Machine Learning **Fall Semester, 2012/13**

Lecture 5: November 18, 2012

Lecturer: Yishay Mansour *Scribe: Yuval Snappir, Ariel Persiko* [1]

1 Regret Minimization

Online Model:

- We are given a sample x_t
- We generate a guess b_t
- We see the true classification $c^*(x_t)$

Let H be a finite class of hypotheses (classification functions). In the HAL algorithm in each iteration we partition H into two groups and go with the majority decision. If $c^* \in H$ then the HAL algorithm will perform $O(log\,|H|)$ errors.

What happens when $c^* \notin H$?
We would like to find the best $h \in H$. Intuitively, we would like to bound the number of errors by the error of the best h and some additional constant.

An algorithmic model:

1. For each step t, the learner chooses a combination of weights $w_1^t, ..., w_{|H|}^t$ for all $h \in H$ such that $\sum_{i=1}^{|H|} w_h^t = 1$, and $w_h^t \geq 0$.
2. For each step t, the learner receives a loss $l^t \in [0,1]^{|H|}$ where $l_i^t \in [0,1]$ is the loss of hypothesis $h \in H$, the loss in the algorithm is $\sum_{h \in H} l_h^t w_h^t$.

2 External Regret

Let

$$L_h^T = \sum_{t=1}^{T} l_h^t, \tag{1}$$

and let

$$L_{best}^T = min_{h \in H} L_h^T. \tag{2}$$

Similarly, L_A^T is the accumulated loss of algorithm A at time T, defined as

$$L_A^T = \sum_{t=1}^{T} \sum_{h=1}^{|H|} w_h^t l_h^t. \tag{3}$$

[1] These notes are based in part on the scribe notes of Ghila Castelnuovo and Ran Roth, Computational Game Theory 2009/2010

After T rounds we would like to achieve:

$$L_A^T - L_{best}^T \leq ExternalRegret.$$

We want to describe algorithms that minimizes the external regret.

3 Deterministic Greedy Algorithm (G)

The algorithm G will try to minimize the regret in the following way:

- For step $t = 1$ we choose h_1.
- For step $t > 1$ we choose some best hypothesis until now, i.e.,

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

if there is more than one such hypothesis we will choose the lexicographically lowest.

Theorem 1 *For algorithm G:* $L_G^T \leq |H| \cdot L_{best}^T + |H| - 1$.

Proof: We define B_k to be the set of time steps where $L_{best}^t = k$. In the first time step of B_k, there are at most $|H|$ hypotheses with $L_h^t = k$. Every time G errs, the number of hypotheses adhering to $L_{best}^t = k$ is reduced by 1. After $|H|$ errors, L_{best}^t must grow by 1. Therefore, $L_G^{B_k}$, the total loss in the time steps B_k, is bound by $|H|$. Hence,

$$L_G \leq \sum_{k=1}^{L_{best}^T} L_G^{B_k} + |H| - 1 \leq |H| \cdot L_{best}^T + |H| - 1.$$

Note that in the worst case in the last iteration has $|H| - 1$ errros, since by definition at least one hypothesis must achieve L_{best}^T. □

Theorem 2 *For each deterministic algorithm D (D picks a hypothesis* $h \in H$ *per time step), there exist a series of losses for which* $L_D^T \geq |H| \cdot L_{best}^T + Tmod|H|$.

Proof: In the worst case, an adversary who knows the algorithm D may cause it to incur a loss of 1 on each h^t, the hypothesis that D selects at time t and any $h \neq h^t$ has loss of 0 at time t. Algorithm D will incur $L_D^T = T$. We can write:

$$L_D^T = T = \tfrac{T}{|H|} \cdot |H| = |H| \cdot \lfloor \tfrac{T}{|H|} \rfloor + Tmod|H|$$

