Notes on Online Learning for CIS 625

Jacob Abernethy

March 30, 2012

1 Randomized Weighted Majority

We're hoping to predict a sequence of bits $y_1, y_2, y_3, \ldots \in \{0, 1\}$. The algorithm we design will make predictions $\hat{y}_1, \hat{y}_2, \hat{y}_3, \ldots \in [0, 1]$. To measure performance, we have a loss function $\ell(\hat{y}, y)$ which takes values in $[0, 1]$ and is convex in its first argument. Two standard loss functions are the absolute loss $|\hat{y} - y|$ and the squared loss $(\hat{y} - y)^2$.

Let us imagine we have a set of n "experts" each of which gives us a prediction on every round. Let's say that expert i say $e_{i,t} \in [0, 1]$ on round t. Now we want to choose $\hat{y}_t$ using our received advice $(e_{1,t}, \ldots, e_{n,t})$ and the past performance of the experts. How shall we do this?

Here's where we use the "Randomized Weighted Majority" algorithm. Let's define the cumulative loss of expert i as

$$L_{i,t} := \sum_{s=1}^{t-1} \ell(e_{i,s}, y_s).$$

Now we can define a "weight" for each expert. Assume we have some parameter $\eta > 0$, then let

$$w_{i,t} := \exp(-\eta L_{i,t}).$$

Let us now use these weights as a level of "trust" in each expert. So our prediction will be a weighted average of the experts' predictions, but the weights will decay with the past performance or the expert. Precisely, we will set

$$\hat{y}_t := \frac{\sum_{i=1}^n w_{i,t} e_{i,t}}{\sum_{i=1}^n w_{i,t}}$$

We now state the big theorem. A good writeup of this result with proof can be found here: `http://goo.gl/m9jR6`.

Theorem 1. *Let $\eta > 0$ and T be arbitrary, and let i^* be the "best expert" at time T, i.e. the i acheiving the minimum cumulative loss $L_{i,T}$. Then we can bound the loss of the algorithm as*

$$L_T := \sum_{t=1}^{T} \ell(\hat{y}_t, y_t) \leq \frac{\ln n + \eta L_{i^*,T}}{1 - \exp(-\eta)}.$$

This bound can be a little hard to interpret, since it has the "learning rate" parameter η in a number of places. The bound becomes a lot nicer once we tune η correctly. I'll spare the details on how to do this, and just mention that this requires having a known bound $\tilde{L}$ on the loss of the best expert $L_{i^*,t}$.

Corollary 1. *If $\tilde{L} \geq L_{i^*,T}$, if we set*

$$\eta = \ln\left(1 + \sqrt{\frac{2 \ln n}{\tilde{L}}}\right)$$

then the bound in the previous theorem becomes

$$L_T \leq L_{i^*,T} + \sqrt{2\tilde{L} \ln n} + \ln n$$

Notice here that our algorithm is performing essentially as well as the best expert in hindsight. Our "regret to the best" is only a constant $\ln n$ plus a square-root term $O(\sqrt{\tilde{L}})$.

2 The (almost identical) Hedge Algorithm

After some initial work in online learning for combining the predictions of experts, it became quite clear that the "weighted majority" trick is actually more general than was originally thought. For the moment, let's forget about experts and think instead about "actions" that we can take. On each round t of a repeated game, we must select a distribution $\mathbf{p}_t$ over these n actions and then, once we have committed to this distribution, we observe a "loss vector" $\boldsymbol{\ell}_t := (\ell_{1,t}, \ldots, \ell_{n,t})$, where $\ell_{i,t}$ is the loss of having chosen action i on round t. The original paper of Freund and Schapire (`http://goo.gl/LiOHb`) imagined a gambler betting on a repeated horse race, so action i would be placing a bet on the ith horse. Since the gambler would want to "hedge" his bets, the algorithm I will now show you became known as the Hedge Algorithm.

