CMSC 858G: Bandits, Experts and Games 10/17/16

Lecture 7: Full feedback and adversarial rewards (part II)

Instructor: Alex Slivkins Scribed by: Mozhi Zhang

Previously, we introduced the best-expert problem, and we proved a $O(\ln K)$ mistake bound for the majority vote algorithm when a *perfect expert* exists, i.e., there is an expert that never makes mistakes. Now let us turn to the more realistic case where there is no perfect expert among the committee.

1 Binary Prediction with Expert Advice: Weighted Majority Algorithm

We extend the majority vote algorithm with a *confidence weight*. At each round, we maintain a weight w_i for each expert i, and we choose the prediction that has the highest total weights. After observing the feedback, we decay the weights of incorrect experts with a factor of $(1-\epsilon)$. This algorithm is called *Weighted Majority Algorithm* (WMA).

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, $w_i \leftarrow w_i(1-\epsilon)$.

Algorithm 1: Weighted Majority Algorithm

To analyze the algorithm, we first introduce some notations.

- Let $\texttt{cost}(a)$ be the total number of mistakes for expert a.
- Let a^* be the best expert: an expert a with the smallest $\texttt{cost}(a)$. Let $\texttt{cost}^* = \texttt{cost}(a^*)$.
- Let $W_t = \sum_{a=1}^{K} w_t(a)$, where $w_t(a)$ is the weight of expert a before round t.
- Let S_t be the set of experts that made incorrect prediction at round t.

We will also use the following fact about logarithm:

Fact 1.1. *For any $x \in (0,1)$, $\ln(1-x) < -x$.*

From the algorithm, we can easily see that $W_1 = K$ and $W_{T+1} > w_t(a^*) = (1-\epsilon)^{\texttt{cost}^*}$. Therefore, we have

$$\frac{W_{T+1}}{W_1} > \frac{(1-\epsilon)^{\texttt{cost}^*}}{K}. \tag{1}$$

Since the weights are non-increasing, we must have

$$W_{t+1} \le W_t \tag{2}$$

If the algorithm makes mistake at round t, then

$$\begin{aligned} W_{t+1} &= \sum_{a=1}^{K} w_{t+1}(a) \\ &= \sum_{a \in S_t} (1-\epsilon) w_t(a) + \sum_{a \notin S_t} w_t(a) \\ &= W_t - \epsilon \sum_{a \in S_t} w_t(a). \end{aligned}$$

Since we are using weighted majority vote, the incorrect prediction must have the majority vote:

$$\sum_{a \in S_t} w_t(a) \ge \tfrac{1}{2} W_t.$$

Therefore, if the algorithm makes mistake at round t, we have

$$W_{t+1} \le (1 - \tfrac{\epsilon}{2}) W_t.$$

Combining with (1) and (2), we get

$$\frac{(1-\epsilon)^{\texttt{cost}^*}}{K} < \frac{W_{T+1}}{W_1} = \prod_{t=1}^{T} \frac{W_{t+1}}{W_t} \le (1 - \tfrac{\epsilon}{2})^M,$$

where M is the number of mistakes. Taking logarithm of both sides, we get

$$\texttt{cost}^* \cdot \ln(1-\epsilon) - \ln K < M \cdot \ln(1 - \tfrac{\epsilon}{2}) < M \cdot (-\tfrac{\epsilon}{2}),$$

where the last inequality follows from Fact 1.1. Rearranging the terms, we get

$$M < \texttt{cost}^* \cdot \tfrac{2}{\epsilon} \ln(\tfrac{1}{1-\epsilon}) + \tfrac{2}{\epsilon} \ln K < \tfrac{2}{1-\epsilon} \cdot \texttt{cost}^* + \tfrac{2}{\epsilon} \cdot \ln K.$$

Where the last step follows from Fact 1.1 with $x = \frac{\epsilon}{1-\epsilon}$. To summarize, we have proved the following theorem.

