CS861: Theoretical Foundations of Machine Learning Lecture 22 - 10/25/2024

University of Wisconsin–Madison, Fall 2024

Lecture 22:Contextual Bandits, Online Convex Optimization

Lecturer: Kirthevasan Kandasamy *Scribed by: Xinyu Li and Zhexuan Liu*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the instructor.*

Contextual Bandits

So far, we have looked at K arms (actions) and competed against the single best action in hindsight. But the best action may depend on contextual information, which may be available to the learner. For example, Advertising (bandits): find the best ad; Targeted advertising (contextual bandits): find the best ad for a given query/user (context).

A policy which has good regular bandit regret may have poor performance in a real-world application.

Definition 1 (The contextual bandit problem)**.** *We will define the contextual bandit problem as follows:*

(1) There are a set of K actions, denoted $[K]$.

(2) At the beginning of each round t, the adversary picks a context $x_t \in \mathcal{X}$. The learner observes x_t.

(3) The learner then chooses an action $A_t \in [K]$.

(4) The adversary simultaneously (i.e. without knowledge of A_t) picks a loss vector $\ell_t \in [0,1]^K$, where $\ell_t(i)$ is the loss for action i.

(5) The learner incurs the loss $\ell_t(A_t)$.

(6) The learner observes only $\ell_t(A_t)$.

Question: How do we define regret here?

- One option is to compete against the best action for the given context:

$$R_T(\pi, \ell, x) = \mathbb{E}\Big[\sum_{t=1}^{T} \ell_t(A_t)\Big] - \min_{e:\mathcal{X}\to[K]} \sum_{i=1}^{T} \ell_t(e(x_t))$$

  where $\mathbb{E}$ is w.r.t. the randomness of policy. Here, we are competing against the single best mapping from contexts to actions.

  - This is like running a separate bandit algorithm for different contexts.
  - And this is challenging if the number of possible contexts is large (possibly infinite), but also unnecessary if there are relationships between contexts. (e.g querying ‘frying pan’ vs ‘non-stick skillet’ in targeted advertising)

- Instead, we will look at a set of N ”experts” who map contexts to actions and we will now be competing against the single best expert in hindsight. Here, the experts could be, say, machine learning models trained on a variety of large datasets.

▲ If the experts are $\{e_1, \dots, e_N\}$, where $e_j : \mathcal{X} \to [K]\ \forall j \in [N]$, then write

$$R_T(\pi, \ell, x) = \mathbb{E}\Big[\sum_{t=1}^{T} \ell_t(A_t)\Big] - \min_{j\in[N]} \sum_{t=1}^{T} \ell_t(e_j(x_t))$$

Question: Can we apply `EXP3` algorithm here by treating the experts as actions?

■ Yes, as we can define a loss vector $\tilde{\ell}_t \in [0,1]^N$, where $\tilde{\ell}_t(j) = \ell_t(e_j(x_t))$. But the regret is going to be large. $R_T \in O(\sqrt{TN\log(N)})$ which is fine as long as the number of experts, N, is small. However, we are usually interested in cases where we have many more experts than possible actions, $N \gg K$. N could be as large as $|\mathcal{X}|^K$(if $\mathcal{X}$ is finite). If the experts are ML models, N could be discretizations of the weights of the model. We wish to reduce from $\text{poly}(N)$ to $\text{poly}\log(N)$.

`EXP4` algorithm Build on `EXP3`, but use the fact that when we observe feedback, we can discount all experts who would have chosen the action.

