Computational Game Theory **Spring Semester, 2009/10**

Lecture 7: April 28 , 2010

Lecturer: Yishay Mansour *Scribe: Ghila Castelnuovo, Ran Roth* [1]

1 Regret Minimization

In this lecture, our goal is to build a strategy with good performance when dealing with repeated games. Let us start with a simple model of regret. In this model a player performs a partial optimization on his actions. Following each action he updates his belief and selects the next actions, dependent on the outcome. We will also show that for a familty of games, socially concave games, if all the players play a strategy to minimize the regret, the game converges to a Nash equilibrium.

2 Full Information Model

The model is defined as follows:

- Single player N = {1}
- A set of actions A=$\{a_1,..,a_m\}$
- For each step t the player chooses an action a_i (or a distribution p^t over A)
- For each step t the player receives a loss $l^t \in [0,1]^m$ where $l_i^t \in [0,1]$ is the loss of action $a_i \in A$
- A player's loss at step t is $\sum_{i=1}^{m} p_i^t l_i^t = l_{ON}^t$.
- Accumulative loss for a player is $L_{ON}^T = \sum_{t=1}^{T} \vec{l^t}\vec{p^t} = \sum_{t=1}^{T} l_{ON}^t$

2.0.1 Question: How to measure the player's achievement?

We must define a way measure the player's achievements. The first proposal is to measure the total loss. In this case, the adversary can create a serie choosing all the losses to be 1, and therefore in won't matter which strategy the player will choose, he will get the same achievement.
Another way is to compare the player's loss to the loss that he would get by choosing the best action at each step which results in a minimal loss $OPT = \sum_{t=1}^{T} min_i\{l_i^t\}$. This measure is similar to competitive online analysis and in our setting no interesting bound can be achieved. We introduce a new metric, which will help us measure the player's achievements.

3 External Regret

Let

$$L_i^T = \sum_{t=1}^{T} l_i^t \text{ the total loss of action } a_i \quad (1)$$

[1] These notes are based in part on the scribe notes of Lior Shapira and Eyal David from 2005/2006 and on the scribe notes of Eitan Yaffe and Noa Bar-Yosef from 2003/2004

Let H be a policy set. (Note that the most common choice is $H = A$)

$$L_{best}^T = min_{h \in H} L_h^T$$

Definition *Regret* $= \min\{L_{ON}^T - L_{best}^T, 0\}$.

We define the Regret as a way to measure the algorithm's performance and we wish to minimize the Regret. This reflects our desire to achieve performance close to the best static choice of action.

3.1 Minimizing External Regret - Deterministic Greedy Algorithm

For convenience we'll assume $l_i^t \in \{0, 1\}$ (so cumulative loss values will be integers). The algorithm G will try to minimize the Regret in the following way:

- For step $t = 1$ we choose a_1.
- For step $t > 1$ we choose the best action until now, i.e.,

$$a^t = arg\min_i L_i^{t-1}$$

Theorem 1 $L_G^T \leq m \cdot L_{best}^T + (m - 1)$

Proof: We define c_k to be the loss of G(the greedy algorithm) from time t, the first time in which $L_{best}^t = k$ and until time t',the first time in which $L_{best}^{t'} = k + 1$. At time t there are at most m actions with $L_i^t = k$. Each time G pays 1, the number of actions with a loss of k is reduced by 1. Therefore

$$c_k \leq m, \text{ which implies that } L_G = \sum_{k=0}^{L_{best}^T} c_k \leq m \cdot L_{best}^T + (m - 1)$$

□

Theorem 2 *For each deterministic algorithm D exists a series for which* $L_D^T \geq m \cdot L_{best}^T + Tmod(m)$

Proof: The opponent, at time t, defines a loss of 1 on a^t, the action that D selects at time t and 0 on the other actions. Algorithm D pays exactly $L_D^T = T$. However, by averaging there is an action i, such that $L_i^T \leq \lfloor \frac{T}{m} \rfloor$. This occurs because T "losses" are divided between m actions. And so $L_D^T \geq m \cdot L_{best}^T + Tmod(m)$ □

4 Randomized Algorithms

4.1 Randomized Greedy Algorithm - RG

Let $S^t = \{i | L_i^t = L_{best}^t\}$. The algorithm will randomly choose the action in the following way:

- For $t = 1$ we select a_i^t at random with $p_i^t = \frac{1}{m}$.
- For $t > 1$ we select a_i^t at random between the actions who had the minimal loss. I.e.

$$p_i^t = \begin{cases} \frac{1}{|S^{t-1}|} & \text{if } i \in S^{t-1} \\ 0 & \text{otherwise} \end{cases}$$

