CMSC 858G: Bandits, Experts and Games 11/03/16

Lecture 9: (Semi-) bandits with linear costs (part II)

Instructor: Alex Slivkins Scribed by: Nicholas Labich

In the second half of this lecture we will study the best-expert problem with linear costs. We will define an algorithm, called Follow the Perturbed Leader (FPL), and prove a regret bound against an oblivious adversary. In particular, this will suffice to complete the proof of Theorem 3.3 from part I of this class.

1 Setup: best-expert problem with linear costs

To be applicable for combinatorial (semi-)bandits, the actions need to be binary vectors. We will posit a more general setting in which the actions can be vectors $a \in [0,1]^d$. More precisely, there is a fixed and known subset $\mathcal{A} \subset [0,1]^d$ of feasible actions. The costs are linear, in the sense that $c_t(a) = a \cdot v_t$ for each round t and each action a, where v_t is the *cost vector* (same for all actions).

Thus, each round t proceeds as follows:

- the adversary chooses the cost vector v_t,
- the algorithm chooses some feasible action $a_t \in \mathcal{A}$, receiving cost $c_t(a_t) = a_t \cdot v_t$,
- the cost vector v_t is revealed.

For formal results, we need to posit an upper bound on the costs. For ease of notation, we assume that $v_t \in [0, U/d]^d$, for some known parameter U. Then $c_t(a) \leq U$ for each action a.

We assume there exists an *optimization oracle*: a subroutine which computes the best action for a given cost vector. Formally, we represent this oracle as a function M from cost vectors to feasible actions such that $M(v) \in \operatorname{argmin}_{a \in \mathcal{A}} a \cdot v$ (ties can be broken arbitrarily). As explained earlier, while in general the oracle is solving an NP-hard problem, polynomial-time algorithms exist for important special cases such as shortest paths. Thus, the implementation of the oracle is domain-specific, and is irrelevant to our analysis.

We will prove the following theorem:

Theorem 1.1. FPL *achieves regret* $\mathbb{E}[R(T)] \leq 2U \cdot \sqrt{dT}$. *The running time of* FPL *in each round is polynomial in d plus one call to the oracle.*

Remark 1.2. The set of feasible actions $\mathcal{A}$ can be infinite (as long as the oracle is provided). For example, if $\mathcal{A}$ is defined by a finite number of linear constraints, the oracle can be implemented via linear programming. Note that Hedge cannot handle infinitely many actions.

We use shorthand $v_{i:j} = \sum_{t=i}^{j} v_t \in \mathbb{R}^d$ to denote the total cost vector between rounds i and j.

2 Defining the algorithm

First attempt. Consider a simple, exploitation-only algorithm called *Follow the Leader*:

$$a_{t+1} = M(v_{1:t}).$$

Equivalently, we play an arm with the lowest average cost, based on the observations so far. As we've discussed in Lecture 7, this approach works for fine IID costs. But it breaks for adversarial costs, as illustrated by the following example:

$$\begin{aligned}
\mathcal{A} &= \{(1,0),\ (0,1)\} \\
v_1 &= (\tfrac{1}{3}, \tfrac{2}{3}) \\
v_t &= \begin{cases} (1,0) & \text{if } t \text{ is even,} \\ (0,1) & \text{if } t \text{ is odd.} \end{cases}
\end{aligned}$$

Then the total cost vector is

$$v_{1:t} = \begin{cases} (i+\frac{1}{3}, i-\frac{1}{3}) & \text{if } t = 2i, \\ (i+\frac{1}{3}, i+\frac{2}{3}) & \text{if } t = 2i+1. \end{cases}$$

Therefore, Follow the Leader picks action $a_{t+1} = (0,1)$ if t is even, and $a_{t+1} = (1,0)$ if t is odd. In both cases, we see that $c_{t+1}(a_{t+1}) = 1$. So the total cost for the algorithm is T, whereas any fixed action achieves total cost at most $1 + T/2$, so regret is, essentially, $T/2$.

The problem is synchronization: an oblivious adversary can force the algorithm to behave in a particular way, and synchronize its costs with algorithm's actions. In fact, similar issue arises for any deterministic algorithm (see Theorem 2.1 in Lecture 7, part II).

Perturb the history! Let us turn to randomized algorithms: let us use randomization to remove the synchronization that turned out so deadly in the example discussed above.

Rather than handing the oracle information provided solely by our adversary ($v_{1:t-1}$), we *perturb* the history. Namely, we pretend there was a 0-th round, with cost vector $v_0 \in \mathbb{R}^d$ sampled from some distribution $\mathcal{D}$. We then give the oracle the "perturbed history", as expressed by the total cost vector $v_{0:t-1}$. This modified algorithm is known as *Follow the Perturbed Leader* (FPL).

Several choices for distribution $\mathcal{D}$ lead to meaningful analyses. For ease of exposition, we posit that each each coordinate of v_0 is sampled independently and uniformly from the interval $[-\frac{1}{\epsilon}, \frac{1}{\epsilon}]$, where ϵ is a parameter to be tuned according to T, U, and d.

3 Analysis of FPL

As a tool to analyze FPL, we consider a closely related algorithm called *Be the Perturbed Leader* (BPL). Imagine that when we need to choose an action at time t, we already know the cost vector v_t, and in each round t we choose $a_t = M(v_{0:t})$. Note that BPL is *not* an algorithm for the best-expert problem (because it uses v_t to choose a_t).

