7.1 Introduction

Remember that we showed last class that so-called *no-regret* algorithms lead to coarse-correlated equilibria. Slightly more formally, we proved that if every player uses an algorithm with with no-regret against adaptive adversaries, then the time-averaged distribution of play is an ϵ-approximate coarse correlated equilibrium (where ϵ is the expected regret of each player, or the max such expected regret if the players have different bounds).

One thing we didn't do was talk about how to actually construct no-regret algorithms. So we'll start off with that today. Then we'll explore other notions of regret and how they correspond to other notions of equilibria (namely, correlated equilibria).

7.2 No-Regret Algorithm: Multiplicative Weights

Since this isn't a class on learning theory I don't want to go too far into this, but I want to at least show a simple no-regret algorithm. This is a classical and famous algorithm which goes under a few different names, depending on the precise variant and community: nowadays it is usually called *Multiplicative Weights*, but it's also known as *Randomized Weighted Majority* and as *Hedge*. Now we're back in the online learning setting, so there's just one player and there's an action set A.

This algorithm is going to make three important assumptions:

1. We're in the "experts" setting, not the "bandit" setting, so after we choose an action a^t we find out the full cost vector c^t (not just $c^t(a^t)$)
2. We know T (the maximum time, sometimes called the "time horizon") at the beginning.
3. The adversary is oblivious.

All of these assumptions can be removed, i.e., we can modify multiplicative weights to work in the bandit setting, without knowing T, and against an adaptive adversary. These modifications are not incredibly difficult, but they're annoying enough that I'll defer them to the reading and just focus on the basic idea in this simple setting.

- Initialize $w^1(a) = 1$ for all $a \in A$.
- For $t = 1, 2, \ldots, T$:
  - Play an action according to the distribution

$$p^t = \frac{w^t}{\sum_{a \in A} w^t(a)},$$

    i.e., interpret the weights as probabilities.

– Given the cost vector c^t, update the weights by setting

$$w^{t+1}(a) = w^t(a) \cdot (1-\epsilon)^{c^t(a)},$$

for all $a \in A$.

We'll instantiate ϵ later but for now just think of it as something small. But note that if ϵ is small then we're still maintaining an "approximately uniform" distribution over the action, which means that we're "exploring" the space. If ϵ is big then we're focusing on actions which have been good in the past: we're "exploiting". That's one reason that these problems are sometimes called "exploration vs. exploitation". What we're going to show is that there's an ϵ which allows us to basically do both, simultaneously.

7.2.1 Analysis

We're going to assume that the adversary is oblivious (as mentioned). for every time t, let $\Gamma^t = \sum_{a \in A} w^t(a)$ be the total weight. Suppose that a^* is the best fixed action in hindsight, i.e.,

$$a^* = \arg\min_{a \in A} \sum_{t=1}^{T} c^t(a) = \sum_{t=1}^{T} c^t(a^*).$$

Let $OPT = \sum_{t=1}^{T} c^t(a^*)$. Then

$$\Gamma^T \geq w^T(a^*) = w^1(a^*) \prod_{t=1}^{T} (1-\epsilon)^{c^t(a^*)} = 1 \cdot (1-\epsilon)^{\sum_{t=1}^{T} c^t(a^*)} = (1-\epsilon)^{OPT}.$$

The *expected* cost of the algorithm at time t is

$$v^t = \sum_{a \in A} \frac{w^t(a)}{\Gamma^t} \cdot c^t(a)$$

So we're trying to bound the total expected cost of the algorithm $\sum_{t=1}^{T} v^t$ in terms of OPT.

Now let's analyze Γ^t a little bit, and see how it evolves over time.

$$\begin{aligned}
\Gamma^{t+1} &= \sum_{a \in A} w^{t+1}(a) = \sum_{a \in A} w^t(a) \cdot (1-\epsilon)^{c^t(a)} \\
&\leq \sum_{a \in A} w^t(a)(1-\epsilon c^t(a)) && \text{(since } (1-\epsilon)^x \leq 1-\epsilon x \text{ for } \epsilon \in [0,1/2] \text{ and } x \in [0,1]) \\
&= \Gamma^t - \sum_{a \in A} \epsilon w^t(a) c^t(a) = \Gamma^t - \epsilon \Gamma^t v^t \\
&= \Gamma^t (1 - \epsilon v^t)
\end{aligned}$$

Now we use our upper and lower bound on Γ^T to get that

$$(1-\epsilon)^{OPT} \leq \Gamma^T \leq \Gamma^1 \prod_{t=1}^{T} (1-\epsilon v^t) = n \prod_{t=1}^{T} (1-\epsilon v^t).$$

If we take natural logs of both sides, we get that

$$OPT \cdot \ln(1-\epsilon) \leq \ln n + \sum_{t=1}^{T} \ln(1-\epsilon v^t)$$

If we compute the Taylor expansion of $\ln(1-x)$ at $x = 0$, we get that $\ln(1-x) = -x - (x^2/2) - (x^3/3) - \ldots$. Thus if $x \leq 1/2$, this implies that $-x - x^2 \leq \ln(1-x) \leq -x$. Applying this to the above inequality, we get that

$$\begin{aligned} & OPT \cdot (-\epsilon - \epsilon^2) \leq \ln n + \sum_{t=1}^{T} (-\epsilon v^t) \\ \implies & \sum_{t=1}^{T} \epsilon v^t \leq OPT(\epsilon + \epsilon^2) + \ln n \\ \implies & \sum_{t=1}^{T} v^t \leq OPT(1+\epsilon) + \frac{1}{\epsilon} \ln n \leq OPT + \epsilon T + \frac{1}{\epsilon} \ln n. \end{aligned}$$