Theorem 1.2. *The number of mistakes made by WMA with parameter $\epsilon \in (0,1)$ is at most*

$$\frac{2}{1-\epsilon} \cdot \texttt{cost}^* + \frac{2}{\epsilon} \cdot \ln K$$

Remark 1.3. This bound is non-trivial if $\texttt{cost}^*$ is small, but it is not strong enough to give us sublinear regret.

2 Hedge Algorithm for Best-Experts Problem

Deterministic algorithms are not ideal, because they can be easily "fooled" by an adversary, as demonstrated by the following theorem.

Notation. Let $\texttt{cost}(a) = \sum_{t=1}^{T} c_i(a)$ be the total cost expert a. Let a^* be the best expert: an expert a with the smallest $\texttt{cost}(a)$. Let $\texttt{cost}^* = \texttt{cost}(a^*)$. Let K be the number of arms and T be the time horizon; denote the set of arms as $[K] = \{1, 2, \ldots, K\}$.

Theorem 2.1. *Consider the best-expert problem with two arms and 0-1 costs. Any deterministic algorithm ALG has regret at least $T/2$ for some deterministic-oblivious adversary.*

Proof. Consider the following family of problem instances: at each round t, exactly one arm has cost 0, and the other arm cost 1 (denote it b_t). Note that $\texttt{cost}(1) + \texttt{cost}(2) = T$, so $\texttt{cost}^* \leq T/2$.

Let us construct the problem instance by induction on t so as to ensure that $a_t \neq b_t$. For round 1, ALG picks some fixed arm a_1, so we define $b_1 = a_1$. Suppose we have defined $b_1, \ldots, b_{t-1}$ for some t. Since ALG is deterministic, it picks some fixed arm a_t in round t; define $b_t = a_t$. Thus, we have constructed a problem instance with $\texttt{cost}(ALG) = T$, which implies regret at least $T/2$. □

We define a randomized algorithm for the best-expert problem, called Hedge. Hedge maintains a weight $w_t(a)$ for each arm (expert) a, with the same update rule as in WMA (generalized beyond 0-1 costs in an obvious way). At each round, the algorithm chooses an arm with probability proportional to the weights. The complete algorithm is shown below:

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

Algorithm 2: Hedge

let us analyze the regret of Hedge. Following the last section, we will use $W_t = \sum_{a=1}^{K} w_t(a)$ for the total weight of all arms at round t.

Step 1: easy observations. The weight of each arm after the last round is

$$w_{T+1}(a) = w_1(a) \prod_{t=1}^{T} (1-\epsilon)^{c_t(a)} = (1-\epsilon)^{\texttt{cost}(a)}.$$

Hence, the total weight of last round satisfies

$$W_{T+1} > w_{T+1}(a^*) = (1-\epsilon)^{\texttt{cost}^*}. \tag{3}$$

From the algorithm, we know that the total initial weight is $W_1 = K$.

Step 2: multiplicative decrease in W_t. We will use the following fact about exponents:

Fact 2.2. *Fix $\epsilon \in (0,1)$. There exist $\alpha, \beta \geq 0$ such that*

$$(1-\epsilon)^x < 1 - \alpha x + \beta x^2 \quad \textit{for all } x > 0. \tag{4}$$

In particular, this holds for (i) $(\alpha,\beta) = (\epsilon, 0)$ and (ii) $\alpha = \ln(\frac{1}{1-\epsilon})$ and $\beta = \alpha^2$.