The `EXP4` algorithm

Just like we built on `Hedge` to arrive at the `EXP3` algorithm, we are going to build on the `EXP3` algorithm to arrive at the `EXP4` algorithm here. We will treat the experts as arms here and run `EXP3` on them, but what we will do differently here is the following: we will use the fact that when we observe feedback, we can discount all the experts who would have chosen the action. Based on this, we can express the pseudocode of `EXP4` as shown below:

Algorithm 1 The EXP-4 algorithm (Exponential weights for exploration and exploitation with experts)

```
Require: Time horizon $T$, learning rate $\eta$
  Let $\tilde{L}_0 \leftarrow \mathbf{0}_N$
  for $t = 1, ..., T$ do
      Observe conte$x_t$
      Construct $\tilde{p}_t$ as follows, $\tilde{p}_t(i) = \frac{e^{-\eta\tilde{L}_{t-1}(i)}}{\sum_{j=1}^{N} e^{-\eta\tilde{L}_{t-1}(j)}}$, for all experts $i \in [N]$
      Construct $p_t \in \Delta([K])$ via, $p_t(a) = \sum_{j=1}^{N} \tilde{p}_t(j)\mathbb{I}(e_j(x_t) = a)$
      Sample $A_t \sim p_t$ and execute $A_t$. Observe $\ell_t(A_t)$.
      Compute action losses, $\hat{\ell}_t(a) \leftarrow \frac{\ell_t(a)}{p_t(a)}\mathbb{I}(A_t = a)\ \forall a \in [k]$
      Compute expert losses, $\tilde{\ell}_t(j) \leftarrow \hat{\ell}_t(e_j(x_t))\ \forall j \in [N]$
      Update cumulative losses, $\tilde{L}_t(j) \leftarrow \tilde{L}_{t-1}(j) + \tilde{\ell}_t(j)\ \forall j \in [N]$
  end for
```

Remark Some observations about the `EXP4` algorithm:

- Instead of explicitly constructing p_t, we can sample an expert E_t from $\tilde{p}_t$ and then choose $A_t = E_t(x_t)$.
- We can write the loss update as

$$\tilde{L}_t(j) \longleftarrow \tilde{L}_{t-1}(j) + \mathbf{1}\{e_j(x_t) = A_t\} \cdot \frac{\ell_t(A_t)}{p_t(A_t)}$$

We are using the probability of choosing A_t (via p_t), and not just the probability of choosing the relevant expert E_t.

▲ Note that we are not discounting only one expert here. That is, we are NOT utilizing the following update rule:

$$\tilde{L}_t(j) \longleftarrow \tilde{L}_{t-1}(j) + \mathbf{1}\{E_t = j\} \cdot \frac{\ell_t(A_t)}{\tilde{p}_t(E_t)}$$

Theorem 1 (Regret bound for `EXP4`)**.** *Suppose that the loss vectors on each round t satisfy $\ell_t \in [0,1]^K$ and we choose $\eta = \sqrt{\frac{\log(N)}{KT}}$. Then for all $T \geq \log(N)/K$, and all $\ell \in [0,1]^{K \times T}$ and $x \in \mathcal{X}^T$, the regret of EXP4 satisfies,*

$$R_T(\pi^{EXP4}, \ell, x) \leq 2\sqrt{KT\log(N)}$$

where N is the number of experts.

We will use Hedge lemma to prove this result.

Lemma 1. *(Hedge Lemma) Let $\lambda = (\lambda_1, \dots, \lambda_T) \in \mathbb{R}_+^N$ be a sequence of losses. Let $\widetilde{p}$ be the sequence of probability vectors chosen by Hedge with learning rate $\eta \in [0,1]$. For any $j \in [N]$, if $\widetilde{p}_t^T \lambda_t \leq 1$ for all t, we have*

$$\sum_{t=1}^{T} \widetilde{p}_t^T \lambda_t - \sum_{t=1}^{T} \lambda_t\left(e_j\right) \leq \frac{\log(N)}{\eta} + \eta \sum_{t=1}^{T} \widetilde{p}_t^T \lambda_t^2$$