Now we'll finally set ϵ to $\sqrt{\frac{\ln n}{T}}$ (note that this is where we assume we know T). This means that the expected regret of the algorithm is

$$\frac{1}{T}\left(\sum_{t=1}^{T} v^t - OPT\right) \leq \frac{1}{T}\left(\epsilon T + \frac{1}{\epsilon}\ln n\right) = \frac{1}{T}\left(\sqrt{T \ln n} + \sqrt{T \ln n}\right) \leq 2\sqrt{\frac{\ln n}{T}}$$

This regret goes to 0 at T goes to ∞, so this proves that multiplicative weights is a no-regret algorithm!

7.3 Correlated Equilibria and Swap Regret

So now we know that no-regret algorithms exist, are relatively simple, and if every player uses one then the time-averaged distribution of play becomes an approximate coarse correlated equilibrium. This leads to an obvious question: is there some other type of learning algorithm which will lead to correlated equilibria? We know that no such algorithm can exist for mixed Nash (due to PPAD-hardness), but that maybe slightly "smarter" learning algorithms will lead to a "stronger" notion of equilibrium than coarse correlated? This turns out to be true, and is what we'll focus on next.

7.3.1 Correlated Equilibria: New Definition

It will be useful for us to use a slightly different but equivalent definition of correlated equilibria. First, let's recall the definition we've been using.

Definition 7.3.1 *A distribution σ over S is a* correlated equilibrium *if*

$$\mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s)|s_i] \leq \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s_{-i}, s_i')|s_i]$$

for all $i \in [k]$ and $s_i, s_i' \in S_i$.

In other words, for every player i, that player has no incentive to switch to some strategy s_i' even if it is told that in the strategy profile the trusted third party drew (from the distribution σ) it is supposed to be playing strategy s_i. So player i does not want to *switch* from s_i to s_i' when told to play s_i.

This way of thinking about it gives a different definition, in terms of "switching functions". A switching function is just a function $\delta : S_i \to S_i$, which we think of as telling to "switch" from some action s_i to $\delta(s_i)$. Let's prove that we can also define correlated equilibria in terms of switching functions. This will be simpler since it removes the weird "conditioning", but harder since now we have functions to quantify over.

Theorem 7.3.2 *σ is a correlated equilibrium if and only if for all $i \in [k]$ and $\delta : S_i \to S_i$,*

$$\mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s)] \leq \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s_{-i}, \delta(s_i)]. \tag{7.3.1}$$

Proof: Let's start with the "only if" direction: showing that if σ is a correlated equilibrium then it satisfies (7.3.1) for all i, δ. So let σ be a correlated equilibrium, let $i \in [k]$, and let $\delta : S_i \to S_i$. Then

$$\begin{aligned}
\mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s)] &= \sum_{a\in S_i} \left(\mathbf{Pr}[s_i = a] \cdot \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s)|s_i = a] \right) \\
&\leq \sum_{a\in S_i} \left(\mathbf{Pr}[s_i = a] \cdot \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s_{-i}, \delta(a))|s_i = a] \right) && \text{(def of CE)} \\
&= \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s_{-i}, \delta(s_i))],
\end{aligned}$$

as required.

Let's prove the "if" direction now. Suppose that $\mathbf{E}_{s\sim\sigma}[C_i(s)] \leq \mathbf{E}_{s\sim\sigma}[C_i(s_{-i}, \delta(s_i)]$ for all $i \in [k]$ and $\delta : S_i \to S_i$. Let $i \in [k]$ and let $a, b \in S_i$, and define the switching function

$$\delta(x) = \begin{cases} b & \text{if } x = a \\ x & \text{otherwise} \end{cases}$$

Then we know that $\mathbf{E}_{s\sim\sigma}[C_i(s)] \leq \mathbf{E}_{s\sim\sigma}[C_i(s_{-i}, \delta(s_i)]$ by assumption. We can rewrite both sides by conditioning on all possibilities for s_i (the "law of total probability"), getting that

$$\sum_{x\in S_i} \left(\mathbf{Pr}[s_i = x] \cdot \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s)|s_i = x] \right) \leq \sum_{x\in S_i} \left(\mathbf{Pr}[s_i = x] \cdot \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s_{-i}, \delta(s_i))|s_i = x] \right).$$

Note that all terms on both sides are exactly the same except with $x = a$, so we can cancel them all out to get

$$\begin{aligned}
&\mathbf{Pr}[s_i = a] \cdot \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s)|s_i = a] \leq \mathbf{Pr}[s_i = a] \cdot \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s_{-i}, b)|s_i = a] \\
\implies &\mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s)|s_i = a] \leq \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s_{-i}, b)|s_i = a].
\end{aligned}$$

Since i, a, b were arbitrary, this holds for all i, a, b. This is precisely the definition of a correlated equilibrium from Definition 7.3.1! ■

So this theorem means that we can use an alternate definition of correlated equilibria:

Definition 7.3.3 *σ is a correlated equilibrium if for all $i \in [k]$ and for all $\delta : S_i \to S_i$,*

$$\mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s)] \leq \mathop{\mathbf{E}}_{s\sim\sigma}[C_i(s_{-i}, \delta(s_i))].$$