NETS 412: Algorithmic Game Theory ♡ February 14, 2017 ♡

Lecture 9

Lecturer: Aaron Roth *Scribe: Aaron Roth*

Achieving No Swap Regret

Recall that we argued last lecture that in any game, if every player plays according to the polynomial weights algorithm, then their empirical play history will converge to a coarse correlated equilibrium very quickly. In this lecture, we will give a new learning algorithm such that if every player plays according to it (in any game!) their empirical play history will converge to a correlated equilibrium.

To do this, we will again revisit the experts setting in which we derived the polynomial weights algorithm, but this time ask for an even stronger guarantee.

Recall our regret-based characterization of correlated equilibria:

Definition 1 *A distribution $\mathcal{D}$ over action profiles is an ϵ-approximate correlated equilibrium if for every player i, and for every strategy modification rule $F_i : A_i \to A_i$:*

$$\mathrm{E}_{a\sim\mathcal{D}}[\mathit{Regret}_i(a, F_i)] \leq \epsilon.$$

Recall that $\mathit{Regret}_i(a, F_i) = u_i(F_i(a_i), a_{-i}) - u_i(a)$.

Just as before, we will think about how to arrive at such distributions $\mathcal{D}$ by instead thinking about regret on a *sequence of play profiles* $a^1, \ldots, a^T$. The new notion of regret we will define is called *swap-regret*. To disambiguate this with the notion of regret on play sequences we discussed previously, we will now call the previous notion *external regret*.

Definition 2 *A sequence of action profiles $a^1, \ldots, a^T$ has swap-regret $\Delta(T)$ if for every player i, and every strategy modification rule $F_i : A_i \to A_i$ we have:*

$$\frac{1}{T}\sum_{t=1}^{T} u_i(a^t) \geq \frac{1}{T}\sum_{t=1}^{T} u_i(F_i(a_i), a_{-i}) - \Delta(T)$$

If $\Delta(T) = o_T(1)$, we say that the sequence of action profiles has no *swap regret.*

Unlike external regret, which merely guarantees that every player is achieving average utility as high as they would have had they played their best *fixed* action in hindsight, a sequence of action profiles with no swap regret guarantees that no player could do substantially better in hindsight even if they were allowed to *swap* every action of a particular type with an arbitrary different action, separately for each of their actions.

The reason we will be interested in regret on sequences of actions is because of the following simple consequence: if a sequence of action profiles has low swap regret, then the distribution $\mathcal{D}$ that selects uniformly at random amongst those profiles will be an approximate correlated equilibrium.

Theorem 3 *If a sequence of action profiles $a^1, \ldots, a^T$ has $\Delta(T)$ swap- regret, then the distribution $\mathcal{D} = \frac{1}{T}\sum_{t=1}^{T} a^t$ (i.e. the distribution that picks among the action profiles $a^1, \ldots, a^T$ uniformly at random) is a $\Delta(T)$-approximate correlated equilibrium.*

Proof This follows immediately from the definitions. For any player i:

$$\begin{aligned}\mathrm{E}_{a^t\sim\mathcal{D}}[\mathrm{Regret}_i(a^t, F_i)] &= \frac{1}{T}\sum_{t=1}^{T}\left(u_i(F_i(a_i^t), a_{-i}^t) - u_i(a^t)\right)\\ &\leq \Delta(T)\end{aligned}$$

■

Hence, we can approach the problem of finding correlated equilibria by reasoning about sequences of action profiles with small swap-regret. To do this, we revisit the problem of selecting amongst a set of experts. Recall the setting:

In rounds $t = 1, \ldots, T$:

1. The algorithm picks an expert $a_t \in \{1, \ldots, k\}$ from among the set of k experts.
2. Each expert i experiences loss ℓ_i^t, and the algorithm experiences loss $\ell_{a_t}^t$.

Write $L_{Alg}^T = \sum_{t=1}^T \ell_{a_t}^t$ for the cumulative loss of the algorithm after T rounds. We want to find an algorithm that can guarantee, for arbitrary sequences of losses:

$$\frac{1}{T} L_{Alg}^T \leq \frac{1}{T} \sum_{t=1}^{T} \ell_{F_i(a_t)}^t + \Delta(T)$$

for all $F_i : [k] \to [k]$ and for $\Delta(T) = o(1)$.

Note that this is a strictly stronger guarantee than the polynomial weights algorithm provides: we recover the guarantee of the polynomial weights algorithm by instantiating the above bound only with *constant* strategy modification rules F_i.

Lets think about how to do this. For a fixed sequence of decisions by our algorithm, define:

$$S_j = \{t : a_t = j\}$$

to be the set of time steps that the algorithm chose expert j.