Proof

Let $j_\star = \operatorname{argmin}_{j \in [N]} \sum_{t=1}^{T} \ell_t\left(e_j\left(x_t\right)\right)$ be the best fixed expert in hindsight. We will apply the lemma with $j \leftarrow j_\star$, and $\lambda_t \in \mathbb{R}_+^N$ where $\lambda_t(i) \leftarrow \widetilde{\ell}_t\left(e_i\left(x_t\right)\right)$. Let us first verify the conditions,

$$\eta = \sqrt{\log(N)/(KT)} \leq 1 \quad \text{as } T \geq \log(N)/K$$

To verify $\widetilde{p}_t^T \widetilde{\ell}_t \leq 1$, recall that $\widetilde{\ell}(j) \leftarrow \widehat{\ell}_t\left(e_j\left(x_t\right)\right) = \frac{\ell_t(e_j(x_t))}{p_t(e_j(x_t))} \mathbf{1}\left(A_t = e_j\left(x_t\right)\right)$. Therefore,

$$\begin{aligned}
\widetilde{p}_t^T \widetilde{\ell}_t &= \sum_{j=1}^{N} \widetilde{p}_t(j) \frac{\ell_t\left(e_j\left(x_t\right)\right)}{p_t\left(e_j\left(x_t\right)\right)} \mathbf{1}\left(A_t = e_j\left(x_t\right)\right) = \frac{\ell_t\left(A_t\right)}{p_t\left(A_t\right)} \sum_{j=1}^{N} \widetilde{p}_t(j) \mathbf{1}\left(A_t = e_j\left(x_t\right)\right) \\
&= \frac{\ell_t\left(A_t\right)}{p_t\left(A_t\right)} \times p_t\left(A_t\right) = \ell_t\left(A_t\right) \leq 1
\end{aligned}$$

Now consider,

$$\mathbb{E}\left[\widetilde{\ell}_t(j) \mid \widetilde{p}_t\right] = p_t\left(e_j\left(x_t\right)\right) \cdot \frac{\ell_t\left(e_j\left(x_t\right)\right)}{p_t\left(e_j\left(x_t\right)\right)} + \left(1 - p_t\left(e_j\left(x_t\right)\right)\right) \cdot 0 = \ell_t\left(e_j\left(x_t\right)\right)$$

Similarly,

$$\mathbb{E}\left[\widetilde{\ell}_t^2(j) \mid \widetilde{p}_t\right] = p_t\left(e_j\left(x_t\right)\right) \cdot \frac{\ell_t^2\left(e_j\left(x_t\right)\right)}{p_t^2\left(e_j\left(x_t\right)\right)} + \left(1 - p_t\left(e_j\left(x_t\right)\right)\right) \cdot 0 = \frac{\ell_t^2\left(e_j\left(x_t\right)\right)}{p_t\left(e_j\left(x_t\right)\right)}$$

Remark: Here, we have $p_t\left(e_j\left(x_t\right)\right) = \sum_k \widetilde{p}_t(k) \mathbf{1}\left(e_k\left(x_t\right) = e_j\left(x_t\right)\right)$ in the denominator. Naively applying EXP3 we will get $\widetilde{p}\left(e_j\left(x_t\right)\right) < p_t\left(e_j\left(x_t\right)\right)$ in the denominator. The estimate for the loss in EXP4 has lower variance since $\mathbb{E}\left[\widetilde{\ell}_t^2 \mid \widetilde{p}_t\right]$ is smaller.

Applying the Hedge lemma with $j \leftarrow j_\star$, we get

$$\sum_{t=1}^{T} \widetilde{p}_t^T \widetilde{\ell}_t - \sum_{t=1}^{T} \widetilde{\ell}_t\left(j_\star\right) \leq \frac{\log(N)}{\eta} + \eta \sum_{t=1}^{T} \widetilde{p}_t^T \widetilde{\ell}_t^2$$

Let us take expectations on both sides.

$$\mathbb{E}[\mathrm{LHS}] = \mathbb{E}\left[\mathbb{E}\left[\mathrm{LHS} \mid p_t\right]\right] = \mathbb{E}[\sum_{t=1}^{T} \mathbb{E}\left[\widetilde{p}_t^{\top} \widetilde{\ell}_t \mid \widetilde{p}_t\right] - \sum_{t=1}^{T} \underbrace{\mathbb{E}\left[\widetilde{\ell}_t\left(j_*\right) \mid \widetilde{p}_t\right]}_{=\ell_t\left(e_{j_*}\left(x_t\right)\right)}]$$

