Minimizing External Regret

Thomas Kesselheim

Preview Last Update: April 16, 2019

Last time, we got to know correlated equilibria and coarse correlated equilibria. We showed that if all players use a no-external-regret algorithm to update their strategy choices, the average history of play will converge to a coarse correlated equilibrium. The only missing piece is: How do these algorithms work?

1 Problem Statement

There is a single player playing T rounds against an adversary, trying to minimize his cost. In each round, the player chooses a probability distribution over N strategies (also termed actions here). After the player has committed to a probability distribution, or mixed strategy as we will say, the adversary picks a cost vector fixing the cost for each of the N strategies.

In round $t = 1, \ldots, T$, the following happens:

- The player picks a probability distribution $p^{(t)} = (p_1^{(t)}, \ldots, p_N^{(t)})$ over his strategies.
- The adversary picks a cost vector $\ell^{(t)} = (\ell_1^{(t)}, \ldots, \ell_N^{(t)})$, where $\ell_i^{(t)} \in [0, 1]$ for all i.
- A strategy $a^{(t)}$ is chosen according to the probability distribution $p^{(t)}$. The player incurs this strategy's cost and gets to know the entire cost vector.

What is the right benchmark for an algorithm in this setting? The *best action sequence in hindsight* achieves a cost of $\sum_{t=1}^{T} \min_{i \in [N]} \ell_i^{(t)}$. However, getting close to this number is generally hopeless as the following example shows.

Example 7.1. *Suppose $N = 2$ and consider an adversary that chooses $\ell^{(t)} = (1, 0)$ if $p_1^{(t)} \geq 1/2$ and $\ell^{(t)} = (0, 1)$ otherwise. Then the expected cost of the player is at least $T/2$, while the best action sequence in hindsight has cost 0.*

Instead, we will swap the sum and the minimum, and compare to $L_{\min}^{(T)} = \mathbf{E}\left[\min_{i \in [N]} \sum_{t=1}^{T} \ell_i^{(t)}\right]$. That is, instead of comparing to the best action sequence in hindsight, we compare to the *best fixed action in hindsight.* The expected cost of some algorithm $\mathcal{A}$ is given as $L_{\mathcal{A}}^{(T)} = \mathbf{E}\left[\sum_{t=1}^{T} \ell_{a^{(t)}}^{(t)}\right]$. The difference of this cost and the cost of the best single strategy in hindsight is called *external regret.*

Definition 7.2. *The* expected external regret *of algorithm $\mathcal{A}$ is defined as $R_{\mathcal{A}}^{(T)} = L_{\mathcal{A}}^{(T)} - L_{min}^{(T)}$.*

Definition 7.3. *An algorithm is called* no-external-regret algorithm *if for any adversary and all T we have $R_{\mathcal{A}}^{(T)} = o(T)$.*

This means that the *average* cost per round of a no-external-regret algorithm approaches the one of the best fixed strategy in hindsight or even beats it.

2 The Multiplicative-Weights Algorithm

By the definition it is not even clear that there are no-external-regret algorithms. Fortunately, there are. In this section, we will get to know the *multiplicative-weights algorithm* (also known as randomized weighted majority or hedge).

The algorithm maintains weights $w_i^{(t)}$, which are proportional to the probability that strategy i will be used in round t. After each round, the weights are updated by a multiplicative factor, which depends on the cost in the current round.

Let $\eta \in (0, \frac{1}{2}]$; we will choose η later.

- Initially, set $w_i^{(1)} = 1$, for every $i \in [N]$.
- At every time t,
  - Let $W^{(t)} = \sum_{i=1}^{N} w_i^{(t)}$;
  - Choose strategy i with probability $p_i^{(t)} = w_i^{(t)}/W^{(t)}$;
  - Set $w_i^{(t+1)} = w_i^{(t)} \cdot (1-\eta)^{\ell_i^{(t)}}$.

Let's build up some intuition for what this algorithm does. First suppose $\ell_i^{(t)} \in \{0, 1\}$. Strategies with cost 0 maintain their weight, while the weight of strategies with cost 1 is multiplied by $(1-\eta)$. So the weight decays exponentially quickly in the number of 1's. Next consider the impact of η. Setting η to zero means that we pick a strategy uniformly at random and continue to do so, on the other hand the higher η the more we punish strategies which incurred a high cost. So we can think of η as controlling the tradeoff between exploration (small η) and exploitation (large η).

