad \text{for all } j \neq i \right\}.$$

Then $\liminf_{n\to\infty} \dfrac{R_n(\pi, \nu)}{\log(n)} \geq c^*(\nu, \mathcal{E})$ *where* $c^*(\nu, \mathcal{E}) = \displaystyle\sum_{i:\Delta_i>0} \frac{\Delta_i(\nu)}{d_i(\nu, \mathcal{E})}$.

Specific Bounds

Theorem

The following hold:

(a) *Let* $\sigma^2 > 0$ *and* $\mathcal{C} = \{\mathcal{N}(\mu, \sigma^2) : \mu \in \mathbb{R}\}$, *then* $d(\mathcal{N}(\mu, \sigma^2), \mathcal{C}, \mu^*) = \frac{(\mu-\mu^*)^2}{2\sigma^2}$.

(b) *Let* $\mathcal{C} = \{\mathcal{N}(\mu, \sigma^2) : \mu \in \mathbb{R}, \sigma^2 > 0\}$, *then* $d(\mathcal{N}(\mu, \sigma^2, \mathcal{C}, \mu) = \frac{1}{2} \log\left(1 + \frac{(\mu-\mu^*)^2}{\sigma^2}\right)$.

(c) *Let* $\mathcal{C} = \{\mathcal{B}(\mu) : \mu \in [0, 1]\}$, *then* $d(\mathcal{B}(\mu), \mathcal{C}, \mu^*) = \mathsf{D}(\mathcal{B}(\mu), \mathcal{B}(\mu^*))$.

(d) *Let* $\mathcal{C} = \{\mathcal{U}(a, b) : a, b \in \mathbb{R}\}$, *then* $d(\mathcal{U}(a, b), \mathcal{C}, \mu^*) = \log\left(1 + \frac{2((a+b)/2-\mu^*)^2}{b-a}\right)$.

Notes

- UCB is asymptotically optimal on Gaussian environments.
- The so-called KL-UCB algorithm is asymptotically optimal on Bernoulli environments.
- Finite-time instance optimality? What are the reasonable policies? E.g., their minimax regret is at most $\mathrm{const} \times \sqrt{nK}$.
- Similar techniques work for deriving high-probability regret lower bounds.

Bibliographic Remarks

- Gaussian tail bounds, see, e.g., Abramowitz and Stegun, 1964.
- Lower bounds, asymptotics: Lai and Robbins (1985); Graves and Lai (1997); Burnetas and Katehakis (1996).
- Finite time, instance dependent, e.g., Wu et al. (2015); Lattimore (2016)
- Minimax: Auer et al. (1995); Gerchinovitz and Lattimore (2016)

Outline

Fundamental Theorem of ML

Theorem ("Bias-Variance Dilemma")

Competing with a poor benchmark does not make sense since even an algorithm that perfectly matches the benchmark will perform poorly.

At the same time, competing with a better benchmark can be harder from a learning point of view and in a specific scenario the gain from aiming to match the performance of a better benchmark may very well be offset by the fact that algorithms that compete with stronger benchmark have to search in a larger space of possibilities.

Contextual Bandits

Interaction

Round $t = 0$: Environment chooses $x_1, \ldots, x_n \in [0,1]^K$, $c_1, \ldots, c_n \in \mathcal{C}$.
For rounds $t = 1, 2, 3, \ldots$:

1. Context c_t is revealed.
2. Based on its past observations (including c_t), the learner chooses an action $A_t \in [K]$. The chosen action is sent to the environment.
3. The environment sends the reward $X_t = x_{t,A_t}$ to the learner.

Goal: Maximize $\sum_{t=1}^{n} X_t$.

Who do you want to compete with? How to define the regret?

Outline

Best Action per Context

Let's compete with the best action per context:

$$S_n = \sum_{c \in \mathcal{C}} \max_{k \in [K]} \sum_{t: c_t = c} x_{t,k} \ ,$$

Note:

$$S_n = \max_{\varphi: \mathcal{C} \to [K]} \sum_{t=1}^{n} x_{t,\varphi(c_t)} \ . \tag{10}$$

Regret: $R_n = S_n - \sum_t X_t$, or

$$R_n = \sum_{c \in \mathcal{C}} \mathbb{E} \left[\max_{k \in [K]} \sum_{t: c_t = c} (x_{t,k} - X_t) \right] \ .$$

The Memory Hug Baseline

Assume $\mathcal{C}$ is finite and (relatively) small. Associate one *anytime* Exp3 to each context (we will use $|\mathcal{C}|$ times as much memory as a single Exp3).