We then have

$$\begin{aligned}\mathbb{E}\left[\widetilde{p}_t^{T} \widetilde{\ell}_t \mid \widetilde{p}_t\right] &= \widetilde{p}_t^{T} \mathbb{E}\left[\widetilde{\ell}_t \mid \widetilde{p}_t\right] = \sum_{j=1}^{n} \widetilde{p}(j) \ell_t\left(e_j\left(x_t\right)\right) = \sum_{j=1}^{n} \widetilde{p}(j) \sum_{a=1}^{K} \ell_t(a) \mathbf{1}\left(a = e_j\left(x_t\right)\right) \\ &= \sum_{a=1}^{K} \ell_t(a) \underbrace{\sum_{j=1}^{n} \widetilde{p}(j) \mathbf{1}\left(a = e_j\left(x_t\right)\right)}_{=p_t(a)} = p_t^{T} \ell_t = \mathbb{E}\left[\ell_t\left(A_t\right) \mid p_t\right].\end{aligned}$$

Therefore,

$$\begin{aligned}\mathbb{E}[\mathrm{LHS}] &= \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{E}\left[\ell_t\left(A_t\right) \mid \widetilde{p}_t\right] - \sum_{t=1}^{T} \ell_t\left(e_{j_\star}\left(x_t\right)\right)\right] \\ &= \mathbb{E}\left[\sum_{t=1}^{T} \ell_t\left(A_t\right)\right] - \min_{j \in [N]} \sum_{t=1}^{T} \ell_t\left(e_j\left(x_t\right)\right) = R_T\left(\pi^{\mathrm{EXP4}}, \ell, x\right)\end{aligned}$$

Now consider the RHS of Hedge inequality,

$$\mathbb{E}[\mathrm{RHS}] = \mathbb{E}\left[\mathbb{E}\left[\mathrm{RHS} \mid \widetilde{p}_t\right]\right] = \frac{\log(N)}{\eta} + \eta \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{E}\left[\widetilde{p}_t^{T} \widehat{\ell}_t^{2} \mid \widetilde{p}_t\right]\right]$$

As losses are bounded in [0, 1], we have

$$\begin{aligned}\mathbb{E}\left[\widetilde{p}_t^{T} \widehat{\ell}_t^{2} \mid \widetilde{p}_t\right] &= \sum_{j=1}^{N} \widetilde{p}_t(j) \frac{\ell_t^2\left(e_j\left(x_t\right)\right)}{p_t\left(e_j\left(x_t\right)\right)} = \sum_{j=1}^{N} \widetilde{p}(j) \sum_{a=1}^{K} \frac{\ell_t^2(a)}{p_t(a)} \mathbf{1}\left(e_j\left(x_t\right) = a\right) \\ &= \sum_{a=1}^{K} \frac{\ell_t^2(a)}{p_t(a)} \sum_{j=1}^{N} \widetilde{p}(j) \mathbf{1}\left(e_j\left(x_t\right) = a\right) = \sum_{a=1}^{K} \frac{\ell_t^2(a)}{p_t(a)} p_t(a) \leq K\end{aligned}$$

Therefore,

$$\mathbb{E}[\mathrm{RHS}] \leq \frac{\log(K)}{\eta} + \eta K T$$

We have,

$$\begin{aligned}R_T\left(\pi^{\mathrm{EXP4}}, \ell, x\right) &\leq \frac{\log(N)}{\eta} + \eta K T \\ &\leq 2\sqrt{K T \log(N)} \quad \text{as } \eta = \sqrt{\log(N)/(K T)}\end{aligned}$$

□

Online Convex Optimization

Definition 2 (Convex set)**.** *A set $\Omega \subset \mathbb{R}^d$ is called convex if, for every two points $\omega, \omega' \in \Omega$ and every $\alpha \in [0,1]$, we have $\alpha\omega + (1-\alpha)\omega' \in \Omega$.*

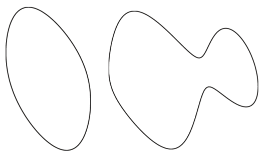

Figure 1: Example of a convex set (left) and a non-convex set (right)

