Machine Learning Theory (CS 6783)

Lecture 20: Contextual Bandits

So far we have seen the Bandit problem, both the multi-armed bandit and linear bandit settings. In this lecture, we will briefly look at fancier versions of bandit problems. Specifically, we will look at a contextual version of bandit problems.

1 Contextual Bandit Problem

Multi-armed bandit problem is one where we have N actions or arms and each day we pull one and get losses based on what we pulled. This setting is great for very simple ad placement problem for instance. Example, we have N ads and we want a strategy to display ad that is more likely to be clicked. However, in more practical scenarios, we don't just place ad's without using other, "contextual" information. For instance, when placing ad, we use information like, what season it is, who the user is what is their history etc. The contextual bandit problem considers this more realistic scenario.

- For $t = 1$ to T
  - Nature produces context $x_t \in \mathcal{X}$
  - Algorithm picks arm $I_t \in [N]$ in a possibly randomized fashion while nature produces loss vector ℓ_t
  - Learner suffers loss $\ell_t[I_t]$

Goal: Minimize regret w.r.t. class of policies $\mathcal{F} \subset [N]^{\mathcal{X}}$ given by

$$\text{Reg}_T = \sum_{t=1}^{T} \ell_t[I_t] - \inf_{f \in \mathcal{F}} \sum_{t=1}^{T} \ell_t[f(x_t)]$$

That is, we want to do as well as the best policy in $\mathcal{F}$ which takes into account context x_t before picking an arm on each round. If $\mathcal{F}$ had just the N constant mappings of picking just each of the N arms ignoring context, then this problem is same as the multi-armed bandit one.

What is a strategy here? Well we could simply ignore the fact that there are only N arms/options on each round and just treat is as a multi-armed bandit problem over $|\mathcal{F}|$ arms (each policy is one arm). However in this case, our regret bounds would be at least $\sqrt{|\mathcal{F}|\ T}$ and since we would like to consider rich class of policies $\mathcal{F}$ whose cardinality could possibly be very very large compared to n, this approach of plainly using the bandit algorithm would fail. However lets consider a more careful reuse of the exponential weights algorithm. Recall from the Bandits lecture that the key to getting the bandit result was to plug in unbiased estimates of losses into a more carefully analyzed full information algorithm. Let us have a second look at what we had there.

2 EXP4 Algorithm

An appropriately modified version of exponential weights algorithm it turns out is effective for contextual bandit problems, even when contexts and losses are adversarially chosen. Specifically, we run exponential weights algorithm over $\mathcal{F}$ and when it comes time to pick an arm at round t, we simply pick an expert $f \in \mathcal{F}$ according to distribution $\hat{y}_t \in \Delta(\mathcal{F})$ and then play the action suggested by this expert as $\hat{y}_t(x_t)$. We use for the exponential weights algorithm, the unbiased estimate of loss of expert $f \in \mathcal{F}$ as:

$$\tilde{\ell}_t[f] = \frac{1}{\sum_{f \in \mathcal{F}: f(x_t) = I_t} \hat{y}_t[f]} \mathbf{1}\{f(x_t) = I_t\} \ell_t[I_t]$$

This algorithm is called EXP4.

Lemma 1. *For the EXP4 algorithm mentioned above, we have the following bound on expected regret:*

$$\mathbb{E}\left[\mathrm{Reg}_T\right] \leq O\left(\sqrt{NT \log |F|}\right)$$

Proof. If we write down the local norm based bound we already derived for exponential weights (mirror descent with entropy regularizer) algorithm, we have that

$$\mathrm{Reg}_T(\tilde{\ell}_1, \ldots, \tilde{\ell}_T) \leq \frac{\eta}{2} \sum_{t=1}^{T} \sum_{f \in \mathcal{F}} \hat{y}_t[f] \tilde{\ell}_t[f]^2 + \frac{\log |\mathcal{F}|}{\eta}$$

where $\tilde{\ell}_1, \ldots, \tilde{\ell}_n$ are the losses over the experts in $\mathcal{F}$ for the T rounds that is fed to the full information algorithm. $\hat{y}_t$ is the distribution over the experts produced by the algorithm in round t. Now the main thing to note is that at round t, if we pick an expert $f \in \mathcal{F}$, we observe loss $\ell_t[f(x_t)]$. But this is loss of arm $f(x_t)$ and so in reality we observe loss of not just f for that round but rather all the other $f' \in \mathcal{F}$ that are such that $f'(x_t) = f(x_t)$. Lets make use of this information. Specifically, let us define our unbiased estimate of loss to be:

$$\tilde{\ell}_t[f] = \frac{1}{\sum_{f \in \mathcal{F}: f(x_t) = I_t} \hat{y}_t[f]} \mathbf{1}\{f(x_t) = I_t\} \ell_t[I_t]$$

