6.1220 LECTURE 18

BRENDON JIANG

Contents

1. Multiplicative Weights

1.1. Expert Problem

Example 1.1: Motivating example

You just moved to Boston, you go to work by car, but go for a walk outside at lunchtime every day. Every morning you ponder whether to take with you an umbrella or sunglasses for your lunch walk. You have no insight to Boston's weather. However, you do have access n weather forecasts (experts) each making a binary rain/sun prediction. Every morning, for the next 365 days, you will decide what to take based on your past experience. Is there a good algorithm for making your decision every day?

If you make the wrong decision you incur a loss of 1; otherwise you incur a loss of 0. Given how unpredictable Boston weather is you can make no modeling assumption about the weather or the forecasters. We want to analyze worst-case, each day the realized weather and the expert forecasts are picked by an adversary with full knowledge of your algorithm.

Solution 1.2: Example algorithm

Suppose in the case you have 2 experts and you pick the historically best expert, breaking ties in favor of expert 2. This has a bad worst case, as they can know which expert you pick each time and make the weather different from the decision of that expert. The total worst case loss is T, the number of days.

Note that there is no way to guarantee loss that approximates that of the prophet's. The adversary can simply assign rain/sun uniformly at random, on each day. The prophet can always achieve a loss

Date: November 2024

of 0, while you get a loss of $T/2$ in expectation. Instead of competing with the omniscient prophet, what if we competed with the best expert? It turns out, this is indeed possible.

Definition 1.3: General Setup: Learning with Expert Advice

Every day has a binary outcome: 0 or 1 (rain or sun). For a sequence of T days, on each day $1, \ldots, T$:

- each expert $i = 1, \ldots, n$ makes a binary prediction $\pi_i^t \in \{0, 1\}$.
- based on these predictions (and history so far) you make your own prediction $d^t \in \{0, 1\}$.
- the binary event on this day is realized $b^t \in \{0, 1\}$.
- you incur loss $m^t = 1 d^t \neq b^t$

Your aggregate loss here then is $\sum_{t=1}^{T} 1_{d^t \neq b^t}$. The loss of the best expert is $\min_i \sum_{t=1}^{T} 1_{\pi_i^t \neq b^t}$. Your hindsight regret is aggregate loss subtracted by loss of best expert. You want to minimize your regret. Your adversary wants to maximize your regret.

1.2. The Weighted Majority Algorithm

Definition 1.4: The Weighted Majority Algorithm

Parameter $\epsilon \in (0, \frac{1}{2}]$. Initialize $w_i^1 = 1$ for all $i = 1, \ldots, n$. On each day $t = 1, \ldots, T$:

- Compare total weight of experts predicting 0 and total weight of experts prediction 1.
- Choose $d^t = 0, 1$ depending on which group has more weight.
- After the binary event we update the weights $w_i^{t+1} = w_i^t$ if $\pi_i^t = b^t$, otherwise multiply it by $w_i^t(1 - \epsilon)$.

Theorem 1.5

The predictions of the weighted majority algorithm satisfy

$$\sum_{t=1}^{T} 1_{d^t \neq b^t} \leq 2(1 + \epsilon) \min_i \sum_{t=1}^{T} 1_{\pi_i^t \neq b^t} + 2\frac{\log n}{\epsilon}$$

Proof

Denote $W^t = \sum_i w_i^t$. We claim two things

(1) $W^{t+1} \leq W^t$

(2) if $d^t = b^t$:

$$\begin{aligned} W^{t+1} &= \sum_i w_i^{t+1} \\ &= \sum_{i:\pi_i^t = b^t} w_i^t + (1-\epsilon) \sum_{i:\pi_i^t \neq b^t} w_i^t \\ &= \sum_i w_i^t - \epsilon \sum_{i:\pi_i^t \neq b^t} w_i^t \\ &\leq \left(1 - \frac{\epsilon}{2}\right) W^t \end{aligned}$$

Then, if m is the total number of mistakes I made, $W^T \leq \left(1-\frac{\epsilon}{2}\right)^m \cdot W^1 = \left(1-\frac{\epsilon}{2}\right)^m \cdot n$. Denote m^* as the total number of mistakes of the best expert. We then know

$$w_{i^*}^T = (1-\epsilon)^{m^*} \leq W^T \leq \left(1-\frac{\epsilon}{2}\right)^m \cdot n$$

Taking the log of both sides,

$$m^* \cdot \log(1-\epsilon) \leq m \cdot \log\left(1-\frac{\epsilon}{2}\right) + \log n$$

Using the inequality $-x - x^2 \leq \log(1-x) \leq -x \forall x \in (0, \frac{1}{2}]$, we get

$$m^*(-\epsilon - \epsilon^2) \leq -m \cdot \frac{\epsilon}{2} + \log n$$

$$m\frac{\epsilon}{2} \leq m^* \epsilon(1+\epsilon) + \log n$$

$$m \leq 2(1+\epsilon)m^* + \frac{2\log n}{\epsilon}$$

Definition 1.6: More General Setup

Still Binary outcome 0 or 1 for each day. For a sequence of T days, on each day $t = 1, \ldots, T$:

- each expert $i = 1, \ldots, n$ makes a binary prediction: $\pi_i^t \in \{0, 1\}$
- based on these predictions, you make your own probabilistic prediction p^t.
- the weather on this day is realized $b^t \in \{0, 1\}$.
- You incur a loss $m^t = \mathbb{E}_{d^t \sim p^t}[1_{d^t \neq b^t}$.

Aggregate loss, loss of best expert, hindsight regret are all defined similarly with this probabilistic prediction p^t. Our adversary can not see the sample $d^t \sim p^t$ when choosing b^t.

Definition 1.7: The Multiplicative Weights Update Algorithm

Parameter $\epsilon \in (0, \frac{1}{2}]$. Initialize $w_i^1 = 1$ for all $i = 1, \ldots, n$. On each day $t = 1, \ldots, T$:

- $d^t = \pi_i^t$ with probability $w_i^t / \sum_i w_i^t$.

- After the binary event we update the weights $w_i^{t+1} = w_i^t$ if $\pi_i^t = b^t$, otherwise multiply it by $w_i^t(1-\epsilon)$.

Theorem 1.8

The above algorithm satisfies

$$\sum_{t=1}^{T} \mathbb{E}_{d^t \sim p^t}[1_{d^t \neq b^t}] \leq (1+\epsilon) \min_i \sum_{t=1}^{T} 1_{\pi_i^t \neq b^t} + \frac{\log n}{\epsilon}$$

Proof

Like the previous proof, we track the total weight. We denote $M_i^t = 1_{\pi_i^t \neq b^t}$ and $M^t = 1_{d^t \neq b^t}$. Thus, $\mathbb{E}[M^t] = \sum_i \frac{w_i^t}{w^t} M_i^t$. This inequality shows the third step of the chain below.

$$\begin{aligned} W^{t+1} &= \sum_i w_i^{t+1} \\ &= \sum_i (1 - \epsilon M_i^t) w_i^t \\ &= \left(1 - \mathbb{E}[M^t]\right) W^t \\ &\leq e^{-\epsilon \mathbb{E}[M^t]} \cdot W^t \end{aligned}$$

We then get $(1-\epsilon)^{m^*} \leq W^t \leq e^{-\epsilon \sum_i \mathbb{E}[M^t]} n$ Doing manipulation on the equations and using easy inequalities then gets us the desired theorem.