CS 684 Algorithmic Game Theory	Scribe: Bernard Wong
Instructor: Eva Tardos	November 9, 2005

Adaptive Game Playing and Zero Sum Games and Routing

In this class, we look at 2-person zero-sum games using (our) kind of learning, called no-regret learning. We will show that this game reaches an approximate Nash, and quantify the error.

1 Pennies and Bins game

First, let's review the game from last time which we will modify today into a zero-sum game. At time t, a_t pennies arrive in bin i. If the player is standing in the bin i at time step t, then he catches the penny. The player tries to maximize the number of pennies he collects by selecting the appropriate bin to stand in at each time step. The goal for to player is to collect at least as many pennies as the best static selection of a bin to stand in, i.e. collect $\geq \max_i \sum_t a_{it} = d$

From last time, we have an algorithm that depends on a variable ε that enables the player to get an expectation of $\geq d - \frac{\varepsilon d}{2} - \varepsilon^{-1} \log n$ pennies. If the player chooses $\varepsilon = 2\sqrt{d \log n}$ ahead of time, then he gets an expectation of $\geq d - 2\sqrt{2d \log n}$. This approaches our desired goal, as when d is large, the second term is relatively small. Of course, a caveat of this algorithm is that the player must choose an appropriate ε, which he must know ahead of time.

2 Mapping to 2-Player Zero Sum Game

The penny and bin game assumes that $0 \leq a_{it} \leq 1$. Therefore, this game is not a 2 player zero-sum game, as a 2 player zero sum game can't have both players make money by definition. The following are the steps to map this game into a zero-sum game.

Step 1:

Assume, without loss of generality, that the game matrix A has entries in the interval $[0, 1]$ by adding a constant α to every entry such that $A \geq 0$. Assume that every time a (row) player plays, he loses α.

Step 2:

We want to normalize the entries in the A matrix to limit the gain/loss per time step. For $\beta = \max a_{ij}$, replace a_{ij} with $\frac{a_{ij}}{\beta}$. No matter what happens, row player has to pay positive amount of money to the column player, and at most 1. Assume A is the original game matrix (with entries that can take any values), and $\tilde{A}$ is the normalized version with entries $\tilde{a}_{ij}$.

3 Approximate Nash

Now we determine what happens if row and column players play the algorithm from last time that assumes positive payoff. We will need to define what the payoff for the row player is (the player dropping the coins), as it was not previously defined.

The column player's selection of a column determines which bin is selected by the player trying to catch the penny. We know that after playing T steps, $d_c = \max_j \sum_t \tilde{a}_{i_t j}$ where i_t is the row choice in step t. By playing the algorithm from last class, we know that the column player gets in expectation $\geq d_c - \frac{\varepsilon d_c}{2} - \varepsilon^{-1} \log n$, where m is the number of rows, and n is the number of columns.

The selection of the row player determines which bin is selected to drop the penny. The row player is forced to play the game since he always loses money at every time step. He simply tries to lose less. The row player defines his payoff as $b_{ij} = 1 - \tilde{a}_{ij}$ where he loses 1 and gains b_j, and also uses no-regret learning. We know that after playing T steps, $d_R = \max_i \sum_t b_{i j_t}$, where j_t is the column players choice in time step t. In real terms, row player loses $T - gain \leq T - d_R + \frac{\varepsilon d_R}{2} - \varepsilon^{-1} \log n$ after T time steps.

Claim: These strategies reach an approximate Nash (approximate due to the secondary terms). Let p_i be the fraction of time row i was played and let q_i be the fraction of time column j was played (after T time steps, row i would have played $p_i T$ times).

Theorem: p and q are approximate Nash, if we can prove that we can choose ε that makes error (how far we are away from Nash) small. How do we check if it is a Nash for the column player? We look at what the player did and check if it is the best strategy.

First we mandate that the players play using the no-regret strategy. Now the column player has regret if there is a column where he would have made more income. How much income would he get in the best single column? $\max_j \sum_t \tilde{a}_{i_t j} = d_c$. We can write this using p by realizing that p_i equals the number of times $i_t = i$, divided by the period T, so $T \max_j \sum_i \tilde{a}_{ij} p_i = d_C$. Now we assumed that the column player played a strategy with small regret, so he must have an income at least

$$\begin{aligned} \text{column player's income} &\geq T \max_j \sum_i \tilde{a}_{ij} p_i - \frac{\varepsilon d_C}{2} - \varepsilon^{-1} \log n \\ &= T(\max_j \sum_i \tilde{a}_{ij} p_i - error). \end{aligned}$$

Now consider the row player. The row player keeps losing money, but we can think of his "income" as $1 - \tilde{a}_{ij}$ that is, we can think that he may lose as much as T in time T (given that 1 is the maximum loss in one step), and if he loses less, that is viewed as a gain to him. Assume, he plays a no-regret learning strategy with this $b_{ij} = 1 - \tilde{a}_{ij}$ as his income. Just as before, the income using a single strategy is then the $d_R = T \max_i \sum_j b_{ij} q_j$ Having played no regret, his income is not much less then this maximum income from the best single strategy.

$$\text{row player's income} \geq T \max_i \sum_j b_{ij} q_j - \frac{\varepsilon d_R}{2} - \varepsilon^{-1} \log n.$$

Now express this in terms of the game matrix $\tilde{A}$. First this income is really T minus his loss. Second, $d_R = T \max_i \sum_j b_{ij} q_j = T - T \min_i \sum_j \tilde{a}_{ij} q_j$. So we get a bound that his loss is not much more than the minimum loss from a single row.

$$\text{row player's loss} \quad \leq \quad T \min_i \sum_j \tilde{a}_{ij} q_j + \frac{\varepsilon d_R}{2} + \varepsilon^{-1} \log n.$$

Now we will show that q and p are approximate Nash strategies in some sense. First, for any matrix $\tilde{A}$ and any probability distributions p and q

$$\max_j \sum_i \tilde{a}_{ij} p_i \geq p\tilde{A}q \geq \min_i \sum_j \tilde{a}_{ij} q_j.$$

Second, the column player's income is the same as the row players loss by the definition of a 0-sum game. We normalize both income and loss by T to simplify the expression and we get

$$\max_j \sum_i \tilde{a}_{ij} p_i - error \leq \frac{1}{T}\text{column player's income} = \frac{1}{T}\text{row players loss} \leq \min_i \sum_j \tilde{a}_{ij} q_j + error.$$

Recall that the max is greater than equal to the min from above, so without the two error terms, this would imply that the max equals the min, and this is exactly the requirement for p and q to form a Nash. So with the error terms they are approximate Nash.

How big is the error? We don't know d_R and d_C ahead of time, but our normalization ensures that $d_R, d_C \leq T$. Using $\varepsilon = 2\sqrt{T \log n}$ both error terms in the regret bounds are bounded by $2\sqrt{2T \log n}$, and the errors in the last inequalities with normalized income is bounded by $2\sqrt{\frac{2 \log n}{T}}$, which tends to 0 as T tends to infinity (and is getting small once $T >> \log n$).