CS 7545: Machine Learning Theory **Fall 2019**

Lecture 15: Follow The Regularized Leader, Multi-Armed Bandit & EXP3

Lecturer: Jacob Abernethy *Scribes: Sam Waters, Satchit Sivakumar*

15.1 Follow The Regularized Leader

The final algorithm described in the class under the purview of Online Convex Optimization is Follow The Regularized Leader. The algorithm takes the following form:

1: **Inputs:** Regularizer $R(x)$, Convex Set K, initial point x_1
2: **for** $t = 1,..,T$ **do**
3: $x_{t+1} = \underset{x \in K}{\operatorname{argmin}} \, \eta \sum_{s=1}^{t} f_s(x) + R(x)$ OR $\underset{x \in K}{\operatorname{argmin}} \, \eta \sum_{s=1}^{t} \langle \nabla f_s(x_s), x \rangle + R(x)$
4: **end for**

Algorithm 1: Follow The Regularized Leader

As described above, there are 2 different versions of the algorithm. For the second version, we have the following Theorem,

Theorem 15.1 *Suppose $x \in int(K)$, then the updating rule*

$$x_{t+1} = \underset{x \in K}{\operatorname{argmin}} \, \eta \sum_{s=1}^{t} \langle \nabla f_s(x_s), x \rangle + R(x)$$

is equivalent to the Online Mirror Descent discussed in Lecture 14

$$x_{t+1} = \underset{x \in K}{\operatorname{argmin}} \, \eta \langle \nabla f_t(x_t), x \rangle + D_R(x, x_t)$$

.

To give some intuition about why this algorithm might be good, the professor showed that as long as x is an interior point, then version 2 of the above algorithm is equivalent to Online Mirror Descent discussed in Lecture 14.

Proof: Now, we know that for any point in the interior of the convex set, the minimum of a convex function f(x) over this set will occur at this point if and only if $\nabla f(x) = \vec{0}$.

Now, let $\phi(x) = \eta \sum \langle \nabla f_s(x_s), x \rangle + R(x)$. Then, taking the gradient and setting to $\vec{0}$ and substituting $x = x_{t+1}$ we can write:

$$-\eta \sum_{s=1}^{t} \nabla f_s(x_s) = \nabla R(x_{t+1}) \tag{15.1}$$

Now, in Online Mirror Descent, we find the minimum of the following function every iteration: $\phi^1_{t+1}(x) = \eta \langle \nabla f_t(x_t), x \rangle + D_R(x, x_t)$. Now, we will take the gradient of this function and set it to $\vec{0}$. We get:

$$\nabla \phi^1_{t+1} = \eta \nabla f_t(x_t) + \nabla \Big(R(x) - R(x_t) - \langle \nabla R(x_t), x - x_t \rangle \Big) = 0 \tag{15.2}$$

Simplifying, we get:

$$\eta \nabla f_t(x_t) + \nabla R(x) - \nabla R(x_t) = 0 \tag{15.3}$$

Rearranging the terms and substituting $x = x_{t+1}$ we get:

$$\nabla R(x_{t+1}) = -\eta \nabla f_t(x_t) + \nabla R(x_t) \tag{15.4}$$

It is easy to see that equation (15.4) is just a recursive representation of equation (15.1). Hence, the value of x at which both $\phi(x)$ and $\phi^1(x)$ attain their minimum each iteration is the same and the algorithms are equivalent.

■

15.2 Multi-Armed Bandits

The bandit setting is in contrast to the full information setting where the entire loss vector is obtained. Rather, in the bandit setting, feedback for an action is limited only to the selected action. This is formalized as follows:

- We assume we are given n actions.
- At every iteration the algorithm selects a time varying probability distribution over actions denoted by $p^t \in \Delta_n$.
- Now, nature responds by choosing a loss vector $l^t \in [0,1]^n$.
- Now, the algorithm samples an action i_t according to p^t and observes only the loss corresponding to this action i.e. $l^t_{i_t}$. The procedure is now repeated.

In this setting, we define regret as follows:

Definition 15.2 (Regret in bandit setting) *The regret in the bandit setting is defined as:*

$$\text{Regret}_T = \sum_{t=1}^T l^t_{i_t} - \min_i \sum_{t=1}^T l^t_i$$

Now, it is clear from the definition that Regret_T is a random variable. To make this more tractable, we will assume the loss vector at each stage is independent of the probability distribution chosen over actions Hence, we consider expected regret defined as follows:

Definition 15.3 (Expected Regret) *The expected regret is defined as:*

$$\text{E}[\text{Regret}_T] = \text{E}[\sum_{t=1}^T p^t \cdot l^t - \min_i \sum_{t=1}^T l^t_i]$$

where the expectation is taken over all the randomness in the algorithm.