Freund and Schapire noticed that we can use the same weighed majority strategy for betting on horses. We can redifine the cumulative loss of an action as simply

$$L_{i,t} := \sum_{s=1}^{t-1} \ell_{i,s}.$$

In other words, if I had just been betting on the ith horse all along, $L_{i,t}$ is how much I would have lost up to time T. With this notion of cumulative loss, the weight of an action is the same

$$w_{i,t} := \exp(-\eta L_{i,t}).$$

Now the gambler must choose a distribution over the n actions, and he shall choose the weighted majority distribution which we've already seen:

$$\mathbf{p}_t := \left(\frac{w_{1,t}}{\sum_{i=1}^n w_{i,t}}, \ldots, \frac{w_{n,t}}{\sum_{i=1}^n w_{i,t}} \right)$$

The cumulative loss of the gambler after T rounds is just his total expected loss had he sampled i_t from $\mathbf{p}_t$ and suffered $\ell_{i_t,t}$. That is

$$L_T := \sum_{t=1}^{T} \mathbf{p}_t \cdot \boldsymbol{\ell}_t.$$

It's worth noting here that the gambler does not necessarily have to play in a randomized fashion. Randomness is necessary in some scenarios, as a gambler may only be able to place a single bet and hence will randomize hoping to do will in expectation. On the other hand, on each round the gambler could simply spread his bet over all n actions according to the distribution $\mathbf{p}_t$, which would be more like a "portfolio" strategy. But because we are looking at linear loss here, in each case the quantity we care about is still $\mathbf{p}_t \cdot \boldsymbol{\ell}_t$ on each round.

Theorem 2. *The cumulative loss of the Hedge Algorithm at time T with parameter $\eta > 0$ satisfies the same bound as Theorem 1; namely,*

$$L_T = \sum_{t=1}^{T} \mathbf{p}_t \cdot \boldsymbol{\ell}_t \leq \frac{\ln n + \eta L_{i^*,t}}{1 - \exp(-\eta)}.$$

Given a bound $\tilde{L} \geq L_{i^*,T}$, and with η tuned appropriately, we have that

$$L_T \leq L_{i^*,T} + \sqrt{2\tilde{L} \ln n} + \ln n$$

3 No Regret and the Minimax Theorem

Such online learning methods are typically referred to as "no-regret" algorithms, for the following reason.

$$\text{As } T \to \infty \qquad \frac{L_T}{T} \to \min_{i=1,\dots,n} \frac{L_{i,T}}{T}.$$

To see why this is true, notice that we need to simply divide both sides of the Hedge regret bound by T. Notice that $\tilde{L}$, which is used to tune η, can always be set as T, since $L_{i,T} \leq T$. Then, it's clear that the average regret $\frac{\sqrt{2T \ln n} + \ln n}{T} \to 0$ as $T \to \infty$. This is why we say "no-regret". We often write this statement in the following way:

$$\frac{L_T}{T} = \min_{i=1,\dots,n} \frac{L_{i,T}}{T} + o(1).$$

Let us ponder the above observation. We we now have is that the performance of our learning algorithm, on average, is essentially *no worse than if it had known the best expert/action in hindsight!* This is kind of a striking fact, given than we made no assumptions on the process generating the sequence of losses (or predictions $e_{i,t}$, or the outcomes y_t) – these could have even been generated by an adversary.

What is quite surprising is that the existence of a no-regret algorithm gives us a simple way to prove the minimax theorem of von Neumann. Here we use the notation Δ_n as the n-dim probability simplex.

Theorem 3. *Let $M \in [0,1]^{n \times m}$ be a payoff matrix for a 2-player zero-sum game. Then we have*

$$\min_{\mathbf{p} \in \Delta_n} \max_{\mathbf{q} \in \Delta_m} \mathbf{p}^\top M \mathbf{q} = \max_{\mathbf{q} \in \Delta_m} \min_{\mathbf{p} \in \Delta_n} \mathbf{p}^\top M \mathbf{q}$$

