5 Full Feedback and Adversarial Costs

We express the outcomes as costs rather than rewards, and we tend to minimize totoal cost.

Problem protocol: Bandits with full feedback, adversarial costs

---

In each round $t \in [T]$:

1. Adversary chooses costs $c_t(a) \geq 0$ for each arm $a \in [K]$.
2. Algorithm picks arm $a_t \in [K]$.
3. Algorithm incurs cost $c_t(a_t)$ for the chosen arm.
4. The costs of all arms, $c_t(a) : a \in [K]$, are revealed.

Problem protocol: Sequential prediction with expert advice

For each round $t \in [T]$:

1. Observation x_t arrives.
2. K experts predict labels $z_{1,t}, \ldots, z_{K,t}$.
3. Algorithm picks expert $e \in [K]$.
4. Correct label z_t^* is revealed, along with the costs $c(z_{j,t}, z_t^*)$, $j \in [K]$ for all submitted predictions.
5. Algorithm incurs cost $c_t = c(z_{e,t}, z_t^*)$.

We will talk about arms, actions and experts interchangeably throughout this chapter.

5.1 Adversaries and regret

A crucial distinction is whether the costs depend on the algorithm's choices. An adversary is called *oblivious* if they don't, and *adaptive* if they do.

The total cost of each arm a is defined as $\mathrm{cost}(a) = \sum_{t=1}^{T} c_t(a)$

Deterministic oblivious adversary. W.l.o.g., the entire "cost table" $(c_t(a) : a \in [K], t \in [T])$ is chosen before round 1. The best arm is naturally defined as argmin $_{a\in[K]}$ $\text{cost}(a)$, and regret is defined as $R(T) = \text{cost}(\text{ALG}) - \min_{a\in[K]} \text{cost}(a)$, where cost(ALG) denotes the total cost incurred by the algorithm.

Randomized oblivious adversary. The adversary fixes a distribution $\mathcal{D}$ over the cost tables before round 1, and then draws a cost table from this distribution. Then IID costs are indeed a simple special case. since $\text{cost}(a)$ is now a random variable whose distribution is specified by $\mathcal{D}$ there are two natural ways to define the "best arm" that are different from one another:

1. $\text{argmin}_a \text{cost}(a)$: this is the best arm in hindsight, i.e., after all costs have been observed. It is a natural notion if we start from
   the deterministic oblivious adversary. Regret is defined as
   $R(T) = \text{cost}(\text{ALG}) - \min_{a\in[K]} \text{cost}(a)$ (*regret*)
2. $\text{argmin}_a \mathbb{E}[\text{cost}(a)]$: this is be best arm in foresight, i.e., an arm you'd pick if you only know the distribution $\mathcal{D}$. This is a natural notion if we start from IID costs. Regret is defined as $R(T) = \text{cost}(\text{ALG}) - \min_{a\in[K]} \mathbb{E}[\text{cost}(a)]$ (*pseudo-regret*)

Adaptive adversary typically models scenarios when algorithm's actions may alter the environment that the algorithm operates in.

Considering the best-observed arm: the best-in-hindsight arm according
to the costs actually observed by the algorithm.

5.2 Initial results: binary prediction with experts advice

binary prediction with experts advice: Expert answers can have only two values: yes or no.

Let us assume that there exists a perfect expert who never makes a mistake. Consider a simple algorithm (*majority vote algorithm*) that disregards all experts who made a mistake in the past, and follows the majority of the remaining experts:

In each round t, pick the action chosen by the majority of the experts who did not err in the past.

Theorem 5.1. Consider binary prediction with experts advice. Assuming a perfect expert, the majority vote algorithm makes at most $\log_2 K$ mistakes, where K is the number of experts.

Proof. Let S_t be the set of experts who make no mistakes up to round t, and let $W_t = |S_t|$. Note that $W_1 = K$, and $W_t \geq 1$ for all rounds t because the perfect expert is always in S_t. If the algorithm makes a mistake at round t, then $W_{t+1} \leq W_t/2$ because the majority of experts in S_t is wrong and thus excluded from S_{t+1}. It follows that the algorithm cannot make more than $\log_2 K$ mistakes.

Theorem 5.2. Consider binary prediction with experts advice. For any algorithm, any T and any K, there is a problem instance with a perfect expert such that the algorithm makes at least $\Omega(\min(T, \log K))$ mistakes.