Why is this an unbiased estimate? Well consider our algorithm. It draws first $g \sim \hat{y}_t$ and then uses action $I_t = g(x_t)$. Hence, for any $f \in \mathcal{F}$,

$$\begin{aligned}
\mathbb{E}_{g \sim \hat{y}_t}\left[\tilde{\ell}_t[f]\right] &= \mathbb{E}_{g \sim \hat{y}_t}\left[\frac{1}{\sum_{f \in \mathcal{F}: f(x_t) = g(x_t)} \hat{y}_t[f]} \mathbf{1}\{f(x_t) = g(x_t)\} \ell_t[g(x_t)]\right] \\
&= \sum_{g \in \mathcal{F}} \hat{y}_t[g] \frac{1}{\sum_{f \in \mathcal{F}: f(x_t) = g(x_t)} \hat{y}_t[f]} \mathbf{1}\{f(x_t) = g(x_t)\} \ell_t[g(x_t)] \\
&= \sum_{g \in \mathcal{F}: g(x_t) = f(x_t)} \hat{y}_t[g] \frac{1}{\sum_{f' \in \mathcal{F}: f'(x_t) = f(x_t)} \hat{y}_t[f']} \ell_t[f(x_t)] \\
&= \ell_t[f(x_t)] \frac{\sum_{g \in \mathcal{F}: g(x_t) = f(x_t)} \hat{y}_t[g]}{\sum_{f' \in \mathcal{F}: f'(x_t) = f(x_t)} \hat{y}_t[f']} \\
&= \ell_t[f(x_t)]
\end{aligned}$$

Hence if we use this new unbiased estimate, we have that:

$$\begin{aligned}
\mathrm{Reg}_T(\tilde{\ell}_1, \ldots, \tilde{\ell}_T) &\leq \frac{\eta}{2} \sum_{t=1}^{T} \sum_{f \in \mathcal{F}} \hat{y}_t[f] \left(\frac{1}{\sum_{f \in \mathcal{F}: f(x_t) = I_t} \hat{y}_t[f]} \mathbf{1}\{f(x_t) = I_t\} \ell_t[I_t] \right)^2 + \frac{\log |\mathcal{F}|}{\eta} \\
&= \frac{\eta}{2} \sum_{t=1}^{T} \sum_{f \in \mathcal{F}} \hat{y}_t[f] \left(\frac{1}{\sum_{f \in \mathcal{F}: f(x_t) = I_t} \hat{y}_t[f]} \right)^2 \mathbf{1}\{f(x_t) = I_t\} \ell_t^2[I_t] + \frac{\log |\mathcal{F}|}{\eta} \\
&= \frac{\eta}{2} \sum_{t=1}^{T} \left(\sum_{f \in \mathcal{F}: f(x_t) = I_t} \hat{y}_t[f] \right) \left(\frac{1}{\sum_{f \in \mathcal{F}: f(x_t) = I_t} \hat{y}_t[f]} \right)^2 \ell_t^2[I_t] + \frac{\log |\mathcal{F}|}{\eta} \\
&= \frac{\eta}{2} \sum_{t=1}^{T} \frac{\ell_t^2[I_t]}{\sum_{f \in \mathcal{F}: f(x_t) = I_t} \hat{y}_t[f]} + \frac{\log |\mathcal{F}|}{\eta}
\end{aligned}$$

Now taking expectation on both sides, we get:

$$\begin{aligned}
\mathbb{E}\left[\mathrm{Reg}_T(\tilde{\ell}_1, \ldots, \tilde{\ell}_T)\right] &\leq \frac{\eta}{2} \sum_{t=1}^{T} \mathbb{E}_{I_t} \left[\frac{\ell_t^2[I_t]}{\sum_{f \in \mathcal{F}: f(x_t) = I_t} \hat{y}_t[f]} \right] + \frac{\log |\mathcal{F}|}{\eta} \\
&\leq \frac{\eta}{2} \sum_{t=1}^{T} \sum_{k=1}^{N} \left(\sum_{f \in \mathcal{F}: f(x_t) = k} \hat{y}_t[f] \right) \frac{\ell_t^2[k]}{\sum_{f \in \mathcal{F}: f(x_t) = k} \hat{y}_t[f]} + \frac{\log |\mathcal{F}|}{\eta} \\
&\leq \frac{\eta}{2} \sum_{t=1}^{T} \sum_{k=1}^{N} \ell_t^2[k] + \frac{\log |\mathcal{F}|}{\eta} \\
&\leq \frac{\eta N T}{2} + \frac{\log |\mathcal{F}|}{\eta}
\end{aligned}$$

Optimizing over η and recalling that expected regret of the bandit algorithm is upper bounded by expected regret of the full information algorithm we conclude that

$$\mathbb{E}\left[\mathrm{Reg}_T\right] \leq O\left(\sqrt{NT \log |F|}\right)$$

□

Thus we can in fact get logarithmic dependence on the number of policies and a $\sqrt{N}$ dependence for regret bound. This algorithm is known as EXP4 algorithm. This algorithm does have the optimal regret bound up to constant factors.

