CS6781 - Theoretical Foundations of Machine Learning

Lecture 14, 16: Follow The Regularized/Perturbed Leader

March 10, 2020

Lecturer: Nika Haghtalab *Readings: N/A*
Scribe: Kate Donahue, Ziyang Wu, and Jonathan Chang

1 Review

Last time, we discussed the statistical and algorithmic aspects of online learning. We investigated how algorithms that perform well offline fare in online settings. We found that ERM does not do well in the online world and indeed any deterministic algorithm performs poorly as well. This is due to the fact that OPT (best solution) kept switching from one time step to another. Today, we will look at the algorithmic implications of this and provide an online no-regret algorithm that is very similar to ERM.

Let us start by recalling a lemma we saw in the last lecture.

Lemma 1.1. *Let x^t to be the ERM performed on the historical data up to the current time step, i.e., $x^t \leftarrow \arg\min_x \sum_{\tau=1}^{t-1} c^\tau(x)$. Then,*

$$\textit{Regret} \leq \sum_{t=1}^{T} \left[c^t(x^t) - c^t(x^{t+1})\right]$$

We proved this in class last week. Note that $c^t(x^t) - c^t(x^{t+1})$ is 0 whenever $x^t = x^{t+1}$ (ERM doesn't switch). Last time, we showed that:

$$\sum_{t=1}^{T} c^t(x^{t+1}) \leq \min_x \sum_{t=1}^{T} c^t(x)$$

by induction. Once we have this, the result is immediate, because

$$\text{Regret} = \sum_{t=1}^{T} \left[c^t(x^t)\right] - \min_x \sum_{t=1}^{T} \left[c^t(x)\right] \leq \sum_{t=1}^{T} \left[c^t(x^t) - c^t(x^{t+1})\right]$$

The implication is that if ERM is stable (x^{t+1} and x^t switch at most k times, or $c^t(x) - c^t(x^{t+1})$ is positive only in k time steps), then regret $\leq k$.

2 Follow the Regularized Leader

In this section, we see how we can stabilize ERM so as to achieve a no-regret algorithm. We start by considering ERM with one small change, we add to the history a new cost function $c^0(\cdot)$. This cost function is often called a *regularizer*. We refer to ERM with a regularizer as Follow the Regularized Leader (FTRL) algorithm. See Algorithm 1 for a description.

Algorithm 1 Follow the Regularized Leader

$R(\cdot)$ regularizer
$x^1 = \text{argmin}_x R(x)$
for $t = 2, 3, 4, ...T$ **do**
    $x^t = \text{argmin}_{x\in\mathcal{X}} R(x) + \sum_{\tau=1}^{t-1} c^\tau(x)$
end for

Theorem 2.1 (FTRL Theorem)**.** *Let x^t be the action played by the FTRL algorithm (Algorithm 1) at time t. Then, for any $x^* \in \mathcal{X}$,*

$$\sum_{t=1}^{T} c^t(x^t) - \sum_{t=1}^{T} c^t(x^*) \leq \sum_{t=1}^{T} \left[c^t(x^t) - c^t(x^{t+1})\right] + \left[R(x^*) - R(x^1)\right]$$

In other words, if you regularize your ERM, the regret is the same as in Lemma 1.1, but includes the difference term for the regularizer.

Proof. In order to prove this, we will follow the same steps as we did before, but we will include the cost on a the 0th day, $c^0(x) = R(x)$. FTRL is equal to ERM but on this altered history. By Lemma 1.1 we have.

$$\sum_{t=0}^{T} c^t(x^t) - c^t(x^*) \leq \sum_{t=0}^{T} c^t(x^t) - c^t(x^{t+1}).$$

$$c^0(x^0) - c^0(x^*) + \sum_{t=1}^{T} c^t(x^t) - c^t(x^*) \leq c^0(x^0) - c^0(x^1) + \sum_{t=1}^{T} c^t(x^t) - c^t(x^{t+1})$$

Cancelling out the $c^0(x^0)$ and bringing across the terms gives us the result we want. $\square$

To see the implications of the FTRL theorem, let us assume that the cost functions we are dealing with are linear. That is, $c^t(\mathbf{x}) = \mathbf{c}^t \cdot \mathbf{x}$. Then our hope is that the choice of regularizer induces large enough cost so that the choice of $\mathbf{x}^t$ and its loss changes slowly, i.e., $c^t(\mathbf{x}^t) - c^t(\mathbf{x}^{t+1}) \leq \epsilon$, but also the regularizer is small enough so that $R(\mathbf{x}^*) - R(\mathbf{x}^1)$ is also small. In that case the regret $\leq \epsilon \cdot T + \max_{\mathbf{x}} R(\mathbf{x})$ is hopefully small. So, we need to choose a regularizer that has these properties.