When context $c_t \in \mathcal{C}$ arrives, the *local/private* Exp3 associated with it is called, to provide A_t.

$T^c(n)$: number of times context $c \in \mathcal{C}$ is seen during the first n rounds.
$R^c(s)$: the regret suffered by the instance of Exp3 associated with c at the end of the round when this instance is used s times.

Then:

$$R_n = \sum_{c \in \mathcal{C}} \mathbb{E}\left[\max_{k \in [K]} \sum_{t : c_t = c} (x_{t,k} - X_t)\right] = \sum_{c \in \mathcal{C}} \mathbb{E}\left[R^c\left(T^c(n)\right)\right] .$$

The Memory Hug Baseline: Regret

Claim: If we set η to $\eta_s = \sqrt{\log(K)/(sK)}$, then $R^c(s) \leq 2\sqrt{sK\log(K)}$ holds for *any* $s \leq 1$.

$$R_n = \sum_{c \in \mathcal{C}} \mathbb{E}\left[R^c(T^c(n))\right], \text{ and } R^c(s) \leq 2\sqrt{sK\log(K)}\ \forall c, s\,.$$

These imply:

$$R_n \leq 2\sqrt{K\log(K)} \sum_{c \in \mathcal{C}} \sqrt{T_c(n)}\,.$$

How big/how small?

The Memory Hug Baseline: Regret II.

$$R_n \leq 2\sqrt{K \log(K)} \sum_{c \in \mathcal{C}} \sqrt{T_c(n)}\,.$$

Best case (smallest upper bound): $T_c(n) = n$ for some context c, $R_n \leq 2\sqrt{Kn \log(K)}$: Same regret as before!

The Memory Hug Baseline: Worst Case

$$R_n \leq 2\sqrt{K \log(K)} \sum_{c \in \mathcal{C}} \sqrt{T_c(n)}\,. \tag{11}$$

Worst-case: $T_c(n) = n/|\mathcal{C}|$:

$$R_n \leq 2\sqrt{K \log(K)|\mathcal{C}|n}\,. \tag{12}$$

How much total reward?

$$\mathbb{E}\left[\sum_{t=1}^{n} X_t\right] \geq S_n - 2\sqrt{K \log(K)|\mathcal{C}|n}\,.$$

Note: S_n may be much bigger, but the regret is always positive for the first $4K \log(K)|\mathcal{C}|$ rounds, guarantee on the total reward is vacuous for all earlier time steps!

The Memory Hug Baseline: Summary

Conclusion: When $|\mathcal{C}|$ is large, we conclude that for a long time, the Memory-Hug algorithm may have a much *worse* total reward than if we just ran a single instance of Exp3.

Calibrating the Baseline

Write $\mathcal{P} = \{\mathcal{C}_1, \ldots, \mathcal{C}_p\}$, $\mathcal{C}_i \subset \mathcal{C}$, $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$ – a partition of $\mathcal{C}$.

Compete with best action per "cluster" $\mathcal{C}_i$:
$S_n = \sum_{i=1}^{p} \max_{k \in [K]} \sum_{t: c_t \in \mathcal{C}_i} x_{t,k}$.

Define

$$\Phi(\mathcal{P}) = \{\varphi : \mathcal{C} \to [K] \ : \ \forall c, c' \in \mathcal{C} \text{ s.t. } c, c' \in P \text{ for some } P \in \mathcal{P}, \\ \varphi(c) = \varphi(c')\}$$

– the set of functions the map contexts in the same partition to the same action.
Then:

$$S_n = \max_{\varphi \in \Phi(\mathcal{P})} \sum_{t=1}^{n} x_{t,\varphi(c_t)} \, .$$

Compare with $\max_{\varphi: \mathcal{C} \to [K]} \sum_{t=1}^{n} x_{t,\varphi(c_t)}$!

Tamed Memory Hug

One Exp3 per cluster $\mathcal{C}_i$. Regret:

$$R_n \leq 2\sqrt{K \log(K)} \sum_{P \in \mathcal{P}} \sqrt{T_P(n)}\,.$$

where $T_P(n)$ is the amount data for $P \in \mathcal{P}$.