By averaging, there must be an hypothesis h, such that $L_h^T \leq \lfloor \frac{T}{|H|} \rfloor$. This occurs because a total of T losses are partition between $|H|$ hypotheses. And so $L_{best}^T \leq \lfloor \frac{T}{|H|} \rfloor$ and we obtain,

$$L_D^T \geq |H| \cdot \lfloor \tfrac{T}{|H|} \rfloor + Tmod|H| \geq |H| \cdot L_{best}^T + Tmod|H|.$$

□

4 Randomized Greedy Algorithm (RG)

Let $H^t = \{h \in H | L_h^t = L_{best}^t\}$. The algorithm RG will randomly choose a hypothesis in the following way:

- For $t = 1$ we select h^t at random with $p_h^t = \frac{1}{|H|}$.
- For $t > 1$ we select h^t at random between the hypotheses who had the minimal loss, i.e,

$$p_h^t = \begin{cases} \frac{1}{|H^{t-1}|} & \text{if } h_i \in H^{t-1} \\ 0 & \text{otherwise} \end{cases}$$

Theorem 3 *For algorithm RG:* $L_{RG}^T \leq (\ln|H| + 1) \cdot L_{best}^T + \ln|H|$.

Proof: We define B_k as before. Let us also define $m = |H^t| \leq |H|$. We assume WLOG that the adversary chooses to give a loss of 1 to exactly one hypothesis out of H^t. It is better for an adversary to choose r hypotheses in different rounds rather than all in same round because

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

Therefore, the expected loss for the time steps in B_k is,

$$E[L_{RG}^{B_k}] = \sum_{i=1}^{m} \frac{1}{i} \leq \ln(m) + 1,$$

and therefore

$$L_{RG} = E[\sum_{k=0}^{L_{best}^T} L_{RG}^{B_k}] \leq (\ln|H| + 1) \cdot L_{best}^T + \ln|H|.$$

□

5 Randomized Weighted Majority Algorithm (RWM)

In the RWM algorithm, we'll choose some constant η. For every hypothesis $h \in H$ at step t we define a weight:

$$w_h^t = (1-\eta)^{L_h^{t-1}},$$

or in a recursive definition:

$$w_h^1 = 1$$
$$w_h^{t+1} = \begin{cases} w_h^t & \text{if } h \text{ is correct in step } t \\ (1-\eta)w_h^t & \text{if } h \text{ mistakes in step } t. \end{cases}$$

The probability of every h at step t will be calculated by normalizing the weights:

$$W^t = \sum_{h \in H} w_h^t, \text{ and } p_h^t = \frac{w_i^t}{W^t}.$$

Theorem 4 *For* $0 < \eta \leq \frac{1}{2}$*:*

$$L_{RWM}^T \leq (1+\eta)L_{best}^T + \frac{ln|H|}{\eta},$$

and for $\eta = min\left\{\frac{1}{2}, \sqrt{\frac{ln|H|}{T}}\right\}$*:*

$$L_{RWM}^T \leq L_{best}^T + 2\sqrt{Tln|H|}$$

Proof: Let:

$$F^t = \frac{\sum_{h \in H: l_h^t = 1} w_h^t}{W^t} = \sum_{h \in H: l_h^t = 1} p_h^t$$

We can see that F^t is the sum of probabilities of the hypotheses that made mistake in time t, which is exactly the loss of RWM in time t, therefore:

$$L_{RWM}^T = \sum_{t=1}^{T} F^t.$$

Furthermore, from previous definitions we can see:

$$W^{t+1} = \sum_{h \in H} w_h^{t+1} = \sum_{h \in H: l_h^t = 0} w_h^{t+1} + \sum_{h \in H: l_h^t = 1} w_h^{t+1}$$

$$= \sum_{h \in H: l_h^t = 0} w_h^t + \sum_{h \in H: l_h^t = 1} (1-\eta) w_h^t$$

$$= \sum_{h \in H} w_h^t - \sum_{h \in H: l_h^t = 1} \eta w_h^t$$

$$= W^t - \eta F^t W^t = W^t(1 - \eta F^t) = W^1 \prod_{\tau=1}^{t} (1 - \eta F^\tau)$$