Theorem 7.4 (Littlestone and Warmuth, 1994)**.** *The multiplicative-weights algorithm, for any choices by the adversary of cost vectors from* $[0,1]$*, guarantees*

$$L_{MW}^{(T)} \leq (1+\eta) L_{min}^{(T)} + \frac{\ln N}{\eta} \quad .$$

Setting $\eta = \sqrt{\frac{\ln N}{T}}$ yields

$$L_{\mathrm{MW}}^{(T)} \leq L_{\mathrm{min}}^{(T)} + 2\sqrt{T \ln N} \quad .$$

Corollary 7.5. *The multiplicative-weights algorithm with* $\eta = \sqrt{\frac{\ln N}{T}}$ *has external regret at most* $2\sqrt{T \ln N} = o(T)$ *and hence is a no-external-regret algorithm.*

3 Non-Adaptive Adversary

It seems particularly difficult to analyze the algorithm because the adversary is allowed to react to the player's choices. It will turn out that this does not actually matter. But as a first step, let us ignore entirely this adaptivity and let us assume that the adversary has to fix the sequence $\ell^{(1)}, \ldots, \ell^{(T)}$ first. Note that this immediately fixes the probability vectors $p^{(1)}, \ldots, p^{(T)}$ as well. They are not random anymore.

Proposition 7.6. *For every fixed non-adaptive sequence* $\ell^{(1)}, \ldots, \ell^{(T)}$ *of cost vectors from* $[0,1]$*, MW guarantees*

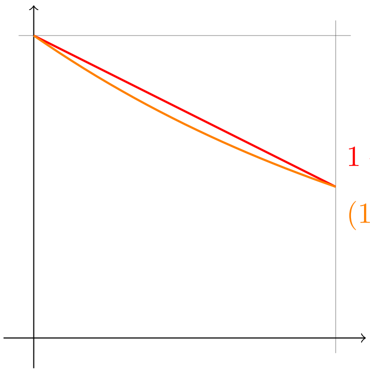
$$L_{MW}^{(T)} \leq (1+\eta) L_i^{(T)} + \frac{\ln N}{\eta} \quad ,$$

where $L_i^{(T)} = \sum_{t=1}^{T} \ell_i^{(t)}$ *is the sum of costs of strategy* i *and* $L_{MW}^{(T)} = \sum_{t=1}^{T} \sum_{i=1}^{N} p_i^{(t)} \ell_i^{(t)}$ *is the expected sum of costs of MW.*

Proof. Let us analyze how the sum of weights $W^{(t)}$ decreases over time. It holds

$$W^{(t+1)} \;=\; \sum_{i=1}^{N} w_i^{(t+1)} \;=\; \sum_{i=1}^{N} w_i^{(t)} (1-\eta)^{\ell_i^{(t)}} \quad .$$

Observe that $(1-\eta)^{\ell} = (1-\ell\eta)$, for both $\ell = 0$ and $\ell = 1$. Furthermore, $(1-\eta)^{\ell}$ is a convex function in ℓ. For $\ell \in [0,1]$ this implies $(1-\eta)^{\ell} \leq (1-\ell\eta)$.



This gives us

$$W^{(t+1)} \leq \sum_{i=1}^{N} w_i^{(t)}(1-\ell_i^{(t)}\eta) = W^{(t)} - \eta \sum_{i=1}^{N} w_i^{(t)} \ell_i^{(t)} .$$

Let $\ell_{\text{MW}}^{(t)}$ denote the expected cost of MW in step t. It holds $\ell_{\text{MW}}^{(t)} = \sum_{i=1}^{N} \ell_i^{(t)} w_i^{(t)} / W^{(t)}$. Substituting this into the bound for $W^{(t+1)}$ gives

$$W^{(t+1)} \leq W^{(t)} - \eta \ell_{\text{MW}}^{(t)} W^{(t)} = W^{(t)}(1-\eta\ell^{(t)}) .$$

As a consequence,

$$W^{(T+1)} \leq W^{(1)} \prod_{t=1}^{T} (1-\eta\ell^{(t)}) = N \prod_{t=1}^{T} (1-\eta\ell_{\text{MW}}^{(t)}) .$$

This means that the sum of weights after step T can be *upper bounded* in terms of the expected costs of MW. On the other hand, the sum of weights after step T can be *lower bounded* in terms of the costs of the best strategy as follows:

$$W^{(T+1)} \geq \max_{1 \leq i \leq N} (w_i^{(T+1)}) = \max_{1\leq i \leq N} \left(w_i^{(1)} \prod_{t=1}^{T} (1-\eta)^{\ell_i^{(t)}} \right) = \max_{1 \leq i \leq N} \left((1-\eta)^{\sum_{t=1}^{T} \ell_i^{(t)}} \right) = (1-\eta)^{L_{\min}^{(T)}} .$$