15.3 EXP3 Algorithm and Analysis

Now, we describe an algorithm to achieve good regret bounds in this setting. Intuitively, the structure of the algorithm looks a lot like the exponential weights algorithm for the hedge setting discussed in an earlier lecture, with the following trick: Since we don't know the full loss vector, we instead use an unbiased estimator for the loss vector- we choose a vector $\hat{l}$ with all 0s except at the i^{th} position where i represents

the action sampled from the probability distribution. The i^{th} coordinate is $\frac{l^t_{i_t}}{p^t_{i_t}}$. A simple calculation shows that this is an unbiased estimator of the loss vector:

$$E_{i_t}[\hat{l}] = \sum_{i=1}^{N} p_i^t \cdot \left[0, ..., 0, \frac{l^t_{i_t}}{p^t_i}, 0, ..., 0\right] = l^t \tag{15.5}$$

Now, in order to show that this trick works and derive the regret bounds for the EXP3 algorithm, we prove the following Lemma which is a variant of a similar lemma proved for the Exponential Weights algorithm:

Lemma 15.4 *Let x be a random variable that only takes non-negative values. Then,* $\log \mathrm{E}[e^{-sx}] \leq -s\mathrm{E}[x] + \frac{s^2}{2}\mathrm{E}[x^2]$

Before proving the lemma, we will state 2 other elementary lemmas without proof that are important in proving the above lemma.

Lemma 15.5 $\log(1+x) \leq x$

Lemma 15.6 *For all $x > 0$, $e^{-sx} \leq -sx + \frac{s^2}{2}x^2 + 1$*

The second lemma is a simple consequence of the Taylor series expansion of e^{-sx}. We will now prove Lemma 15.4.

Proof: Using lemma 15.6 we can write the following equation:

$$\log \mathrm{E}[e^{-sx}] \leq \log \mathrm{E}[1 - sx + \frac{s^2}{2}x^2] \tag{15.6}$$

Simplifying, we get:

$$\log \mathrm{E}[e^{-sx}] \leq \log\left(1 - \mathrm{E}[sx - \frac{s^2}{2}x^2]\right) \tag{15.7}$$

Now, using Lemma 15.5, we get:

$$\log \mathrm{E}[e^{-sx}] \leq -\mathrm{E}[sx - \frac{s^2}{2}x^2] = -s\mathrm{E}[x] + \frac{s^2}{2}\mathrm{E}[x^2] \tag{15.8}$$

■

We can now formally describe the EXP3 algorithm and prove its regret bounds. The algorithm is given below following which we state and prove a theorem that bounds its regret:

1: Initialize $w_1^1, w_2^1, ...w_n^1$
2: **for** $t = 1,..,\mathrm{T}$ **do**
3: $\quad p^t = \vec{w}^t / ||\vec{w}^t||_1$
4: $\quad$ sample $i_t \sim p^t$, observe $\ell^t_{i_t}$
5: $\quad \vec{\hat{\ell}}^t = [0, 0, ...0, \ell^t_{i_t}/p^t_{i_t}, 0, ...0]$
6: $\quad \forall i : w_i^{t+1} = w_i^t \exp\left(-\eta \hat{\ell}_i^t\right)$
7: **end for**

Algorithm 2: EXP3 Algorithm

Theorem 15.7 $\forall i$: $\mathrm{E}[\sum_{t=1}^{T} p^t \ell^t - \min_{i\in[n]} \sum_{t=1}^{T} \ell_i^t] \leq \frac{\log n}{\eta} + \frac{\eta}{2}Tn$

Proof: Let $\phi_t = -\frac{1}{\eta}\log(\sum_{i=1}^n w_i^t)$. Notice :

$$\phi_{t+1} - \phi_t = -\frac{1}{\eta}\log(\frac{\sum_{i=1}^n w_i^{t+1}}{\sum_{i=1}^n w_i^t}) \tag{15.9}$$

$$= -\frac{1}{\eta}\log(\frac{\sum_{i=1}^n w_i^t \exp(-\eta\hat{\ell}_i^t)}{\sum_{i=1}^n w_i^t}) \tag{15.10}$$

Let x be a random variable taking value $\hat{\ell}_i^t$ with probability $p_i^t = \frac{w_i^t}{\sum_{j=1}^n w_j^t}$:

$$= -\frac{1}{\eta}\log(\sum_{i=1}^n p_i^t \exp(-\eta\hat{\ell}_i^t)) \tag{15.11}$$

$$= -\frac{1}{\eta}\log \mathop{\mathrm{E}}_x[\exp(-\eta x)] \tag{15.12}$$

By lemma 15.4:

$$\geq -\frac{1}{\eta}(-\eta \mathop{\mathrm{E}}_x[x] + \frac{\eta^2}{2}\mathop{\mathrm{E}}_x[x^2]) \tag{15.13}$$

Because $\mathrm{E}[x] = \vec{p^t}\cdot\vec{\hat{\ell}^t}$:

$$= \vec{p^t}\cdot\vec{\hat{\ell}^t} - \frac{\eta}{2}\sum_{i=1}^n p_i^t(\hat{\ell}_i^t)^2 \tag{15.14}$$