Therefore:

$$W^{T+1} = |H| \prod_{t=1}^{T} (1 - \eta F^t).$$

On the other hand:

$$\forall h \in H \quad W^{T+1} \geq w_h^{T+1} = (1-\eta)^{L_h^T},$$

and therefore:

$$W^{T+1} \geq (1-\eta)^{L_{best}^T}.$$

Together we get:

$$(1-\eta)^{L_{best}^T} \leq |H| \prod_{t=1}^{T} (1 - \eta F^t)$$

$$L_{best}^T ln(1-\eta) \leq ln|H| + \sum_{t=1}^{T} ln(1 - \eta F^t),$$

It is known that $\forall z \in \left[0, \frac{1}{2}\right]$ we have $-z - z^2 \leq ln(1-z) \leq -z$. This is true both for $z = \eta$ on the left of the last inequality and $z = \eta F^t$ on the right (since $0 \leq F^t \leq 1$ by definition). Therefore:

$$L_{best}^T(-\eta - \eta^2) \leq ln|H| - \sum_{t=1}^{T} \eta F^t$$

$$\eta \sum_{t=1}^{T} F^t \leq (\eta + \eta^2) L_{best}^T + ln|H|$$

$$\sum_{t=1}^{T} F^t \leq (1+\eta) L_{best}^T + \frac{ln|H|}{\eta}$$

Combining this with the fact from the beginning: $\sum_{t=1}^{T} F^t = L_{RWM}^T$, we get:

$$L_{RWM}^T \leq (1+\eta)L_{best}^T + \frac{ln|H|}{\eta}.$$

The second part of the theorem is derived from the first part. Since $L_{best}^T \leq T$, if $\eta = \sqrt{\frac{ln|H|}{T}} \leq \frac{1}{2}$ then from the last inequality we get:

$$L_{RWM}^T \leq L_{best}^T + \eta L_{best}^T + \frac{ln|H|}{\eta} \leq L_{best}^T + \eta T + \frac{ln|H|}{\eta} = L_{best}^T + \sqrt{\frac{ln|H|}{T}}T + \frac{ln|H|}{\sqrt{\frac{ln|H|}{T}}} = L_{best}^T + 2\sqrt{Tln|H|}$$

On the other hand, if $\eta = \frac{1}{2} < \sqrt{\frac{ln|H|}{T}}$, then $\sqrt{T} < 2\sqrt{ln|H|}$, and $T \leq 2\sqrt{Tln|H|}$. Clearly the regret is at most T.

□

5.1 Lower Bounds for Online Weighted Algorithms

After showing upper bounds for the external regret, we should ask ourselves how well can online weighted minimize the regret? We'll try to find a lower bound for the regret in 2 cases:

5.1.1 $T = \frac{1}{2}log_2|H|$ short time case

In this case let's build a distribution in which for every step, for any $h \in H$, the probability that h is correct equals to the probability that it mistakes:

$$\forall h \in H, 1 \leq t \leq T \quad Pr[l_h^t = 0] = \frac{1}{2} = Pr[l_h^t = 1]$$

And decide that the steps are independant (no memory between steps). Therefore

$$\forall h \in H \quad Pr[L_h^T = 0] = \left(\tfrac{1}{2}\right)^T = \left(\tfrac{1}{2}\right)^{\frac{1}{2}log_2|H|} = \tfrac{1}{\sqrt{|H|}}$$

$$\forall h \in H \quad Pr[L_h^T \neq 0] = 1 - \tfrac{1}{\sqrt{|H|}}$$

$$Pr[L_{best}^T \neq 0] = Pr[\forall h \in H \quad L_h^T \neq 0] = \left(1 - \tfrac{1}{\sqrt{|H|}}\right)^{|H|} \approx e^{-\sqrt{|H|}}$$

$$Pr[L_{best}^T = 0] = Pr[\exists h \in H \quad L_h^T = 0] = 1 - \left(1 - \tfrac{1}{\sqrt{|H|}}\right)^{|H|} \approx 1 - e^{-\sqrt{|H|}}$$

