CS 684 Algorithmic Game Theory	Scribe: Kevin Walsh
Instructor: Eva Tardos	November 7, 2004

Adaptive Game Playing: Weighted Majority

We continue looking at the adaptive learning game introduced in the previous lecture. In each round of the basic game, a player tries to catch pennies that a dealer is dropping into bins. In this lecture, we formalize one possible player strategy, analyze its expected payoff, and consider a few generalizations of the game.

1 Learning Game

The game consists n bins, a player and a dealer, and game play proceeds in rounds. In each round, the dealer picks some bins in which to put a penny. The player picks a single bin each round, and gets rewarded if the dealer put a penny in the bin this round. Here we are allowing the dealer to place multiple pennies each round, but at most one in any single bin. The player's payoff in each round is either zero, or one penny, and so we call this a *zero-one game*.

The player sees all the actions of the dealer after each round ends, and so can learn which bins get the most pennies. The dealer does not see the player's actions before he needs to choose his action, but can see it after the round, so he also can use a learning strategy if he wants to. In the traditional learning approach, we typically assume that the dealer puts pennies according to some distribution, and the player's goal is to learn the distribution and adjust playing strategy accordingly. But from a game theory perspective, the dealer can have an arbitrary strategy, that depends on the player's strategy, or previous steps, and yet the player tries to learn how to make money.

We want to analyze the worst case outcome for the player's strategy, and so we can assume that the dealer knows the player strategy (but not the player's moves) a priori. So, the player strategy must randomize over all bins. The player tries to collect a lot of pennies by picking a good bin in each round. As a baseline, we define $d = d_{max}$ as the maximum number of pennies to fall in a single bin, which is the best payoff that could be achieved by picking and staying with a single bin. A simple player strategy that selects bin i with probability $1/n$ has expected payoff a $1/n$th fraction of all the pennies, but d_{max} can be much higher.

2 Player Strategy

The intuition is that the player picks a bin with probability proportional to the number of pennies that have fallen in the bin previously. Strict proportionality has several obvious problems. Instead, the player maintains a *weight* for each bin, and picks bins proportionally to the weights. Define:

$w_{i,t} \geq 0$	the weight of bin i for round t
$W_t = \sum_i w_{i,t}$	the total weight of bins in round t
$w_{i,0} = 1$	the initial weight of bin i, so $W_0 = n$
$p_{i,t} = w_{i,t}/W_t$	probability of picking bin i in round t

The weights are updated as pennies fall into each bin. If a penny falls in bin i during round t, then the player sets $w_{i,t+1} = (1+\epsilon)w_{i,t}$, where ϵ will be defined later.

3 Expected Payoff

We claim that the expected payoff of this strategy is close to d_{max}. Define

$a_{i,t}$	the number of pennies newly placed in bin i in round t
V_t	the expected number of new pennies collected in round t

For the basic game, we will have $a_{i,t} \in \{0,1\}$ and $V_t \in \{0,1\}$. By definition

$$V_t = \sum\nolimits_i p_{i,t} a_{i,t} = \sum\nolimits_i a_{i,t} \frac{w_{i,t}}{W_t}$$

so the total expected payoff over all rounds is just $\sum_t V_t = \sum_t \sum_i p_{i,t} a_{i,t}$. The weights are independent of the player's moves, so we can look at how the total weight changes after each round.

$$\begin{aligned} W_{t+1} &= W_t + \epsilon \sum\nolimits_i a_{i,t} w_{i,t} \\ &= W_t + \epsilon W_t \sum\nolimits_i a_{i,t} \frac{w_{i,t}}{W_t} \\ &= W_t + \epsilon W_t V_t \quad = \quad W_t(1+\epsilon V_t) \end{aligned}$$

Intuitively, if d_{max} is high, one bin's weight will dominate the total final weight W_f and cause W_t to grow approximately as $(1+\epsilon)$ per round, and so by the above $V_t \approx 1$. More formally,

$$W_f = W_0 \prod\nolimits_t (1+\epsilon V_t) = n \prod\nolimits_t (1+\epsilon V_t)$$

Also $W_f \geq \max_i w_i = (1+\epsilon)^d$. Combining these, and recalling that $\epsilon \geq \ln(1+\epsilon) \geq \epsilon - \frac{\epsilon^2}{2}$, we get

$$\begin{aligned} n \prod\nolimits_t (1+\epsilon V_t) &\geq (1+\epsilon)^d \\ \ln n + \sum\nolimits_t \ln(1+\epsilon V_t) &\geq d \ln(1+\epsilon) \\ \ln n + \epsilon \sum\nolimits_t V_t &\geq d(\epsilon - \frac{\epsilon^2}{2}) \\ \sum\nolimits_t V_t &\geq d - \frac{\ln n}{\epsilon} - d\frac{\epsilon}{2} \end{aligned}$$

The left term is exactly the total expected payoff, so the player selects selects ϵ to maximize the right term. This happens when $\epsilon = \sqrt{2\frac{\ln n}{d}}$, giving a payoff $\sum_t V_t \geq d - 2\sqrt{2d \ln n}$ close to d.

There is a slight cheat here. The player does not know d at the start of the game, and so cannot select ϵ. There is no clean solution to this problem: the player must learn d and adjust ϵ as the game is played.

4 Extensions

The above analysis considered a zero-one game, with $a_{i,t} \in \{0,1\}$, where the dealer always passes out whole pennies, and at most one penny per bin. If we allow larger rewards, $a_{i,t} \geq 1$, then no

strategy is likely to do well, since missing a single very large reward can throw off the players entire payoff. If we allow fractional but still positive rewards, $0 \leq a_{i,t} \leq 1$, the result from the previous section still holds. In fact, the entire analysis above is valid for this case as well, since we never used the assumption of whole pennies.