Proof. Let $v_1 = \min_{\mathbf{p} \in \Delta_n} \max_{\mathbf{q} \in \Delta_m} \mathbf{p}^\top M \mathbf{q}$ and let $v_2 = \max_{\mathbf{q} \in \Delta_m} \min_{\mathbf{p} \in \Delta_n} \mathbf{p}^\top M \mathbf{q}$. It is easy to show that $v_1 \geq v_2$ (this is known as "weak duality"). The easy proof of this inequality is just to see that minimizing player choosing $\mathbf{p}$ would rather play 2nd, hence can achieve a smaller value when minimizing *within* the $\max_{\mathbf{q}}$ objective. The more technical proof is to note that, if we take the optimal $\mathbf{p}^*$ for v_1, then we

$$\min_{\mathbf{p} \in \Delta_n} \max_{\mathbf{q} \in \Delta_m} \mathbf{p}^\top M \mathbf{q} = \max_{\mathbf{q} \in \Delta_m} \mathbf{p}^{*\top} M \mathbf{q} \geq \max_{\mathbf{q} \in \Delta_m} \min_{\mathbf{p} \in \Delta_n} \mathbf{p}^\top M \mathbf{q}$$

where the inequality holds because $\mathbf{p}^*$ may not be the optimal choice for $\mathbf{p}$ when chosen as a function of $\mathbf{q}$ (i.e. when the min player gets to go 2nd).

Now we prove "strong duality", $v_1 \leq v_2$. To achieve this, we take an odd detour and imagine a repeated game where, on each round t, the minimizing player chooses a distribution $\mathbf{p}_t$ (to be defined soon) having learned from the past observations. The maximizing player gets to see this $\mathbf{p}_t$ and in response chooses $\mathbf{q}_t := \arg\max_{\mathbf{q} \in \Delta_m} \mathbf{p}_t^\top M \mathbf{q}$. Notice that since $\mathbf{p}_t$ was possibly not chosen optimally, we have that for every t,

$$\min_{\mathbf{p} \in \Delta_n} \max_{\mathbf{q} \in \Delta_m} \mathbf{p}^\top M \mathbf{q} \leq \max_{\mathbf{q} \in \Delta_m} \mathbf{p}_t^\top M \mathbf{q} = \mathbf{p}_t^\top M \mathbf{q}_t.$$

Now how do we choose $\mathbf{p}_t$? Let's use the Hedge Algorithm! We'll define the loss vectors to be $\boldsymbol{\ell}_t := M \mathbf{q}_t$ which is natural since $\ell_{i,t}$ is the cost of choosing action i for the minimizing player given that the maximizing player chose $\mathbf{q}_t$. Now let's use our no-regret statement to control how much cost the minimizing player suffered on average:

$$\begin{aligned}
\frac{1}{T} \sum_{t=1}^{T} \mathbf{p}_t^\top M \mathbf{q}_t &= \frac{1}{T} \mathbf{p}_t \cdot \boldsymbol{\ell}_t \\
\text{(no-regret)} \quad &\leq \frac{1}{T} \min_i L_{i,T} + o(1) \\
&= \frac{1}{T} \min_{\mathbf{p} \in \Delta_n} \mathbf{p} \cdot \left(\sum_{t=1}^{T} \boldsymbol{\ell}_t \right) + o(1) \\
&= \min_{\mathbf{p} \in \Delta_n} \mathbf{p}^\top M \left(\frac{1}{T} \sum_{t=1}^{T} \mathbf{q}_t \right) + o(1)
\end{aligned}$$

Notice now that $\hat{\mathbf{q}} := \frac{1}{T}\sum_{t=1}^{T}\mathbf{q}_t$ is a distribution, but it is probably not the optimal distribution for the maximizing player! Hence we have

$$\min_{\mathbf{p}\in\Delta_n} \mathbf{p}^\top M\left(\frac{1}{T}\sum_{t=1}^{T}\mathbf{q}_t\right) + o(1) \leq \max_{\mathbf{q}\in\Delta_m}\min_{\mathbf{p}\in\Delta_n} \mathbf{p}^\top M\mathbf{q} + o(1).$$

What we have just shown is that

$$v_1 = \min_{\mathbf{p}\in\Delta_n}\max_{\mathbf{q}\in\Delta_m} \mathbf{p}^\top M\mathbf{q} \leq \frac{1}{T}\sum_{t=1}^{T}\mathbf{p}_t^\top M\mathbf{q}_t \leq \max_{\mathbf{q}\in\Delta_m}\min_{\mathbf{p}\in\Delta_n} \mathbf{p}^\top M\mathbf{q} + o(1) = v_2 + o(1).$$

Since T can be chosen in order that the $o(1)$ term is arbitrarily small, we have that $v_1 \leq v_2$ and we are done. □