Because we chose a uniform distribution on $\{0,1\}$, for any online algorithm ON we get:

$$E[L_{ON}^T] = \frac{1}{2}T = \frac{1}{4}log|H|$$

There are only T samples and therefore

$$E[L_{best}^T] \leq Te^{-\sqrt{|H|}}$$

Using the linearity of the expected value we get that

$$E[\text{Regret}] \geq E[L_{ON}] - E[L_{best}^T] = \frac{1}{4}log|H| - \frac{1}{2}log|H|e^{-\sqrt{|H|}} \approx \frac{1}{4}log|H|$$

5.1.2 $|H| = 2$

Let's assume that $H = \{h_1, h_2\}$. In this case let's build a distribution in which for every step, with probability $\frac{1}{2}$ we have that h_1 is correct and h_2 mistakes, and with probability $\frac{1}{2}$ we have that h_2 is correct and h_1 mistakes. Let's also decide that the steps are not correlated (no memory between steps). Therefore, if some online algorithm ON decides to put at step t the weights: $(p, 1-p)$, we'll get:

$$E[l_{ON}^t] = \frac{1}{2}p + \frac{1}{2}(1-p) = \frac{1}{2}$$

Therefore:

$$E[L_{ON}^T] = \frac{1}{2}T$$

What about L_{best}^T? Because in each step exactly one hypothesis is right: $L_{h_1}^T + L_{h_2}^T = T$, or in other words:

$L_{h_1}^T = \frac{1}{2}T - \frac{1}{2}(L_{h_2}^T - L_{h_1}^T)$

$L_{h_2}^T = \frac{1}{2}T - \frac{1}{2}(L_{h_1}^T - L_{h_2}^T)$

$L_{best}^T = min\left\{L_{h_1}^T, L_{h_1}^T\right\} = \frac{1}{2}T - \frac{1}{2}|L_{h_1}^T - L_{h_2}^T| = \frac{1}{2}T - \frac{1}{2}|2L_{h_1}^T - T| = \frac{1}{2}T - |L_{h_1}^T - \frac{1}{2}T| = \frac{1}{2}T - |L_{h_1}^T - E[L_{h_1}^T]|$

$E[\text{Regret}_{ON}] = E[L_{ON}^T] - E[L_{best}^T] = E[\frac{1}{2}T - L_{best}^T] = E\left[|L_{h_1}^T - E[L_{h_1}^T]|\right]$

Since h_1 can mistake in every step with probability $\frac{1}{2}$, $L_{h_1}^T$ has a binomial distribution with $p = \frac{1}{2}$. Therefore for any constant c the probability that $|L_{h_1}^T - E[L_{h_1}^T]| \geq c\sqrt{T}$ is always greater than some constant probability $p_c > 0$ (because when $T \to \infty$ we get a normal distribution for $\frac{L_{h_1}^T - E[L_{h_1}^T]}{\sqrt{T}}$). Finally, using Markov's inequality with $c\sqrt{T}$, we can get:

$$E[\text{Regret}_{ON}] = E\left[|L_{h_1}^T - E[L_{h_1}^T]|\right] \geq c\sqrt{T}Pr\left[|L_{h_1}^T - E[L_{h_1}^T]| \geq c\sqrt{T}\right] \geq (c\sqrt{T})p_c = \Omega(\sqrt{T})$$

6 Multi Arm Bandit

In this model, there is a set A of actions from which the player has to choose in every step $1 \leq t \leq T$ (he can choose it randomly). After choosing the action, he can see the loss of his action, but cannot see the losses of other actions that he could choose.

6.1 Stochastic Model

For every action $a \in A$, the player will **lose** according to a random variable X_a with distribution over $[0, 1]$.