One guiding observation will be the following: To achieve the desired bound, it would be sufficient that for every j:

$$\frac{1}{|S_j|} \sum_{t \in S_j} \ell_{a_t}^t \leq \frac{1}{|S_j|} \min_i \sum_{t \in S_j} \ell_i^t + \Delta(T)$$

In other words, we can achieve no *swap* regret if we can achieve no *external* regret separately on each sequence of actions S_j. This is because the best strategy modification rule in hindsight simply swaps each action j for the best fixed action in hindsight over S_j.

Following this idea, we will give an algorithm for achieving no swap regret that will run k separate copies of the polynomial weights algorithm, one for each action j.

The algorithm will work as follows:

1. Initialize k copies of the PW algorithm one for each action $j \in [k]$.
2. At each time t, denote by $q(1)^t, \ldots, q(k)^t$ the distribution maintained by each copy of the PW algorithm over the experts. We will combine these into a single distribution over experts $p^t \equiv (p_1^t, \ldots, p_k^t)$
3. The losses $\ell_1^t, \ldots, \ell_k^t$ for the experts arrive. To each copy i of the PW algorithm, we *report* losses $p_i^t \ell_1^t, \ldots, p_i^t \ell_k^t$ for each of the k experts. (i.e. to copy i, we report the true losses scaled by p_i^t).

The above algorithm is fully specified, except for how we combine the k PW distributions $q(1)^t, \ldots, q(k)^t$ into a single distribution over experts p^t. We do so as follows. For each expert j, define:

$$p_j^t = \sum_{i=1}^{k} p_i^t \cdot q(i)_j^t$$

The above set of equations have a unique solution (note that there are k linear equations in k unknowns).

Given the definition, we have two equivalent ways of viewing how experts are picked at each round t. Either:

1. Each expert i is chosen with probability p_i^t or
2. With probability p_i^t we select the i'th copy of the polynomial weights algorithm, and then select expert j according to the probability distribution $q(i)^t$.

From the perspective of the i'th copy of polynomial weights, its expected loss at round t is:

$$\sum_{j=1}^{k} q(i)_j^t \cdot (p_i^t \ell_j^t) = p_i^t \sum_{j=1}^{k} q(i)_j^t \ell_j^t$$

Therefore, by the guarantee of the polynomial weights algorithm, we have for all experts j^*:

$$\underbrace{\frac{1}{T}\sum_{t=1}^{T} p_i^t \sum_{j=1}^{k} q(i)_j^t \ell_j^t}_{LHS} \leq \underbrace{\frac{1}{T}\sum_{t=1}^{T} p_i^t \ell_{j^*}^t + 2\sqrt{\frac{\log k}{T}}}_{RHS}$$

Summing the left hand side of the above over all k copies of the PW algorithm, we get:

$$LHS = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{k} p_i^t \sum_{j=1}^{k} q(i)_j^t \ell_j^t = \frac{1}{T}\sum_{t=1}^{T}\sum_{j=1}^{k} p_j^t \ell_j^t = \frac{1}{T} L_{ALG}$$

i.e. the LHS is just the expected average loss of our algorithm! (Recall our two ways of viewing p_j...).

Recall that we can instantiate the RHS of the above inequality for any j^* we choose, and we can make a separate choice for each i. Fixing an arbitrary strategy modification rule $F : [k] \to [k]$, let us make the following choice: for each i, we choose $j^* = F(i)$. Then, summing up over all i on the right, we get:

$$RHS = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{k} p_i^t \ell_{F(i)}^t + 2k\sqrt{\frac{\log k}{T}}$$

Combining the two sides, we get:

$$\frac{1}{T} L_{ALG} \leq \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{k} p_i^t \ell_{F(i)}^t + 2k\sqrt{\frac{\log k}{T}}$$

In other words, we have proven the following theorem:

Theorem 4 *There is an experts algorithm that, against an arbitrary sequence of losses, after T rounds achieves $\Delta(T)$-swap regret for:*

$$\Delta(T) = 2k\sqrt{\frac{\log k}{T}}$$

A couple of things to note:

First, $\Delta(T) = o(1)$, and so this is a no-swap-regret algorithm, and hence if every player plays according to it in an arbitrary game, play converges to the set of correlated equilibria. Note that just like the polynomial weights algorithm, players need not know anything about the game to play it - they only need to be able to compute their utilities for the action profiles *actually played*, for each of their own actions.

Second, convergence is *fast*. Setting $\Delta(T) \leq \epsilon$, we see that we reach ϵ-swap regret (and hence ϵ-approximate correlated equilibrium) after T steps for:

$$T = \frac{4k^2 \ln(k)}{\epsilon^2}$$

This is larger than the time we needed to achieve no external regret by a factor of k^2, but still polynomial in the number of actions. Hence, not only do correlated equilibria exist in all games, they are always easy to find.