Let us turn to the more realistic case where there is no perfect expert among the committee. majority vote不再适用，因为每个专家都可能出错，最终会把所有专家都删掉

We assign a confidence weight $w_a \geq 0$ to each expert a

Whenever an expert makes a mistake, multiply the weight of that expert by a factor $1-\epsilon$ for some fixed parameter $\epsilon > 0$. Choose a prediction with a largest total weight.

Algorithm 5.1: Weighted Majority Algorithm

parameter : $\epsilon \in [0, 1]$

Initialize the weights $w_i = 1$ for all experts.
For each round t:
  Make predictions using weighted majority vote based on w.
  For each expert i:
    If the i-th expert's prediction is correct, w_i stays the same.
    Otherwise, $w_i \leftarrow w_i(1-\epsilon)$.

Theorem 5.4. The number of mistakes made by WMA with parameter $\epsilon \in (0, 1)$ is at most $\frac{2}{1-\epsilon} \cdot \mathrm{cost}^* + \frac{2}{\epsilon} \cdot \ln K$. ($\mathrm{cost}^*$ is the times of making a wrong prediction of the best expert)

5.3 Hedge Algorithm

Deterministic algorithms are not sufficient for this goal, because they can be easily “fooled” by an oblivious adversary:

Theorem 5.5. Consider online learning with K experts and $0-1$ costs. Any deterministic algorithm has total cost T for some deterministic oblivious adversary, even if cost* $\leq T/K$.

Essentially, a deterministic- oblivious adversary just knows what the algorithm is going to do, and can rig the prices accordingly.

(deterministic algorithm不是说每次选择的都是同一个专家，而是算法选择的专家由observe到的cost完全确定且不带有随机性。比如一个简单的算法：每次选择上一次作出正确prediction的专家。那么一开始设计price/cost table的时候，就可以设计成：t轮cost为0的专家，t+1轮的cost为1)

Algorithm 5.2: Hedge algorithm for online learning with experts

parameter: $\epsilon \in (0, \frac{1}{2})$

Initialize the weights as $w_1(a) = 1$ for each arm a.

For each round t:

Let $p_t(a) = \frac{w_t(a)}{\sum_{a'=1}^{K} w_t(a')}$.

Sample an arm a_t from distribution $p_t(\cdot)$.

Observe cost $c_t(a)$ for each arm a.

For each arm a, update its weight

$w_{t+1}(a) = w_t(a) \cdot (1-\epsilon)^{c_t(a)}$.

$w_t(a)$ Weight of expert a at time t

$p_t(a)$ probability of choosing expert a at time t

For bounded cost:

Theorem 5.7. Consider an adaptive adversary such that cost* $\leq U$ for some number U known to the algorithm. Then Hedge with parameter $\epsilon = \sqrt{\ln K/(2U)}$ satisfies $\mathbb{E}\left[\mathrm{cost(ALG)} - \mathrm{cost}^*\right] < 2\sqrt{2} \cdot \sqrt{U \ln K}$

For unbounded cost:

$G_t = \sum_a p_t(a) \cdot c_t(a)^2 = \mathbb{E}\left[c_t(a_t)^2 \mid \vec{w}_t\right]$. Here $\vec{w}_t = (w_t(a) : a \in [K])$ is the vector of weights at round t

Lemma 5.8. Assume we have $\sum_{t \in [T]} \mathbb{E}[G_t] \leq U$ for some number U known to the algorithm. Then Hedge with parameter $\epsilon = \sqrt{\ln K/(3U)}$ as regret
$\mathbb{E}[\operatorname{cost}(\mathrm{ALG}) - \operatorname{cost}^*] < 2\sqrt{3} \cdot \sqrt{U \ln K}$

Consider the upper bound here: $\mathbb{E}[c_t(a)] \leq \mu$ and $\operatorname{Var}(c_t(a)) \leq \sigma^2$ for all rounds t and all arms a (5.11)

Theorem 5.9. Consider online learning with experts, with a randomizedoblivious adversary. Assume the costs are independent across rounds. Assume upper bound (5.11) for some μ and σ known to the algorithm. Then Hedge with parameter $\epsilon = \sqrt{\ln K/\left(3T\left(\mu^2 + \sigma^2\right)\right)}$ has regret
$\mathbb{E}[\operatorname{cost}(\mathrm{ALG}) - \operatorname{cost}^*] < 2\sqrt{3} \cdot \sqrt{T\left(\mu^2 + \sigma^2\right) \ln K}$