Theorem 2.2. *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be $\mathcal{X} = \{\mathbf{x} \mid \|\mathbf{x}\|_2 \leq 1\}$ and assume that the cost functions on each step are linear and $\|\mathbf{c}\|_2 \leq 1$. Consider the FTRL algorithm with the following regularizer*

$$R(\mathbf{x}) = \sqrt{\frac{T}{2}}\|\mathbf{x}\|_2^2.$$

Then the regret of FTRL is at most $\sqrt{2T}$.

Proof. Note that $\mathbf{x}^1 = \mathbf{0}$ is the minimizer of the regularizer. For all other time steps we have

$$\mathbf{x}^{t+1} = \text{argmin}_{\mathbf{x}}\sqrt{\frac{T}{2}}\|\mathbf{x}\|_2^2 + \sum_{\tau=1}^{t}\mathbf{c}^\tau \cdot \mathbf{x}.$$

We find this minimum by taking the gradient and setting it to be equal to 0 as follows.

$$2\sqrt{\frac{T}{2}} \cdot \mathbf{x} + \sum_{\tau=1}^{t}\mathbf{c}^\tau = 0$$

$$\mathbf{x}^{t+1} = -\frac{1}{\sqrt{2T}}\sum_{\tau=1}^{t}\mathbf{c}^\tau$$

If we had done this at the time step before, we would have gotten:

$$\mathbf{x}^{t} = -\frac{1}{\sqrt{2T}}\sum_{\tau=1}^{t-1}\mathbf{c}^\tau.$$

So

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \frac{1}{\sqrt{2T}}\mathbf{c}^t.$$

This is online gradient descent! It also shows that $\mathbf{c}^t \cdot \mathbf{x}^t - \mathbf{c}^t \cdot \mathbf{x}^{t+1}$ changes slowly. Then,

$$\begin{aligned}
\text{Regret} &\leq \sum_{t=1}^{T}\mathbf{c}^t \cdot (\mathbf{x}^t - \mathbf{x}^{t+1}) + R(\mathbf{x}^*) - R(\mathbf{0}) \\
&\leq \sum_{t=1}^{T}\left(\mathbf{c}^t \cdot \frac{1}{\sqrt{2T}} \cdot \mathbf{c}^t\right) + \sqrt{\frac{T}{2}}\|\mathbf{x}^*\|_2^2 \\
&\leq \sum_{t=1}^{T}\frac{1}{\sqrt{2T}} + \sqrt{\frac{T}{2}} \\
&= \sqrt{\frac{T}{2}} + \sqrt{\frac{T}{2}} \leq \sqrt{2T}
\end{aligned}$$

$\square$

Note that in the above application of FTRL, it's possible that we play $\mathbf{x}^t \notin \mathcal{X}$, i.e., it is possible that $\|\mathbf{x}^t\| > 1$. Can we get similar guarantees as in Theorem 2.2 if we limited FTRL to play within $\mathcal{X}$? Yes, when $\mathcal{X}$ is a convex body as is in Theorem 2.2. In that case, at every step of the optimization, let $\hat{\mathbf{x}}^t$ be what FTRL suggests, and let $\mathbf{x}^t \in \mathcal{X}$ be the closest point to $\hat{\mathbf{x}}^t$. That is $\mathbf{x}^t \in \mathcal{X}$ is a projection of $\hat{\mathbf{x}}^t$ on the convex set $\mathcal{X}$. Note that the distance between two points after projection on a convex body is only smaller than before. That is,

$$\|\mathbf{x}^t - \mathbf{x}^{t+1}\| \leq \|\hat{\mathbf{x}}^t - \hat{\mathbf{x}}^{t+1}\|.$$

So the stability property maintained by FTRL still holds here after projection.

3 Follow the Perturbed Leader

Let's consider an online routing game, where every day we decide what route to take from home to work. There is a graph $G = (V, E)$ where the domain of our actions are valid paths in G, shown by the set $\mathcal{X} \subseteq \{0,1\}^E$. Our cost function is a vector $\mathbf{c}$ where entry c_i is the traffic or delay on edge i. When taking route $\mathbf{x}$ our total delay is $\mathbf{c} \cdot \mathbf{x}$. While this is a linear cost function, our domain set $\mathcal{X}$ is not convex. Even though this problem is not a convex optimization problem, we can still solve the ERM efficiently in time $poly(|E|)$ by using Dijkstra's algorithm. In this section, we ask whether for linear functions we can turn any efficient ERM into a no-regret algorithm that is also efficient, even if the problem is non-convex?

