CS269: Machine Learning Theory
Lecture 9: Regret Minimization and Randomized Weighted Majority
October 25, 2010

Lecturer: Jacob Abernethy
Scribe: Cesar Romero and Suming Chen

1 Regret

Assume we are playing a game. In this game, we see samples of random variables and we would like to estimate the value of the next sample. Let's say that we want to "guess" the value of X_{t+1}, $\hat{X}$. In this game, we will "suffer" the square distance $(\hat{X} - X_{t+1})^2$. A typical strategy is to set $\hat{X} = \frac{1}{T}\sum_i^T X_t$. It turns out that this is actually pretty good. In fact, if we know that X comes from a Gaussian distribution, this is the optimal strategy.

Assume we know that X_i is drawn from a distribution $X \sim N(\mu, \sigma^2)$. The optimal thing to do in this case is to set $\hat{X} = \mu$. In general, however, we don't know the true distribution. A common question in statistics is: How well can I do using the information from my samples compared to how well I could have done had I known the distribution in advance? This is precisely where the notion of regret comes from, which we could state as

$$Cost_T(Alg) - Cost(OPT)$$

Clearly, we would like this quantity to be equal to zero in the limit as T gets very large.

2 Comparative Benchmarks

It's interesting to consider this notion of regret even when the data is not i.i.d. but chosen by an adversary. That brings us to a general definition. Informally, we can think of the regret of an algorithm on a particular sequence of data as the difference between how well an optimal algorithm that could observe the sequence of data in advance could do. A question we would like to answer is: What can the optimal algorithm (OPT) do with that data?

We cannot define OPT to be the optimal action on each round because that's not really a fair comparator. Instead, we define it to be the optimal fixed strategy w^*. Imagine that we have some cost function $Cost(W_t, X_t)$, where W_t is the algorithm's decision and X_t is the "outcome".

$$Regret(Alg) = \sum_{t=1}^{T} cost(W_t, X_t) - \min_{W*} \sum_{t=1}^{T} cost(W*, X_t) = f(T) \tag{1}$$

We want $f(T) = o(T)$, i.e. $\frac{f(T)}{T} \to 0$. This is similar to the stochastic setting, in which we want error to go to 0 over time. It turns out that, in many cases, we can actually guarantee this. Let us now define a model.

	R	P	S
R	0	1	-1
P	-1	0	1
S	1	-1	0

Table 1: matrix of losses for player 1 for rock paper sissors

3 The Expert Advice Model

Recall the expert advice model that we introduced in the last lecture. On a sequence of rounds $t = 1, \ldots, T$ a player chooses an action $i_t \in \{1, \ldots, n\}$. The adversary chooses costs or *losses* for each action $l_t(1), \ldots, l_t(n) \in [0, 1]$. The player pays $l_t(i_t)$, and observes l_t.

This is an unfair scenario for the player because the adversary can always choose the function that forces the player to pay the highest cost. Instead, we change the model by letting the player pick a distribution over the actions $\{1, \ldots, n\}$. Then the player pays $\mathbb{E}[l_t(I)]$, observes l_t, updates $p_{t+1} \in \Delta_n$, where Δ_n is the probability simplex over the n actions. This is typically called the "Expert" or "Hedge" setting[1]. The actions are sometime referred to as "experts". For this particular setting we define regret as

$$Regret = \sum_{t=1}^{T} p_t \cdot l_t - \min_{i \in \{1, \cdots, n\}} \sum_{t=1}^{T} l_t(i)$$

Claim 1. $\exists$ Algo : $Regret = O(\sqrt{T \log n})$.

In particular we will show that the Randomized Weighted Majority is such an algorithm (see Section 5). But before we prove Claim 1, we will show that the existence of such an algorithm leads to a simple proof of a well-known result from game theory called the Minimax Theorem.

4 Proof of the Minimax Theorem

A two player zero-sum game is defined by a matrix $M \in R^{n \times m}$. Player 1 chooses i, and Player 2 chooses j. The loss (gain) for P1 (P2) is M_{ij}. For example, Table 1 shows the payoff matrix M for the well known game rock, paper, scissors. Rows represent Player 1 and columns represent Player 2. Each entry is the cost of the outcome for Player 1 (e.g. Paper beats Rock so $M_{P,R}$ has a cost of -1 for Player 1).

The question posed by Von Neumann was: What if each player has to announce what their strategy is before they play? Intuitively, it seems that the person who can choose their strategy second has the advantage. However, one can show mathematically that it doesn't really matter. (i.e. min max is equal to max min)[2].

Theorem 1 (Von Neumann Minimax Theorem)**.**

$$\min_{p \in \Delta_n} \max_{q \in \Delta_m} \sum_{i,j} p(i) q(i) M_{ij} = \max_{q \in \Delta_m} \min_{p \in \Delta_n} \sum_{i,j} p(i) q(i) M_{ij}$$

[1]As a side note, there is a particular setting calling the "Bandit Setting" where the player only observes $l_t(I)$. Even in this scenario, we can still "do well".

[2]The original proof used Brouwer's fixed-point theorem.