Compare with

$$R_n \leq 2\sqrt{K \log(K)} \sum_{c \in \mathcal{C}} \sqrt{T_c(n)}\,.$$

Outline

Towards Bandits with Expert Advice

Partitioning: $S_n = \max_{\varphi \in \Phi(\mathcal{P})} \sum_{t=1}^{n} x_{t,\varphi(c_t)}$.
No partitioning $S_n = \max_{\varphi:\mathcal{C}\to[K]} \sum_{t=1}^{n} x_{t,\varphi(c_t)}$ (special partition, if you wish).

How about other sets of functions?

- Use "similarity of context"?
- Train your favourite supervised method off-line?

Bandits with Expert Advice

Interaction

Round $t = 0$: Environment chooses $x_1, \ldots, x_n \in [0,1]^K$.
For rounds $t = 1, 2, 3, \ldots$

1. Expert m chooses a distribution $E_m^{(t)}$ over $[K]$ (recommendations).
2. Based on its past observations (if any), the learner chooses an action $A_t \in [K]$. The chosen action is sent to the environment.
3. The environment sends the reward $X_t = x_{t,A_t}$ to the learner.

Goal: Compete with best expert.
Regret: $R_n = \mathbb{E}\left[\max_m \sum_{t=1}^n E_m^{(t)} x_t - \sum_{t=1}^n X_t\right]$.

Outline

Exp4

1. Inputs: $\eta, \gamma \geq 0$.
2. $Q_1 = (1/M, \ldots, 1/M) \in [0,1]^M$ (row vector).
3. In rounds $t = 1, 2, \ldots$ do:
   1. Receive "advice" $E^{(t)} \in [0,1]^{M \times K}$: Row m is $E^{(t)}_m$.
   2. Choose the action $A_t \sim P_{t,\cdot}$ at random, where $P_t = Q_t E^{(t)}$
   3. The reward $X_t = x_{t,A_t}$ is received
   4. The rewards of all the actions are estimated; say: $\hat{X}_{ti} = 1 - \frac{\mathbb{I}_{\{A_t = i\}}}{P_{ti} + \gamma}(1 - X_t)$
   5. Propagate the rewards to the experts: $\tilde{X}_t = E^{(t)} \hat{X}_t$
   6. The distribution Q_t is updated using exponential weighting: $Q_{t+1,i} = \frac{\exp(\eta \tilde{X}_{ti}) Q_{ti}}{\sum_j \exp(\eta \tilde{X}_{tj}) Q_{tj}}$, $i \in [M]$

Note: A_t can be chosen in two steps, first sampling M_t from Q_t and then choosing A_t from $E^{(t)}_{M_t,\cdot}$.

Regret Upper Bound for Exp4

Let

$$E_t^* = \sum_{s=1}^{t} \sum_i \max_{m'} E_{m',i}^{(s)} ,$$

Theorem (Regret of Exp4))

If η is chosen appropriately, the regret of Exp4 satisfies

$$R_n \leq \mathbb{E}\left[\sqrt{2\log(M)E_n^*}\right] . \tag{13}$$

This is achieved with $\eta = \sqrt{(2\log M)/E_n^*}$.
Feasible choice: $\eta_t = \sqrt{\log M/E_t^*}$; slightly increased regret bound ($\sqrt{2} \to 2$).

How Good/Bad?

Let $E_n^* = \sum_{s=1}^{n} \sum_i \max_{m'} E_{m',i}^{(s)}$. Then: $R_n \leq \mathbb{E}\left[\sqrt{2\log(M)E_n^*}\right]$.

Best case: All experts agree: $E_m^{(t)} = E_{m'}^{(t)}$ for all m, m'. Then,

$$E_n^* = \sum_{s=1}^{n} \sum_i E_{1,i}^{(s)} = n$$

and

$$R_n \leq \mathbb{E}\left[\sqrt{2\log(M)n}\right] .$$

How Good/Bad? - II.

Worst-case. Experts disagree.
Since $\max_{m'} E^{(s)}_{m',i} \leq 1$, $E^*_n \leq nK$. Then,

$$R_n \leq \mathbb{E}\left[\sqrt{2nK\log(M)}\right] .$$

Good with few actions.
Few experts? $\max_m E^{(t)}_{mi} \leq \sum_m E^{(t)}_{mi} \leq M$, we get $E^*_n \leq Mn$ and thus

$$R_n \leq \sqrt{2nM\log(M)}\,.$$

Corollary

$$R_n \leq \sqrt{2n\min(M,K)\log(M)}\,.$$

Outline

Exp4 vs. Memory Hug

Memory hug regret:

$$R_n \leq 2\sqrt{K \log(K)|\mathcal{C}|n}.$$

Take Exp4 with $M = K^{|\mathcal{C}|}$ experts: One for each map $\varphi : \mathcal{C} \to [K]$!
From $R_n \leq \sqrt{2n \min(M, K) \log(M)}$, since $\log(M) = |\mathcal{C}| \log(K)$, we get

$$R_n \leq \sqrt{2nK|\mathcal{C}| \log(K)}$$

– same as above.