Let us consider the case of expert's advice. We have $\mathcal{K}$ be the simplex of n dimensions (all probability distributions over all experts): $\{\mathbf{x} : \ x_i \geq 0 \ \forall i, \ \sum_{j=1}^n x_j = 1\}$. Set $\mathbf{c}$ to be such that c_i is the cost of expert i, which is between 0 and 1. Theorem 2.2 suggests that if we pick a strongly convex cost of $\sqrt{T/2}\|\mathbf{x}\|_2^2$ then we can get a no-regret algorithm. While this is true, note that $\sqrt{T/2}\|\mathbf{x}\|_2^2$ cannot be interpreted as cost of experts at a time step, simply because it is not linear. So, we ask whether we can use other methods of providing regularization that lead to $c^0(\mathbf{x}) = R(\mathbf{x})$ referring to the cost of the experts on time step 0?

The following algorithm, called Follow the Perturbed Leader (FTPL), achieves this exactly. It takes a cost function c^0 that assigns random costs to each expert at time 0. Then, at every time steps it picks the expert whose cumulative cost including step 0 is minimized.

Algorithm 2 Follow the Perturbed Leader

for $t = 1, 2, 3, 4, \ldots T$ **do**
  $\mathbf{c}^0 \sim$ distribution
  $\mathbf{x}^t = \text{argmin}_{\mathbf{x}} c^0(\mathbf{x}) + \sum_{\tau=1}^{t-1} c^\tau(\mathbf{x})$
end for

For an adaptive adversary, it is important that we re-draw $\mathbf{c}^0$ at every step to preserve the randomness of our algorithm. But for the oblivious adversary we could take $\mathbf{c}^0$ once at the beginning and reuse the same cost throughout. The expected regret of both algorithms is the same.

Theorem 3.1. *Let $\mathcal{X} \subset \mathbb{R}^n$ be any set (not-necessarily convex) such that $\max_{\mathbf{x},\mathbf{x}'} \|\mathbf{x} - \mathbf{x}'\|_1 \leq D$. Assume that the cost functions are such that $\|\mathbf{c}^t\|_1 \leq A$. Furthermore, assume that for any $\mathbf{x}$ and $\mathbf{x}$, $|\mathbf{c} \cdot \mathbf{x}| \leq R$. Then, Follow the Perturbed Leader with $\mathbf{c}^0 \sim \mathrm{Unif}\left[0, \frac{2}{\epsilon}\right]^n$ has*

$$\mathbb{E}\left[\mathit{Regret}\right] \leq \frac{2D}{\epsilon} + TRA\epsilon.$$

Note that for $\epsilon = \sqrt{\frac{2D}{ART}}$, this leads to $\mathbb{E}\left[\mathit{Regret}\right] \leq \sqrt{2ARDT}$.

Proof. Recall from Theorem 2.2 that

$$\mathbb{E}\left[\text{Regret}\right] \leq \mathbb{E}\left[\mathbf{c}^0 \cdot (\mathbf{x}^* - \mathbf{x}^1) + \sum_{t=1}^{T} \mathbf{c}^t \cdot (\mathbf{x}^t - \mathbf{x}^{t+1})\right] \tag{1}$$

For the first term in the above inequality we have

$$\mathbf{c}^0 \cdot (\mathbf{x}^* - \mathbf{x}^1) \leq \|\mathbf{c}^0\|_\infty \|\mathbf{x}^* - \mathbf{x}^1\|_1 \leq \frac{2}{\epsilon} D.$$

Next, we will analyze the second term in Equation 1. By linearity of expectation, we only need to show that $\mathbb{E}[\mathbf{c}^t \cdot \mathbf{x}^t] - \mathbb{E}[\mathbf{c}^t \cdot \mathbf{x}^{t+1}] \leq D\epsilon/2$ for a fixed time step t. To help us with the notation, we'll define $\mathrm{Box}(\mathbf{a}, \mathbf{b})$ as the box with origin at $\mathbf{a}$ and length $\mathbf{b}$. Note that FTPL is choosing a random $\mathbf{c}^0 \sim \mathrm{Box}(\mathbf{0}, \frac{\mathbf{2}}{\boldsymbol{\epsilon}})$.