Proof: Let's play repeatedly: I choose $p_t \in \Delta_n$ using an algorithm with vanishing regret. Then the opponent chooses $q_t \in \Delta_m$ to be the response that hurts me most. Define $l_t := Mq_t$. I know that by Claim 1, for any ϵ, it is possible to choose T large enough that the statement holds.

$$\frac{1}{T}\left(\sum_{t=1}^{T} p_t \cdot l_t - \min_{p\in\Delta_n}\sum_{t=1}^{T} p \cdot l_t\right) \leq \epsilon$$

This implies that

$$\begin{aligned}
\min_{p\in\Delta_n}\max_{q\in\Delta_m} p^T M q &\leq \frac{1}{T}\sum_{t=1}^{T} p_t M q_t \\
&= \frac{1}{T}\sum_{t=1}^{T} p_t \cdot l_t \\
&\leq \frac{1}{T}\min_{p\in\Delta_n}\sum_{t=1}^{T} p \cdot l_t + \epsilon \\
&\leq \max_{q\in\Delta_m}\min_{p\in\Delta_n} p^T M q + \epsilon
\end{aligned}$$

The cost of min max is bounded by max min plus ϵ, which we can set arbitrarily close to 0. $\square$

5 Randomized Weighted Majority - Hedge Algorithm

We now introduce the Randomized Weighted Majority algorithm. We define $L_t := \sum_{s=1}^{t} l_s$ to be the vector of cumulative losses of the experts at time t. The algorithm chooses an expert at time t according to the distribution p_t, where

$$w_t(i) := exp(-\eta \cdot L_t(i)) \qquad \text{Weight assigned to expert } i \text{ at time } t$$

$$p_t(i) := \frac{w_t(i)}{\sum_{j=1}^{n} w_t(j)} \qquad \text{Probability of choosing expert } i \text{ at time } t$$

Here $\eta > 0$ is a parameter of the algorithm.

Theorem 2. *The regret for the Randomized Weighted Majority Algorithm is* $O(\sqrt{T \log n})$.

The following are two small facts that will be useful for the proof:

1. For any x, $\log(1+x) \leq x$.[3]
2. For any $x \in [0,1]$, for any α, $e^{\alpha x} \leq (e^{\alpha}-1)x + 1$.

[3] We have seen this before, but stated slightly differently. It is equivalent to $1 + x \leq e^x$, which we used in Lecture 2 and in the problem set.

Proof Sketch: We define the following potential function, which we will use in the analysis:

$$\Phi_t = -\log \sum_{i=1}^{n} w_t(i) \tag{2}$$

We first compute the difference in the potential of any two iterations

$$\begin{aligned}
\Phi_{t+1} - \Phi_t &= -\log \left(\frac{\sum_{i=1}^{n} w_{t+1}(i)}{\sum_{i=1}^{n} w_t(i)} \right) \\
&= -\log \left(\frac{\sum_{i=1}^{n} w_t(i) exp(-\eta l_t(i))}{\sum_{i=1}^{n} w_t(i)} \right) \\
&= -\log \left(\sum_{i=1}^{n} p_t(i) exp(-\eta l_t(i)) \right) \\
&\geq -\log \left(\sum_{i=1}^{n} p_t(i)((e^{-\eta} - 1) l_t(i) + 1) \right) && \text{By trick 2} \\
&= -\log \left(1 + \sum_{i=1}^{n} p_t(i)(e^{-\eta} - 1) l_t(i) \right) \\
&\geq (1 - e^{-\eta}) p_t \cdot l_t && \text{By trick 1}
\end{aligned}$$

Using this, we can get an upper bound on the total loss of the algorithm. Let $i^* \in \{1, \cdots, n\}$ be the best performing expert. and let L^* be an upper bound on the loss of the best expert. (If we don't have any prior information about performance, we can set $L^* = T$.) We have

$$\begin{aligned}
\sum_{t=1}^{T} p_t \cdot l_t &\leq \frac{1}{1 - e^{-\eta}} (\Phi_{T+1} - \Phi_1) \\
&= \frac{1}{1 - e^{-\eta}} \left(\log n - \log \sum_{i=1}^{n} exp(-\eta L_t(i)) \right) \\
&\leq \frac{1}{1 - e^{-\eta}} \left(\log n - \log(exp(-\eta L_t(i^*))) \right) \\
&= \frac{1}{1 - e^{-\eta}} (\log n + \eta L_t(i^*)) \\
&\approx (1 + \eta) L_t(i^*) + \frac{\log n}{\eta} && \text{(we'll see this in the next lecture..)} \\
&= L_t(i^*) + \eta L_T(i^*) + \frac{\log n}{\eta} \\
&\leq L_t(i^*) + 2\sqrt{\log(n) L^*} && \text{where } L^* \text{ is bounded by } T
\end{aligned}$$

□

We obtain the bound on the regret by subtracting $L_t(i^*)$ on both sides since the left side becomes the regret and the right is the asymptotic expression we were looking for. This is the result of setting $\eta = \sqrt{\log n / L^*}$. In the next lecture, we'll finish the steps of this proof that we skipped.