Using this fact with $x = c_t(a)$, we will get the following:

$$\begin{aligned}
\frac{W_{t+1}}{W_t} &= \sum_{a\in[K]} (1-\epsilon)^{c_t(a)} \cdot \frac{w_t(a)}{W_t} \\
&< \sum_{a\in[K]} (1 - \alpha c_t(a) + \beta c_t(a)^2) \cdot p_t(a) \\
&= \sum_{a\in[K]} p_t(a) - \alpha \sum_{a\in[K]} p_t(a) c_t(a) + \beta \sum_{a\in[K]} p_t(a) c_t(a)^2 \\
&= 1 - \alpha F_t + \beta G_t, \qquad (5)
\end{aligned}$$

where

$$\begin{aligned}
F_t &= \sum_a p_t(a) \cdot c_t(a) = \mathbb{E}\left[c_t(a_t) \mid \vec{w}_t\right] \\
G_t &= \sum_a p_t(a) \cdot c_t(a)^2 = \mathbb{E}\left[c_t(a_t)^2 \mid \vec{w}_t\right].
\end{aligned}$$

Here $\vec{w}_t = (w_t(a) : a \in [K])$ is the vector of weights at round t. Notice that the total expected cost of the algorithm is $\mathbb{E}[\texttt{cost}(ALG)] = \sum_t \mathbb{E}[F_t]$.

Step 3: a naive attempt. Using the (5), we can obtain:

$$\frac{(1-\epsilon)^{\texttt{cost}^*}}{K} \leq \frac{W_{T+1}}{W_1} = \prod_{t=1}^{T} \frac{W_{t+1}}{W_t} < \prod_{t=1}^{T} (1 - \alpha F_t + \beta G_t).$$

However, it is not clear how to connect the right-hand side to $\sum_t F_t$ so as to argue about the total cost of the algorithm.

Step 4: the telescoping product. Taking a logarithm on both sides of (5) and using Fact (1.1), we get

$$\ln \frac{W_{t+1}}{W_t} < \ln(1 - \alpha F_t + \beta G_t) < -\alpha F_t + \beta G_t.$$

Inversing the signs and summing over t on both sides, we get

$$\begin{aligned}
\sum_{t=1}^{T} (\alpha F_t - \beta G_t) &< - \sum_{t=1}^{T} \ln \frac{W_{t+1}}{W_t} \\
&= - \ln \prod_{t=1}^{T} \frac{W_{t+1}}{W_t} \\
&= - \ln \frac{W_{T+1}}{W_1} \\
&= \ln W_1 - \ln W_{T+1} \\
&< \ln K - \ln(1-\epsilon) \cdot \text{cost}^*,
\end{aligned}$$

where we used (3) in the last step. Taking expectation on both sides, we obtain:

Lemma 2.3. *For α, β from Fact 2.2, we have:*

$$\alpha\,\mathbb{E}[\texttt{cost}(ALG)] < \beta \sum_{t=1}^{T} \mathbb{E}[G_t] + \ln K - \ln(1-\epsilon)\,\mathbb{E}[\texttt{cost}^*].$$

In what follows, we use this lemma in two different ways.

Step 5: main theorem. Using Lemma Lemma 2.3 with $\alpha = \epsilon$ and $\beta = 0$, we obtain:

$$\mathbb{E}[\texttt{cost}(ALG)] < \tfrac{\ln K}{\epsilon} + \underbrace{\tfrac{1}{\epsilon}\ln(\tfrac{1}{1-\epsilon})}_{\leq 1+2\epsilon \text{ if } \epsilon \in (0,\frac{1}{2})} \mathbb{E}[\texttt{cost}^*].$$

$$\mathbb{E}[\texttt{cost}(ALG) - \texttt{cost}^*] < \frac{\ln K}{\epsilon} + 2\epsilon\ \mathbb{E}[\texttt{cost}^*].$$

This yields the following theorem:

Theorem 2.4. *Assume we have* $\texttt{cost}^* \leq U$ *for some number* $U \leq T$ *known to the algorithm. Then the regret of Hedge with parameter* $\epsilon = \sqrt{\ln K/(2U)}$ *is*

$$\mathbb{E}[\texttt{cost}(ALG) - \texttt{cost}^*] < 2\sqrt{2} \cdot \sqrt{U \ln K}.$$