Figure 1 shows the core idea. $\mathbf{x}^t$ is chosen to optimize $\mathbf{x} \cdot \sum_{\tau=0}^{t-1} \mathbf{c}^\tau$ and $\mathbf{x}^t$ is chosen to optimize $\mathbf{x} \cdot \sum_{\tau=0}^{t} \mathbf{c}^\tau$, when $\mathbf{c}^0$ is taken from the blue and orange boxes respectively. The parameters are chosen so as to make the overlap between these two boxes very large. When $\mathbf{c}^0$ is within this overalap then intuitively whether or not $\mathbf{c}^t$ was observed by the algorithms matters less. This shows that with high probability $\mathbf{c}^t \cdot \mathbf{x}^t$ is the same as $\mathbf{c}^t \cdot \mathbf{x}^{t+1}$.

Let's formalize this using Figure 2.

$$p := \Pr_{\mathbf{c}^0 \sim \mathrm{Box}(\mathbf{0}, \mathbf{2}/\boldsymbol{\epsilon})}\left[\mathbf{c}^0 \notin \mathrm{Box}(\mathbf{c}^t, \frac{\mathbf{2}}{\boldsymbol{\epsilon}})\right] = \frac{\mathrm{Box}(\mathbf{0}, \frac{\mathbf{2}}{\boldsymbol{\epsilon}}) \setminus \mathrm{Box}(\mathbf{c}^t, \frac{\mathbf{2}}{\boldsymbol{\epsilon}})}{\mathrm{Box}(\mathbf{0}, \frac{\mathbf{2}}{\boldsymbol{\epsilon}})} = \frac{\mathrm{Box}(\mathbf{0}, \frac{\mathbf{2}}{\boldsymbol{\epsilon}}) \setminus \mathrm{Box}(0, \frac{\mathbf{2}}{\boldsymbol{\epsilon}} - \mathbf{c}^{\mathbf{t}})}{\mathrm{Box}(\mathbf{0}, \frac{\mathbf{2}}{\boldsymbol{\epsilon}})}$$

We can upper bound this probability by summing over the dimensions (which involves double counting):

$$p \leq \frac{\sum_i c^t(i) \left(\frac{2}{\epsilon}\right)^{n-1}}{\left(\frac{2}{\epsilon}\right)^n} \leq A\frac{\epsilon}{2}$$

So, the problematic case (sampling from different areas) only happens with small probability at each stage. Let's use this to calculate $\mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^t\right]$, by taking $\mathbf{c}^0 \sim \mathrm{Box}(\mathbf{0}, \mathbf{2}/\boldsymbol{\epsilon})$ and conditioning on whether or not it falls in the overlap.

$$\mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^t\right] = \Pr\left[\mathbf{c}^0 \notin \mathrm{Box}\left(\mathbf{c}^t, \frac{2}{\epsilon}\right)\right] \mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^t | \mathbf{c}^0 \notin \mathrm{Box}\left(\mathbf{c}^t, \frac{2}{\epsilon}\right)\right]$$

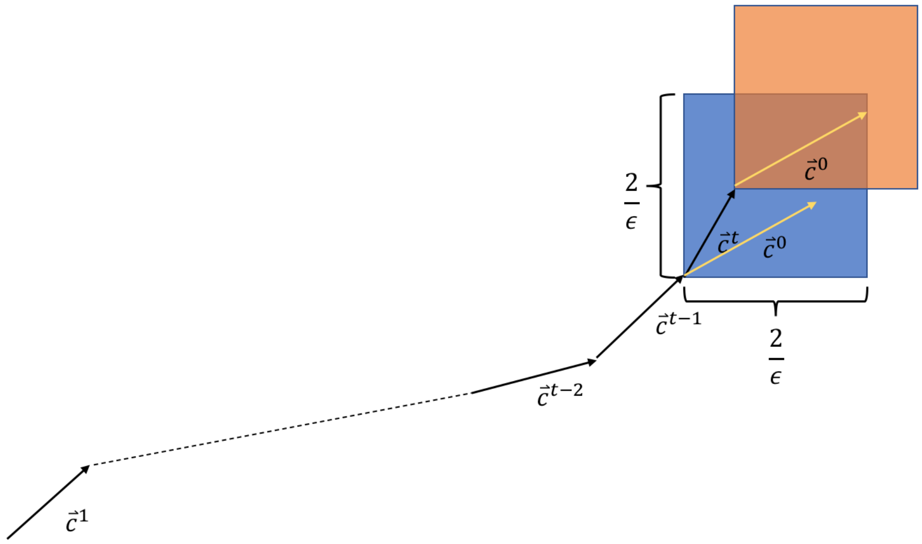

Figure 1: Diagram showing the vector of costs placed end-to-end up to time $t-1$. The blue box contains the range of possible areas for $\mathbf{c}^0$ and the orange one is the range of possible areas for $\mathbf{c}^0$, given $\mathbf{c}^t$. The diagram shows that they largely overlap, so having any one particular $\mathbf{c}^t$ being present or not does not largely matter.