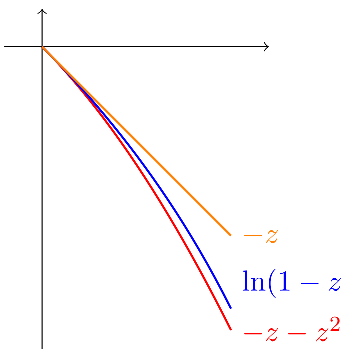
Combining the bounds and taking the logarithm on both sides gives us

$$L_{\min}^{(T)} \ln(1-\eta) \leq (\ln N) + \sum_{t=1}^{T} \ln(1-\eta\ell_{\text{MW}}^{(t)}) .$$

In order to simplify, we will now use the following estimation

$$-z - z^2 \leq \ln(1-z) \leq -z ,$$

which holds for every $z \in [0, \frac{1}{2}]$.



This gives us

$$\begin{aligned} L_{\min}^{(T)}(-\eta-\eta^2) &\leq (\ln N) + \sum_{t=1}^{T}(-\eta \ell^{(t)}) \\ &= (\ln N) - \eta L_{\mathrm{MW}}^{(T)} \ . \end{aligned}$$

Finally, solving for $L_{\mathrm{MW}}^{(T)}$ gives

$$L_{\mathrm{MW}}^{(T)} \leq (1+\eta) L_{\min}^{(T)} + \frac{\ln N}{\eta} \ . \quad \square$$

4 Adaptive Adversary

The above argument works against a non-adaptive adversary. That is, the sequence of cost vectors $\ell^{(1)}, \ldots, \ell^{(T)}$ is fixed before the player does anything. Somewhat surprisingly, the guarantee continues to hold even if the adversary can adapt to the player's decisions. Note that this way the point of comparison, the best strategy in hindsight, changes depending on the choices made by the player.

Proposition 7.7. *The multiplicative-weights algorithm, for any (possibly adaptive) choices by the adversary of cost vectors from* $[0,1]$*, guarantees*

$$L_{MW}^{(T)} \leq (1+\eta) L_{min}^{(T)} + \frac{\ln N}{\eta} \ .$$

Proof. We would like to bound $\mathbf{E}\left[\sum_{t=1}^{T} \ell_{a^{(t)}}^{(t)}\right] = \sum_{t=1}^{T} \mathbf{E}\left[\ell_{a^{(t)}}^{(t)}\right]$. The difficulty is that $\ell_{a^{(t)}}^{(t)}$ depends on all cost vectors and actions taken so far as well as the randomization in the current. However, if we keep everything fixed that happened in previous rounds, the $p^{(t)}$ vector is fixed and the probability that action j is played is $p_j^{(t)}$. Stated differently, we can write out the conditional expectation as

$$\mathbf{E}\left[\ell_{a^{(t)}}^{(t)} \,\middle|\, \ell^{(1)}, \ldots, \ell^{(t-1)}, a^{(1)}, \ldots, a^{(t-1)}\right] = \sum_{j=1}^{N} p_j^{(t)} \ell_j^{(t)} \ .$$

This is true for every conditional expectation. We can get rid of the conditioning but just taking the expectation over the conditioned random variables. So

$$\mathbf{E}\left[\ell_{a^{(t)}}^{(t)}\right] = \mathbf{E}\left[\sum_{j=1}^{N} p_j^{(t)} \ell_j^{(t)}\right] \ ,$$

and by linearity of expectation

$$\mathbf{E}\left[\sum_{t=1}^{T} \ell_{a^{(t)}}^{(t)}\right] = \sum_{t=1}^{T} \mathbf{E}\left[\ell_{a^{(t)}}^{(t)}\right] = \sum_{t=1}^{T} \mathbf{E}\left[\sum_{j=1}^{N} p_j^{(t)} \ell_j^{(t)}\right] = \mathbf{E}\left[\sum_{t=1}^{T}\sum_{j=1}^{N} p_j^{(t)} \ell_j^{(t)}\right] \ .$$

Observe that the argument of the expectation is $\sum_{t=1}^{T}\sum_{j=1}^{N} p_j^{(t)} \ell_j^{(t)}$. This is exactly the term that we bounded in Proposition 7.6. It does not even talk about the actually chosen actions $a^{(t)}$ but only about the probability vectors.

Stated differently, we use the fact that the probability vectors $p^{(t)}$ are generated in a *deterministic* way. That is, the adversary can anticipate them and therefore adaptivity does not help. $\square$

Recommended Literature

- Chapter 4 in the AGT book.
- Tim Roughgarden's lecture notes `http://theory.stanford.edu/~tim/f13/l/l17.pdf` and lecture video `https://youtu.be/ssAEgJKRe9o`
- N. Littlestone, M. Warmuth. The Weighted Majority Algorithm. Information and Computation 108(2):212–261, 1994.