The analysis proceeds in two steps. We first show that BPL comes "close" to the optimal cost

$$\texttt{OPT} = \min_{a \in \mathcal{A}} \texttt{cost}(a) = v_{1:t} \cdot M(v_{1:t}),$$

and then we show that FPL comes "close" to BPL. Specifically, we will prove:

Lemma 3.1. *For each value of parameter* $\epsilon > 0$,
(i) $\texttt{cost(BPL)} \le \texttt{OPT} + \frac{d}{\epsilon}$
(ii) $\mathbb{E}[\texttt{cost(FPL)}] \le \mathbb{E}[\texttt{cost(BPL)}] + \epsilon \cdot U^2 \cdot T$

Then choosing $\epsilon = \frac{\sqrt{d}}{U\sqrt{T}}$ gives Theorem 1.1. Curiously, note that part (i) makes a statement about realized costs, rather than expected costs.

Step I: BPL comes close to OPT. By definition of the oracle M,

$$v \cdot M(v) \leq v \cdot a \quad \text{for any cost vector } v \text{ and feasible action } a. \tag{1}$$

The main argument proceeds as follows:

$$\begin{aligned}
\texttt{cost(BPL)} + v_0 \cdot M(v_0) &= \sum_{t=0}^{T} v_t \cdot M(v_{0:T}) \qquad \text{(by definition of BPL)} \\
&\leq v_{0:T} \cdot M(v_{0:T}) \qquad \text{(see Claim 3.2 below)} \qquad (2) \\
&\leq v_{0:T} \cdot M(v_{1:T}) \qquad \text{(by (1) with } a = M(v_{1:T})) \\
&= v_0 \cdot M(v_{1:T}) + \underbrace{v_{1:T} \cdot M(v_{1:T})}_{\texttt{OPT}}.
\end{aligned}$$

Subtracting $v_0 \cdot M(v_0)$ from both sides, we obtain Lemma 3.1(i):

$$\texttt{cost(BPL)} - \texttt{OPT} \leq \underbrace{v_0}_{\in[-\frac{1}{\epsilon},-\frac{1}{\epsilon}]^d} \cdot \underbrace{[M(v_{1:T}) - M(v_0)]}_{\in[-1,1]^d} \leq \frac{d}{\epsilon}.$$

The missing step (2) follows from the following claim, with $i = 0$ and $j = T$.

Claim 3.2. *For all rounds $i < j$, $\sum_{t=1}^{j} v_t \cdot M(v_{i:t}) \leq v_{i:j} \cdot M(v_{i:j})$.*

Proof. The proof is by induction on $j - i$. The claim is trivially satisfied for the base case $i = j$. For the inductive step:

$$\begin{aligned}
\sum_{t=i}^{j-1} v_t \cdot M(v_{i:t}) &\leq v_{i:j-1} \cdot M(v_{i:j-1}) \qquad \text{(by inductive hypothesis)} \\
&\leq v_{i:j-1} \cdot M(v_{i:j}) \qquad \text{(by (1) with } a = M(v_{i:j})).
\end{aligned}$$

Add $v_j \cdot M(v_{i:j})$ to both sides to complete the proof. □

Step II: FPL comes close to BPL. We compare the expected costs of FPL and BPL round per round. Specifically, we prove that

$$\mathbb{E}[\underbrace{v_t \cdot M(v_{0:t-1})}_{c_t(a_t) \text{ for FPL}}] \leq \mathbb{E}[\underbrace{v_t \cdot M(v_{0:t})}_{c_t(a_t) \text{ for BPL}}] + \epsilon U^2. \tag{3}$$

Summing up over all T rounds gives Lemma 3.1(ii).

It turns out that for proving (3) much of the structure is irrelevant. Specifically, one can denote $f(u) = v_t \cdot M(u)$ and $v = v_{1:t-1}$, and, essentially, prove (3) for arbitrary $f()$ and v.

Claim 3.3. *For any vectors $v \in R^d$ and $v_t \in [0, U/d]^d$, and any function $f : \mathbb{R}^d \to [0, R]$,*

$$\left| \mathop{\mathbb{E}}_{v_0 \sim \mathcal{D}} [f(v_0 + v) - f(v_0 + v + v_t)] \right| \leq \epsilon U R.$$

Informally, this claim states that adding v_t to $v_0 + v$ imparts a small change the output of the function f. This precisely captures why including noise in our cost vector is beneficial.

The proof of Claim 3.3 involves *coupling*, a general technique from probability theory. As a typical example for what this technique does, let X and Y be random variables drawn from some distributions P_X and P_Y, resp., and suppose we wish to analyze $\mathbb{E}[f(X) + f(Y)]$. Then

$$\mathbb{E}[f(X) + f(Y)] = \mathbb{E}[f(X') + f(Y')]$$

for any jointly distributed random variables X', Y' that are "consistent" with X, Y in the sense that the marginal distribution of X' and Y' coincide with P_X and P_Y. Therefore, instead of analyzing X, Y directly, it is sometimes easier to consider X', Y' for an appropriately constructed joint distribution.

Applied to Claim 3.3, this technique works as follows. We take $X = v_0 + v$, $Y = v_0 + v + v'$. We construct jointly distributed random variables X', Y' such that their respective marginals coinside with X and Y, and moreover it holds that $\Pr[X' = Y'] \geq 1 - \epsilon U$. The claim follows trivially from the existence of such X', Y'. Constructing the desired X', Y' is a little tedious, we omit it for now.

References