3 ERM Oracle Efficient Algorithms

A drawback of this algorithm though is that when $|\mathcal{F}|$ is very large, this algorithm is computationally inefficient. This is because, the algorithm maintains a distribution over $\mathcal{F}$ and the computation time per round is linear in $|\mathcal{F}|$. In general, this issue can be real. However, practically, to alleviate this issue, one might want to assume access to an ERM oracle. That is, given a sequence of instances,

$x_1, \ell_1, \ldots, \ell_m, x_m$, we assume that we can efficiently compute the $f \in \mathcal{F}$ that minimizes empirical loss. That is an oracle that either exactly or approximately returns

$$\widehat{f}_{\text{ERM}} = \operatorname*{argmin}_{f \in \mathcal{F}} \sum_{t=1}^{m} \tilde{\ell}_t[f(x_t)]$$

Now in general, assumption of access to such an ERM oracle may not mean we can minimize regret. However, if one assumes that context and loss are drawn from a fixed distribution, that is the iid stochastic setting, then it is not hard to see that one can achieve regret that goes to 0 and only has a poly-log dependence on $|\mathcal{F}|$. To see this, say n were known in advance and consider the algorithm that for first m rounds simply picks the uniform distribution over strategies and hence builds an unbiased estimate over loss vectors as $\hat{\ell}_t = N e_{I_t} \ell_t[I_t]$, Now just as in multi-armed bandit setting, $(x_1, \hat{\ell}_1), \ldots, (x_m, \hat{\ell}_m)$ is an unbiased estimate of $(x_1, \ell_1), \ldots, (x_m, \ell_m)$. Now from Hoeffding Azuma inequality with union bound, one can argue that f_{ERM} obtained by running ERM oracle run on the estimate is order $N\sqrt{\log |\mathcal{F}|/m}$ sub-optimal. Hence if we use this hypothesis for all future $n - m$ rounds we obtain the upper bound of order

$$\mathbb{E}\left[\text{Reg}_T\right] \leq (T-m)N \sqrt{\log |\mathcal{F}|/m} + m \leq T\sqrt{N \log |\mathcal{F}|/m} + m$$

Using $m = N^{2/3}(\log |\mathcal{F}|)^{1/3} T^{2/3}$ we get, the final bound of order $O(N^2 \log |\mathcal{F}|/n)^{1/3}$. If one uses a better concentration than just Hoeffding Azuma, the bound can be improved to $O(N \log |\mathcal{F}|/n)^{1/3}$

Another algorithm that is close in spirit to the so called epsilon-greedy algorithm which in every round explores uniformly with a small probability and exploits by using ERM based on estimated losses so far with remaining probability. This epsilon-greedy algorithm also attains the same upper bound as the one this above naive algorithm attains. The idea is to play the following, first define $\hat{f}_t$ as

$$\hat{f}_t = \text{ERM}(x_1, \tilde{\ell}_1, \ldots, x_{t-1}, \tilde{\ell}_{t-1})$$

Now the algorithm for ϵ-greedy plays $\hat{y}_t(x_t) = \hat{f}_t(x_t)$ with probability $1 - \gamma$ and $\hat{y}_t(x_t)$ is drawn from the uniform distribution over the N Arms with probability γ. The main idea is the following. First, the estimate of loss $\tilde{\ell}_t \in \mathbb{R}^N$

$$\tilde{\ell}_t = \frac{\ell_t[I_t]}{P(I_t)} e_{I_t}$$

The key idea to notice is that the loss is bounded by $\frac{N}{\gamma}$ and further, the variance of the loss is bounded by $\frac{N}{\gamma}$ and so if we use Freedman's inequality which is basically a variance based version of Hoeffding Azuma, we can guarantee that the estimated losses are close to expected (which is ℓ_t on average).

It turns out that this regret bound is not optimal. One can view the above as requiring that on every round our loss estimates have variance bounded for every policy in the class by the same value of $\frac{N}{\gamma}$. But clearly this seems more than necessary since as we go along, we know that only some policies are candidates for being good policies and so a more careful analysis would be to do a more careful exploration where we only explore in a way where for good policies, we have a nice bound on variance of loss estimates. Such an algorithm needs to maintain a set of good policies $\mathcal{F}_t$ on every round that shrinks as we go. Then we could pick an exploration distribution that only needs to satisfy the guarantee that on such good policies, the variance is low. Such an algorithm is referred to as policy elimination. One can in fact obtain the optimal rate of $\sqrt{N \log |\mathcal{F}|/n}$ bound

using this algorithm. However, such an algorithm is not efficient as we need to maintain a set $\mathcal{F}_t$ which is prohibitive. But this idea can be explored in a smarter fashion to design an algorithm that is ERM oracle efficient and also attains the optimal regret bound. We will look at a sketch of this in the next lecture.