An example for this model is an imagniary casino in which we get a total of T turns to play in A slot machines, and in every machine we can earn in a different distribution a value between 0 and 1. On one hand, we want to test all the machines enough times, so we could estimate which machine has the best expectation value (or close enough to the best expectation). On the other hand, we don't want to focus on the one machine that we think to be good enough, and not "waste" our turns. This tradeoff is called: "Exploration vs Exploitation".

We'll now study some algorithms to solve this tradeoff.

6.2 Test and Play

This algorithm is composed of 2 phases:

1. Test: For every $a \in A$, sample it $O\left(\frac{1}{\epsilon^2} ln \frac{|A|}{\delta}\right)$ times, and then calculate the average loss $\hat{\mu}_a$.

2. Play: Choose the action with the best average loss: $a_{test} = argmin_{a \in A} \hat{\mu}_a$, and use it for the rest of the turns.

Theorem 5 *Let:*

$$\mu^* = min_{a \in A} \mu_a \text{ and } a^* = argmin \mu_a,$$

With probability of at least $1 - \delta$*, the action* a_{test} *that was chosen after the Test phase, maintains:*

$$\mu^* - \mu_{a_{test}} \leq \epsilon$$

Proof: From the Chernoff Bound, we get that $|\hat{\mu_a} - \mu_a| \leq \frac{\epsilon}{2}$. Therefore:

$$\mu_{a_{best}} \geq \hat{\mu}_{a_{best}} - \frac{\epsilon}{2} \geq \hat{\mu^*_{a^*}} - \frac{\epsilon}{2} \geq \mu^* - \epsilon$$

□

6.2.1 Upper Bound for the Regret of Test And Play

Theorem 6 *The regret of TEST and PLAY is at most* $O\left(T^{\frac{2}{3}} |A|^{\frac{1}{3}} ln^{\frac{1}{3}} (|A|T)\right)$

Proof: We'll now bound the Regret of the last algorithm.

Because the regret for every action in every time step is at most 1, the regret of any action that we choose is at most the number of steps that we use it. Therefore, in the first phase we get a regret of at most: $O\left(\frac{|A|}{\epsilon^2} ln \frac{|A|}{\delta}\right)$.

In the second phase, if $\mu_{a_{test}}$ is ϵ-close to μ^*, then the expected value of the regret is at most $\epsilon \cdot T$. Otherwise, we'll stay with the bound T. Note: the first case has $1 - \delta$ probability and the second has δ probability.

Together we get:

$$E[Regret] \leq O\left(\frac{|A|}{\epsilon^2} ln \frac{|A|}{\delta}\right) + \epsilon T(1 - \delta) + T\delta$$

By taking $\delta = \frac{1}{T}$ we can get:

$$E[\text{Regret}] = O\left(\frac{|A|}{\epsilon^2} ln\left(|A|T\right)\right) + \epsilon T(1 - \tfrac{1}{T}) + 1$$

$$= 1 - \epsilon + \epsilon T + O\left(\frac{|A|}{\epsilon^2} ln\left(|A|T\right)\right)$$

To get the lowest regret bound, we can set: $\epsilon = \left(\frac{|A| ln(|A|T)}{T}\right)^{\frac{1}{3}}$, so:

$$\epsilon T = |A|^{\frac{1}{3}} ln^{\frac{1}{3}} (|A|T) T^{\frac{2}{3}}$$

$$\frac{|A|}{\epsilon^2} ln(|A|T) = \frac{|A|}{\left(\frac{|A| ln(|A|T)}{T}\right)^{\frac{2}{3}}} ln\left(|A|T\right) = |A|^{\frac{1}{3}} ln^{\frac{1}{3}} \left(|A|T\right) T^{\frac{2}{3}}$$

And finally we get the bound:

$$E[\text{Regret}] = O\left(T^{\frac{2}{3}} |A|^{\frac{1}{3}} ln^{\frac{1}{3}} (|A|T)\right)$$

□

The main qualitative diffrence is in the dependence on number of action $|A|$. In the MAB the dependence is polynomial while in the full information we have a logarithimc dependence on $|A|$.