Theorem 3 $L_{RG} \leq (\ln(m) + 1) \cdot L_{best}^T + \ln m$

Proof: We define c_k as before. We assume that the opponent chooses to give a loss of 1 to one action out of S^t. First, it is always better for the opponent to select to give a loss to some action in S^t because he knows that the player will choose his action out of that set. In addition it is better to choose two actions in differing rounds rather than the same round because

$$\forall r, m \in \mathbb{N} \quad \frac{r}{m} \leq \frac{1}{m} + \frac{1}{m-1} + ... + \frac{1}{m-r+1}.$$

Therefore the expected loss for a round is

$$E[c_k] = \sum_{i=1}^{m} \frac{1}{i} \leq \ln(m) + 1$$

and therefore

$$L_{RG} = E[\sum_{k=0}^{L_{best}^T} c_k] = (\ln(m) + 1) \cdot L_{best}^T + \ln(m).$$

□

4.2 Randomized Weighted Majority algorithm

We saw that using randomization we improved from a ratio of m to $O(\ln(m))$. How can RG be improved? We notice that performance suffers when S^t is small and so we'll try giving actions a positive probability, even if they aren't in S^t.

The idea:

- For each action a_i we define a weight w_i^t such that $w_i^t = (1-\eta)^{L_i^{t-1}}$, when initially $w_i^1 = 1$ (since $L_0^i = 0$)
- The RWM algorithm selects a distribution $p_i^t = \frac{w_i^t}{W^t}$ where $W^t = \sum_{i=1}^{m} w_i^t$. (initially $p_i^1 = \frac{1}{m}$).

The algorithm:

- for $t = 1$: $w_i^1 = 1$, $p_i^1 = \frac{1}{m}$.
- for $t \geq 2$:
  $w_i^t = \begin{cases} (w_i^{t-1})^{(1-\eta)} & \text{if } l_i^{t-1} = 1 \\ w_i^{t-1} & \text{if } l_i^{t-1} = 0 \end{cases}$
  $W^t = \sum_{i=1}^{m} w_i^t$.
  $p_i^t = \frac{w_i^t}{W^t}$.

Theorem 4

$$\text{For } \eta \leq \frac{1}{2} \qquad L_{RWM}^T \leq (1+\eta) L_{best}^T + \frac{\ln(m)}{\eta}. \tag{2}$$

$$\text{For } \eta = \min\{\frac{1}{2}, \lceil \frac{\ln(m)}{T} \rceil\} \qquad L_{RWM}^T \leq L_{best}^T + 2\sqrt{T \ln(m)}. \tag{3}$$

Proof: The main idea is to follow the value of W^t. On the first hand we know that if RWM will have a large loss, then W^t will decrease significantly. On the second hand,

$$W^T \geq w_{best}^T = (1-\eta)^{L_{best}^T}. \tag{4}$$

and this gives us a bound to the maximum loss of RWM. Let

$$F^t = \frac{\sum_{i:l_i^t=1} w_i^t}{W^t} \tag{5}$$

the weight of the actions with loss 1 in step t. Note that F^t is exactly the loss of the algorithm RWM in step t, directly by the definition of the algorithm.

$$W^{t+1} = W^t(1-F^t) + W^t F^t(1-\eta) = W^t - \eta F^t W^t = W^t(1-\eta F^t). \tag{6}$$

And, because the decrease of W^t is proportional to the loss of RWM, we get that

$$(1-\eta)^{L_{best}^T} \leq W^{T+1} = W^1 \prod_{t=1}^{T}(1-\eta F^t) = m\prod_{t=1}^{T}(1-\eta F^t). \tag{7}$$

After taking the log of both sides, we get

$$L_{best}^T \ln(1-\eta) \leq \ln(m) + \sum_{t=1}^{T} \ln(1-\eta F^t) \tag{8}$$

and because $\ln(1-z) \leq -z$, we have

$$L_{best}^T \ln(1-\eta) \leq \ln(m) - \sum_{t=1}^{T} \eta F^t = \ln(m) - \eta L_{RWM}^T. \tag{9}$$

Rearranging the terms we get that:

$$L_{RWM}^T \leq \frac{\ln(m)}{\eta} - \frac{\ln(1-\eta)}{\eta} L_{best}^T \tag{10}$$

Since $-\ln(1-z) \leq z + z^2$ for each $z \in [0, \frac{1}{2}]$.

$$L_{RWM}^T \leq (1+\eta)L_{best}^T + \frac{\ln(m)}{\eta} \tag{11}$$

□

4.2.1 Lower bounds for WM

1. For m operations and $T = \frac{1}{2}$, we get that $Regret = \Omega(\ln(m))$. This is because of the following:
   We can create a distribution for the loss, such that with probability $\frac{1}{2}$ we have $l_i^t = 1$ and with probability $\frac{1}{2}$ we have $l_i^t = 0$.
   This means that with high probability there is an action which has a zero loss, which means that with high probability $L_{best} = 0$.