Step 6: unbounded costs. Next, we consider the case where the costs can be unbounded, but we have an upper bound on $\mathbb{E}[G_t]$. We use Lemma 2.3 with $\alpha = \ln(\frac{1}{1-\epsilon})$ and $\beta = \alpha^2$ to obtain:

$$\alpha\,\mathbb{E}[\texttt{cost}(ALG)] < \alpha^2 \sum_{t=1}^{T} \mathbb{E}[G_t] + \ln K + \alpha\,\mathbb{E}[\texttt{cost}^*].$$

Dividing both sides by α and moving terms around, we get

$$\mathbb{E}[\texttt{cost}(ALG) - \texttt{cost}^*] < \frac{\ln K}{\alpha} + \alpha \sum_{t=1}^{T} \mathbb{E}[G_t] < \frac{\ln K}{\epsilon} + 3\epsilon \sum_{t=1}^{T} \mathbb{E}[G_t],$$

where the last step uses the fact that $\epsilon < \alpha < 3\epsilon$ for $\epsilon \in (0, \frac{1}{2})$. Thus:

Lemma 2.5. *Assume we have* $\sum_{t=1}^{T} \mathbb{E}[G_t] \leq U$ *for some number* U *known to the algorithm. Then Hedge with parameter* $\epsilon = \sqrt{\ln K/(3U)}$ *as regret*

$$\mathbb{E}[\texttt{cost}(ALG) - \texttt{cost}^*] < 2\sqrt{3} \cdot \sqrt{U \ln K}.$$

Step 7: unbounded costs with small expectation and variance. We can use Lemma 2.5 to derive a regret bound when costs may be unbounded, but their expectation and variance are small. Specifically, consider an *randomized-oblivious* adversary, and assume that the costs are independent across rounds. Further, assume an upper bound:

$$\mathbb{E}[c_t(a)] \leq \mu \text{ and } \mathrm{Var}(c_t(a)) \leq \sigma^2 \text{ for all rounds } t \text{ and all arms } a. \tag{6}$$

Then for each round t we have:

$$\mathbb{E}[c_t(a)^2] = \mathrm{Var}(c_t(a)) + \mathbb{E}[c_t(a)]^2 \leq \sigma^2 + \mu^2.$$

$$\mathbb{E}[G_t] = \sum_{a \in [K]} p_t(a)\,\mathbb{E}[c_t(a)^2] \leq \sum_{a \in [K]} p_t(a)(\mu^2 + \sigma^2) = \mu^2 + \sigma^2.$$

Summing this upper bound over all rounds t, and plugging it into Lemma 2.5, we obtain:

Theorem 2.6. *Consider the best-experts problem with a randomized-oblivious adversary. Assume the costs are independent across rounds. Assume upper bound (6) for some μ and σ known to the algorithm. Then Hedge with parameter $\epsilon = \sqrt{\ln K/(3T(\mu^2+\sigma^2))}$ has regret*

$$\mathbb{E}[\texttt{cost}(ALG) - \texttt{cost}^*] < 2\sqrt{3} \cdot \sqrt{T(\mu^2+\sigma^2)\ln K}.$$

Remark 2.7. In the next lecture, we will use Lemma 2.5 to analyze a bandit algorithm.

3 Bibliographic remarks

This material is presented in various courses and books on online learning, e.g. Cesa-Bianchi and Lugosi (2006) and Hazan (2015). This lecture follows a lecture plan from (Kleinberg, Spring 2007), but presents the analysis of Hedge a little differently (so as to make it immediately applicable to the analysis of EXP3/EXP4 in the next lecture).

References

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games.* Cambridge Univ. Press, 2006.

Elad Hazan. Introduction to Online Convex Optimization. *Foundations and Trends® in Optimization*, 2 (3-4):157–325, 2015.

Robert Kleinberg. Lecture notes: *CS683: Learning, Games, and Electronic Markets* (week 1), Spring 2007. Available at `http://www.cs.cornell.edu/courses/cs683/2007sp/lecnotes/week1.pdf`.