7 UCB Model

In the previous algorithm, the exploitation phase started only after the exploration phase had already ended. While this enables an almost accurate understanding of each and every arm (or action), the regret increases due to the long, and possible unnecessary, exploration phase. We will introduce now a more efficient algorithm, one that takes into account two factors:

1. **Current knowledge**: The less observed loss the action has, the higher the chancess of picking it.
2. **Uncertainty measurement**: The less we explored the action, the higher the chances of picking it.

These two factors yield a similar algorithm to the one we have already seen, but such that calculates an *Upper Confidence Bound (UCB)* on the expected profit of every action. This bound, rather than the empricial expected profit of the action, dictates which action to try next. We will present the UCB with losses and the goal of minimizing the losses.
So, for every action j, we hold L_t - the total amount of loss from action j, and T_j - the number of times we selected action j. We can easily see that the empirical expected loss is given by $\frac{L_j}{T_j}$. As stated before, we would add an additional factor to the empricical expected loss, in order to calculate the upper confidence bound.
For initialization , we try every possible action once, so as to have some notion about its expected loss.

7.1 UCB Algorithm

Initialization
for t in 1,..,d
  Pull arm $y_t = t$
  Observe lose l_t
  Set $L_t = l_t$ and $T_t = 1$
end for
Loop
for $t = d+1, d+2, ..$
  Pull arm $y_t \in Argmin_j(\frac{L_j}{T_j} + \sqrt{\frac{2ln(n)}{T_j}})$
  Observe lose l_t
  Set $L_{y_t} = L_{y_t} + r_t$ and $T_{y_t} = T_{y_t} + 1$
end for

7.2 UCB Analysis

Theorem 7 *Let $\Delta_a = \mu^* - \mu_a$, the UCB regret is:*

$$Regret(UCB) \leq 8ln(T)\sum_{a \neq a^*} \frac{1}{\Delta_a} + 4\sum_{a \neq a^*} \Delta_a$$

Proof: We shall bound the expected regret. The expected regret equals:

$$E[Regret] = \sum_{a \neq a^*} \Delta_a E[T_a].$$

$$T_a = 1 + \sum_{t=|A|+1}^{T} I\{a^t = a\}$$

$$\leq l + \sum_{t=|A|+1}^{T} I\{a^t = a, T_a^{t-1} \geq l\}$$

$$\leq l + \sum_{t=|A|+1}^{T} \sum_{r=l}^{T} \sum_{s=0}^{T} I\{\hat{\mu}_a^r - \sqrt{\frac{2log(t)}{r}} \leq \hat{\mu}_{a^*}^s - \sqrt{\frac{2log(t)}{s}}\}$$

The last inequality over comes the dependencies, by adding a sample of action a and a^* up to T times, and considering all combinations of number of trials of a and a^*.

Now observe that the indicator is 1 implies that at least one of the following must hold:

$$\hat{\mu}_{a^*}^s \geq \mu^* + \sqrt{\frac{2log(t)}{s}} \tag{4}$$

$$\hat{\mu}_a^r \leq \mu_a - \sqrt{\frac{2log(t)}{r}} \tag{5}$$

$$\mu^* \leq \mu_a + 2\sqrt{\frac{2log(t)}{r}} \tag{6}$$

If simultaneously (4), (5) and (6) do not had, then $\hat{\mu}_a > \hat{\mu}_{a^*}$.
Using the Chernoff bound we can bound the probabilities of the first and second events by $\frac{1}{t^4}$.
For $l = |\frac{8log(t)}{\Delta_a^2}|$ the third event can't hold because

$$0 < \mu^* - \mu_a - 2\sqrt{\frac{2log(t)}{r}} = \mu^* - \mu_a - \Delta_a = 0$$

Thus

$$E[T_a^t] \leq l + \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{r=l}^{t-1} \frac{2}{t^4} \leq \frac{8log(t)}{\Delta_a^2} + 1 + \frac{\pi^2}{3}$$

□