   $$1 - (1 - (\frac{1}{2})^T)^m = 1 - (1 - \frac{1}{\sqrt{m}})^m \approx 1 - e^{-\sqrt{m}} \tag{12}$$

   Hence, the loss's expectation for the best action is

   $$E[L_{best}^T] \leq e^{-\sqrt{m}} \ln(m) \approx 0. \tag{13}$$

   For each online algorithm, there is an expectation of loss $\frac{1}{2}$ for each time t. Therefore

   $$E[L_{ON}^T] = \frac{1}{2}T = \frac{1}{2}(\frac{1}{2}\ln(m)) \tag{14}$$

   $$E[Regret_{ON}] \geq \frac{1}{2}(\frac{1}{4}\ln(m)) = \Omega(\ln(m)) \tag{15}$$

2. For 2 operations and time T we get $Regret = \Omega(\sqrt{T})$. We create the distribution for the actions' loss such that with probability $\frac{1}{2}$ we give $(0,1)$ and with probability $\frac{1}{2}$ we give $(1,0)$.
   For every Online Algorithm $E[L_{ON}^T] = \frac{1}{2}T$. For L_{best} there is always an action with loss of at most $\frac{1}{2}T$, and with constant probability the loss of the best action is at most $\frac{1}{2}T - c\sqrt{T}$.
   Hence, $E[Regret_{ON}] \geq c'\sqrt{T}$.

5 Socially Concave Games

In this part of the lecture we return to games with muliple players and analyze what happens if all the players use an external regret minimization strategy. In particular, we show that for a certain class of games such strategies lead to a convergence towards equilibrium.

Definition Socially Concave Games

Let G be a game with players $N = \{1, ..., n\}$, utility functions u_i and actions taken from a set A. We say that G is socially concave if two requirements hold:

1. There exists a convex combination $\{\lambda_i\}$ such that the function $g(x) = \sum_{i \in N} \lambda_i u_i(x)$ is a concave function. (A convex combination means $\forall i, \lambda_i \geq 0$ and $\sum_i \lambda_i = 1$).
2. For every player i and an action $a_i \in A$, we have that $h(y) = u_i(a_i, y)$ is a convex function. In other words, fixing the player's behavior and looking at the actions of the other players gives a convex function.

5.1 Example of a socially concave game - Linear Cournot

An example of a socially concave game is a linear cournot competition, where the production costs for each player are convex in the amount produced. The utiliy functions are givn by:

$$u_i(s) := s_i p(s) - c_i(s_i)$$

The price is a linear function in the total amount produced:

$$p(s) := a - b \sum s_i$$

We show that the two requirements of a socially concave game hold:

1. Choosing a uniform combination λ_i gives:

$$g(s) = \frac{1}{n} \sum u_i(s) = \frac{1}{n} [\sum_{i \in N} s_i(a - b \sum_{j \in N} s_j) - \sum_{i \in N} c_i(s_i)]$$

   Taking the second derivative of the first sum shows it is concave. The second sum is a sum of concave functions and thus concave. The whole expression is therefore a concave function.

2. For a player i and a fixed action s_i the utility function becomes linear in the other actions and in particular convex:

$$u_i(s_i, x_{-i}) = s_i(a - \sum_{j \neq i} x_j - s_i) - c_i(s_i)$$

5.2 Equilibrium of socially concave games under regret minimization

In order to formally state the theorem we define an ϵ-Nash equilibrium:

Definition ϵ-Nash equilibrium

Let $G = (N, A, \{u_i\})$ be a game. $x \in A^n$ is an ϵ-Nash equilibrium if for every player $i \in N$ and every action $a_i \in A$ it holds that:

$$u_i(x) \geq u_i(a_i, x_{-i}) - \epsilon$$

Theorem 5 *G is a socially concave game. Assume each player i uses a strategy with external regret at most $R_i(t)$. Assume also that the game is played t times. Then the following holds:*

1. *The average state $\hat{x}^t$ is ϵ-Nash, where:*

$$\hat{x}^t := \frac{1}{t}\sum_{\tau=1}^{t} x^\tau$$

and:

$$\epsilon = \frac{1}{\lambda_{min}}\sum_{j\in N}\frac{R_j(t)\lambda_j}{t}$$

2. *The average utility of each player $\hat{u}_i^t$ is close to the utility of the average state:*

$$\hat{u}_i^t := \frac{1}{t}\sum_{\tau=1}^{t} u_i(x^\tau)$$

and:

$$|\hat{u}_i^t - u_i(\hat{x}_i^t)| \leq \frac{1}{\lambda_i}\sum_{j\in N}\frac{\lambda_j R_j(t)}{t}$$