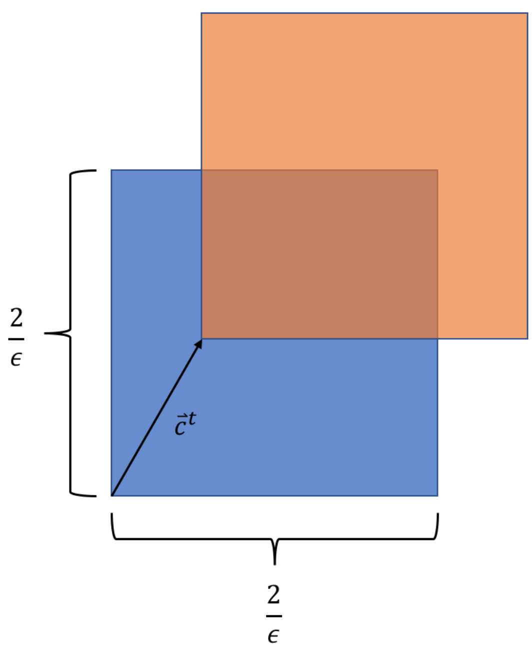

Figure 2: Figure demonstrating ways of estimating the size of the overlap

$$+ \Pr\left[\mathbf{c}^0 \in \text{Box}\left(\mathbf{c}^t, \frac{2}{\epsilon}\right)\right] \mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^t | \mathbf{c}^0 \in \text{Box}\left(\mathbf{c}^t, \frac{2}{\epsilon}\right)\right]$$
$$= \underbrace{p \cdot \mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^t | \mathbf{c}^0 \notin \text{Box}\left(\mathbf{c}^t, \frac{2}{\epsilon}\right)\right]}_{(1)} + (1-p) \cdot \underbrace{\mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^t | \mathbf{c}^0 \in \text{Box}\left(\mathbf{c}^t, \frac{2}{\epsilon}\right)\right]}_{(2)}$$

Next, let's calculate $\mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^{t+1}\right]$. This time, we condition on whether $\mathbf{c}^0 \notin \text{Box}(\mathbf{0}, \frac{\mathbf{2}}{\boldsymbol{\epsilon}} - \mathbf{c}^t)$. Note that as we showed in the definition of p, the probability of this event is p as well.

$$\mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^{t+1}\right] = \underbrace{p \cdot \mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^{t+1} | \mathbf{c}^0 \notin \text{Box}\left(0, \frac{2}{\epsilon} - \mathbf{c}^t\right)\right]}_{(3)} + (1-p) \cdot \underbrace{\mathbb{E}\left[\mathbf{c}^t \cdot \mathbf{x}^{t+1} | \mathbf{c}^0 \in \text{Box}\left(\mathbf{0}, \frac{2}{\epsilon} - \mathbf{c}^t\right)\right]}_{(4)}$$

Recall that

$$\mathbf{x}^t \leftarrow \text{argmin } \mathbf{c}^0 \cdot \mathbf{x} + \sum_{\tau=1}^{t-1} \mathbf{c}^\tau \cdot \mathbf{x}$$

$$\mathbf{x}^{t+1} \leftarrow \text{argmin } \mathbf{c}^0 \cdot \mathbf{x} + \mathbf{c}^t \cdot \mathbf{x} + \sum_{\tau=1}^{t-1} \mathbf{c}^\tau \cdot \mathbf{x}.$$

Therefore, the distribution of $\mathbf{x}^{t+1}$ conditional on $\mathbf{c}^0 \in \text{Box}(\mathbf{0}, \frac{\mathbf{2}}{\boldsymbol{\epsilon}} - \mathbf{c}^t)$ is the same as the distribution of $\mathbf{x}^t$ conditional on $\mathbf{c}^0 \in \text{Box}(\mathbf{c}^t, \frac{\mathbf{2}}{\boldsymbol{\epsilon}})$. This is the key idea that is showing that terms (2) and (4) are equal to each other.

Next, we see that terms (1) and (3) are close to each other. That is,

$$(3) - (4) \leq p \cdot \mathbf{c}^t(\mathbf{x}^t - \mathbf{x}^{t+1}) \leq A\frac{\epsilon}{2}2R \leq \epsilon AR$$

We have T of these time steps, so the overall contribution is $TRA\epsilon$.

Combining all this into Equation 1, we get

$$\mathbb{E}[\text{Regret}] \leq \frac{2D}{\epsilon} + TRA\epsilon$$

as desired.

$\square$