Recall $\mathrm{E}_{i_t\sim p_t}[\vec{\hat{\ell}^t}] = \vec{\ell^t}$:

$$\mathop{\mathrm{E}}_{i_t\sim p^t}[\phi_{t+1} - \phi_t | i_1 \dots i_{t-1}] \geq \mathop{\mathrm{E}}_{i_t\sim p^t}[\vec{p^t}\cdot\vec{\hat{\ell}^t} - \frac{\eta}{2}\sum_{i=1}^n p_i^t(\hat{\ell}_i^t)^2 | i_1 \dots i_{t-1}] \tag{15.15}$$

$$= \vec{p^t}\cdot\vec{\ell^t} - \frac{\eta}{2}\mathop{\mathrm{E}}_{i_t\sim p^t}[\sum_{i=1}^n p_i^t(\hat{\ell}_i^t)^2 | i_1 \dots i_{t-1}] \tag{15.16}$$

Because $\vec{\hat{\ell}^t}$ is a function of $i_1 \dots i_{t-1}$ and $\hat{\ell}_j^t = 0$ for all $j \neq i$:

$$= \vec{p^t}\cdot\vec{\ell^t} - \frac{\eta}{2}\mathop{\mathrm{E}}_{i_t\sim p^t}[p_i^t(\hat{\ell}_i^t)^2] \tag{15.17}$$

$$= \vec{p^t}\cdot\vec{\ell^t} - \frac{\eta}{2}\sum_{i=1}^n p_i^t(p_i^t(\hat{\ell}_i^t)^2) \tag{15.18}$$

$$= \vec{p^t}\cdot\vec{\ell^t} - \frac{\eta}{2}\sum_{i=1}^n p_i^t(p_i^t(\frac{\ell_i^t}{p_i^t})^2) \tag{15.19}$$

$$\geq \vec{p^t}\cdot\vec{\ell^t} - \frac{\eta}{2}n \tag{15.20}$$

By the law of total expectation:

$$\underset{i_t \sim p^t}{\mathrm{E}}[\phi_{T+1} - \phi_1] = \underset{i_t \sim p^t}{\mathrm{E}}[\sum_{t=1}^{T} \phi_{t+1} - \phi_t] \tag{15.21}$$

$$= \underset{i_t \sim p^t}{\mathrm{E}}[\sum_{t=1}^{T} \underset{i_t \sim p^t}{\mathrm{E}}[\phi_{t+1} - \phi_t | i_1 \dots i_{T-1}]] \tag{15.22}$$

$$\geq \underset{i_t \sim p^t}{\mathrm{E}}[\sum_{t=1}^{T} p^t \ell^t] - \frac{\eta}{2} nT \tag{15.23}$$

$$= \mathrm{E}[L_T(\mathrm{EXP3})] - \frac{\eta}{2} nT \tag{15.24}$$

At time $t = T + 1$:

$$w_i^{T+1} = \exp\left(-\eta \sum_{i=1}^{T} \hat{\ell}_i^t\right) \tag{15.25}$$

$$\phi_{T+1} = -\frac{1}{\eta} \log(\sum_{i=1}^{n} w_i^{T+1}) \tag{15.26}$$

$$\leq -\frac{1}{\eta} \log(w_i^{T+1}) \tag{15.27}$$

$$= -\frac{1}{\eta} \log(\exp\left(-\eta \sum_{i=1}^{T} \hat{\ell}_i^t\right)) \tag{15.28}$$

$$= \sum_{i=1}^{T} \hat{\ell}_i^t \tag{15.29}$$

$$\implies \underset{i_t \sim p^t}{\mathrm{E}}[\phi_{T+1} - \phi_1] \leq \sum_{i=1}^{T} \hat{\ell}_i^t - \phi_1 \leq \sum_{i=1}^{T} \hat{\ell}_i^t + \frac{\log n}{\eta}, \tag{15.30}$$

provided that

$$\phi_1 = -\frac{\log n}{\eta}. \tag{15.31}$$

Therefore:

$$\mathrm{E}[L_T(\mathrm{EXP3})] - \frac{\eta}{2} nT \leq \underset{i_t \sim p^t}{\mathrm{E}}[\phi_{T+1} - \phi_1] \leq \sum_{i=1}^{T} \hat{\ell}_i^t + \frac{\log n}{\eta} \tag{15.32}$$

$$\implies \mathrm{E}[L_T(\mathrm{EXP3})] - \sum_{i=1}^{T} \hat{\ell}_i^t \leq \frac{\eta}{2} nT + \frac{\log n}{\eta} \tag{15.33}$$

$$\implies \mathrm{Regret}_T(\mathrm{EXP3}) \leq \frac{\eta}{2} nT + \frac{\log n}{\eta} \tag{15.34}$$

■

Corollary 15.8 *For* $\eta = \sqrt{\frac{2 \log n}{nT}}$, $\mathrm{E}[\mathrm{Regret}_T] \leq \sqrt{2Tn \log n}$.

Notice that this bound is very similar to the regret bounds for other algorithms such as the exponential weights algorithm, except that $\mathrm{L}_t(i^*)$ is replaced with Tn. This is because the loss of the best expert i^* is bounded by T, and because it is now necessary to observe each of the n possible actions in order to accurately understand the regret.