Proof: We have:

$$\begin{aligned}
\hat{u}_i^t &= \frac{1}{t}\sum_{\tau=1}^{t} u_i(x^\tau) \\
&\geq \max_{a_i\in A}\frac{1}{t}[\sum_{\tau=1}^{t} u_i(a_i, x_{-i}^\tau) - R_i(t)] \\
&\geq \frac{1}{t}[\sum_{\tau=1}^{t} u_i(BR(\hat{x}_{-i}^t), x_{-i}^\tau) - R_i(t)] \\
&\geq u_i(BR(\hat{x}_{-i}^t), \hat{x}_{-i}^t) - \frac{R_i(t)}{t}
\end{aligned}$$

where the last inequality is due to convexity (requirement no. 2 in the definition of concave games).

Since the best response brings the utility of player i to maximum, we have:

$$\hat{u}_i^t \geq u_i(BR(\hat{x}_{-i}^t), \hat{x}_{-i}^t) - \frac{R_i(t)}{t} \geq u_i(\hat{x}^t) - \frac{R_i(t)}{t} \tag{16}$$

On the other hand, we have from the concavity of $g(x)$:

$$\begin{aligned}
\sum_{i\in N}\lambda_i u_i(\hat{x}^t) &= g(\frac{1}{t}\sum_{\tau=1}^{t} x^\tau) \\
&\geq \frac{1}{t}\sum_{\tau=1}^{t} g(x^\tau) \\
&= \frac{1}{t}\sum_{\tau=1}^{t}\sum_{i\in N}\lambda_i u_i(\hat{x}^\tau) \\
&= \sum_{i\in N}\lambda_i\frac{1}{t}\sum_{\tau=1}^{t} u_i(\hat{x}^\tau) \\
&= \sum_{i\in N}\lambda_i\hat{u}_i^t
\end{aligned}$$

Thus we get:

$$\sum_{i\in N} \lambda_i u_i(\hat{x}^t) \geq \sum_{i\in N} \lambda_i \hat{u}_i^t \tag{17}$$

By taking the convex combination of inequality (16) for all players, and combining it with inequality (17) we get:

$$\begin{aligned}\sum_{i\in N} \lambda_i \hat{u}_i^t &\geq \sum_{i\in N} \lambda_i u_i(BR(\hat{x}_{-i}^t), \hat{x}_{-i}^t) - \sum_{i\in N} \frac{\lambda_i R_i(t)}{t} \\ &\geq \sum_{i\in N} \lambda_i u_i(\hat{x}^t) - \sum_{i\in N} \frac{\lambda_i R_i(t)}{t} \\ &\geq \sum_{i\in N} \lambda_i \hat{u}_i^t - \sum_{i\in N} \frac{\lambda_i R_i(t)}{t}\end{aligned}$$

Thus,

$$|\sum_{i\in N} \lambda_i [u_i(BR(\hat{x}_{-i}^t), \hat{x}_{-i}^t) - u_i(\hat{x}^t)]| \geq \sum_{i\in N} \frac{\lambda_i R_i(t)}{t} \tag{18}$$

But since this is in fact a sum of positive values (the utility of the best response for player i is always equal or greater than choosing any other action), we can ommit the absolute value. Dividing by λ_{min} and using a simple mean bound, we get:

$$\forall i, \quad u_i(BR(\hat{x}_{-i}^t), \hat{x}_{-i}^t) - u_i(\hat{x}^t) \leq \frac{1}{\lambda_i} \sum_{i\in N} \frac{\lambda_i R_i(t)}{t} = \epsilon \tag{19}$$

Because this is true for the best response, it holds for any other action player i can take. We have shown that $\hat{x}$ is an ϵ-Nash equilibrium.

It remains to show the second part of the theorem. From (16) it follows that:

$$\sum_{j\in N} \lambda_j [u_j(\hat{x}^t) - \hat{u}_j^t] \geq 0 \tag{20}$$

Therefore, for a specific player i we have:

$$u_i(\hat{x}^t) - \hat{u}_i^t \leq \frac{1}{\lambda_i} \sum_{j\in N, j\neq i} \lambda_j [u_j(\hat{x}^t) - \hat{u}_j^t] \tag{21}$$

Combining this with (17) we get

$$u_i(\hat{x}^t) - \hat{u}_i^t \leq \frac{1}{\lambda_i} \sum_{j\in N} \frac{\lambda_j R_j(t)}{t} \tag{22}$$

Thus:

$$|u_i(\hat{x}^t) - \hat{u}_i^t| \leq \frac{1}{\lambda_i} \sum_{j\in N} \frac{\lambda_j R_j(t)}{t} \tag{23}$$

□