Definition 3 (Convex function)**.** *A function $f : \Omega \to \mathbb{R}$ is convex if Ω is a convex set and $\forall\alpha \in [0,1]$ and all $u, v \in \Omega$ we have, $f(\alpha u + (1-\alpha)v) \leq \alpha f(u) + (1-\alpha)f(v)$. Equivalently, f is convex if, for all $\omega \in \Omega$, there exists $g \in \mathbb{R}^n$ such that $\forall\omega' \in \Omega$, we have $f(\omega') \geq f(\omega) + g^T(\omega' - \omega)$.*

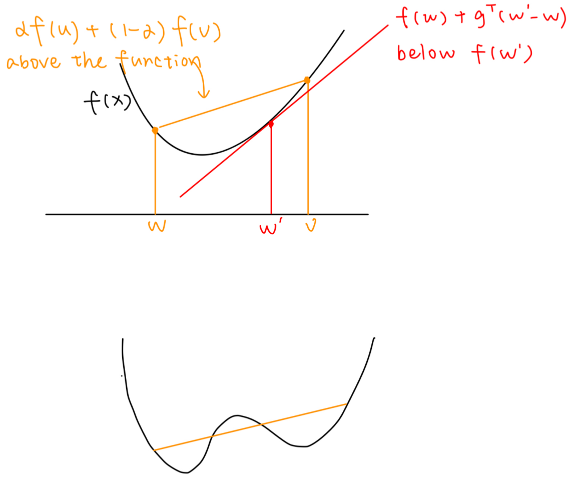

Figure 2: Example of a convex function (top) and a non-convex set (bottom)

Definition 4 (Subgradients and Subdifferentials)**.** *Any g which satisfies the theorem above is called a subgradient of f at ω. The set of all subgradients of ω are called the subdifferential, and denoted $\partial f(\omega)$.*

Some useful facts about subgradients:

- *If f is differentiable at ω, then $\partial f(\omega) = \{\nabla f(\omega)\}$.*
- $\mathbf{0} \in \partial f(\omega) \iff \omega \in \operatorname{argmin}_{\omega\in\Omega} f(\omega)$.
- *If $g_1 \in \partial f_1(\omega)$ and $g_2 \in \partial f_2(\omega)$, then*

$$\alpha g_1 + \beta g_2 \in (\alpha\partial f_1 + \beta\partial f_2) \qquad \textit{for all } \alpha, \beta \in \mathbb{R}$$

Definition 5 (strong convexity)**.** *A convex function* $f : \Omega \to \mathbb{R}$ *is* α*-strongly convex in some norm* $\|\cdot\|$ *if,* $f(\omega') \geq f(\omega) + g^T(\omega' - \omega) + \frac{\alpha}{2}\|\omega' - \omega\|^2 \quad \forall g \in \partial f(\omega)$.

Remark. If f *is strongly convex in* $\|\cdot\|_2$*, this is equivalent to saying that* $f(\omega) - \frac{\alpha}{2}\|\omega\|_2^2$ *is convex, i.e* f *is at least as convex as a quadratic function.*

Define $h(\omega) = f(\omega) - \frac{\alpha}{2}\|\omega\|_2^2$*. Then,* $g \in \partial f(\omega) \Longleftrightarrow g - \alpha\omega \in \partial h(\omega)$.

$$\begin{aligned}
&h(\omega') \geq h(\omega) + (g - \alpha\omega)^T(\omega' - \omega) \Longleftrightarrow \\
&\quad f(\omega') - \frac{\alpha}{2}\|\omega'\|_2^2 \geq f(\omega) - \frac{\alpha}{2}\|\omega\|_2^2 + (g - \alpha\omega)^T(\omega' - \omega) \Longleftrightarrow \\
&\quad f(\omega') \geq f(\omega) + g^T(\omega' - \omega) + \frac{\alpha}{2}\|\omega - \omega'\|_2^2
\end{aligned}$$

Acknowledgements

The note is based on the scribed lecture materials prepared in Fall 2023 by Haoyue Bai & Deep Patel.