Memory hug? What memory hug??

- Memory requirements for "memory hug": $O(|\mathcal{C}|K)$
- Memory requirements for Exp4: $O(M) = O(K^{|\mathcal{C}|})$!

When to use Exp4?

- Proofs (figuring out an upper bound);
- Few (good) experts, $|\mathcal{C}|$ is large.

Notes

- High probability bounds: Sure
- Lower bound: Matching in the worst-case sense (worst-case over all possible advices).
- Open: Exp4 adapts to how agreeing the experts are. Is this the best we can have?

Bibliographical Remarks

- History of contextual bandits: Tewari and Murphy (2017)
- Exp4 algorithm: Auer et al. (2002)
- The tighter bounds presented are due to McMahan and Streeter (2009)
- Exp4-IX: Neu (2015)
- Exp3.P: Beygelzimer et al. (2011)

Summary

- Adversarial bandits with oblivious environments

  Reactive environments?
- Exp3: $R_n = O(\sqrt{nK \log(K)})$
- Exp4 and Contextual Bandits
  - Bias-variance
  - Prediction with expert advice
  - Adaptation to expert alignment by learning rate tuning
- Lower bounds: Reduction to hypothesis testing (aka information theory, statistics)
- Afternoon: Linear bandits

References I

Abramowitz, M. and Stegun, I. A. (1964). *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*, volume 55. Courier Corporation.

Audibert, J.-Y. and Bubeck, S. (2009). Minimax policies for adversarial and stochastic bandits. In *Proceedings of Conference on Learning Theory (COLT)*, pages 217–226.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. (2002). The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32:48–77.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. (1995). Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*, pages 322–331. IEEE.

Auer, P. and Chiang, C. (2016). An algorithm with nearly optimal pseudo-regret for both stochastic and adversarial bandits. In *Proceedings of the 29th Conference on Learning Theory, COLT 2016, New York, USA, June 23-26, 2016*, pages 116–120.

Beygelzimer, A., Langford, J., Li, L., Reyzin, L., and Schapire, R. E. (2011). Contextual bandit algorithms with supervised learning guarantees. In *AISTATS*, pages 19–26.

Bubeck, S. and Slivkins, A. (2012). The best of both worlds: Stochastic and adversarial bandits. In *COLT*, pages 42.1–42.23.

Burnetas, A. and Katehakis, M. (1996). Optimal adaptive policies for sequential allocation problems. *Advances in Applied Mathematics*, 17:122–142.

References II

Gerchinovitz, S. and Lattimore, T. (2016). Refined lower bounds for adversarial bandits. In *Advances in Neural Information Processing Systems (NIPS)*.

Graves, T. and Lai, T. (1997). Asymptotically efficient adaptive choice of control laws in controlled Markov chains. *SIAM J. Contr. and Opt.*, 35(3):715–743.

Hazan, E. and Kale, S. (2011). Better algorithms for benign bandits. *Journal of Machine Learning Research*, 12(Apr):1287–1311.

Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22.

Lattimore, T. (2016). Regret analysis of the anytime optimally confident UCB algorithm. Technical report.

McMahan, H. B. and Streeter, M. J. (2009). Tighter bounds for multi-armed bandits with expert advice. In *COLT*.

Neu, G. (2015). Explore no more: Improved high-probability regret bounds for non-stochastic bandits. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28*, pages 3168–3176. Curran Associates, Inc.

Seldin, Y. and Lugosi, G. (2017). An improved parametrization and analysis of the EXP3++ algorithm for stochastic and adversarial bandits. In *COLT*, pages 1743–1759.

Tewari, A. and Murphy, S. A. (2017). From ads to interventions: Contextual bandits in mobile health. In *Mobile Health - Sensors, Analytic Methods, and Applications*, pages 495–517.

Wu, Y., György, A., and Szepesvári, C. (2015). Online learning with gaussian payoffs and side observations. In *NIPS*, pages 1360–1368.