Lecture 04: Online Learning with Bandit Feedback

Lecturer: John Lazarsfeld

September 25, 2025

Abstract

Introduction to the bandit feedback setting and adversarial Multi-Armed Bandits; the EXP3 algorithm and a proof of its epected regret guarantee.

Disclaimer: These lecture notes are preliminary notes that will evolve and will be updated over time, this is the first iteration of the course. The treatment is not comprehensive but focuses on some of the main ideas with accompanying proofs.

Recap from Lecture 03. In the previous lectures, we have introduced general families of online learning algorithms (FTRL, OMD, and FTPL) and have established analysis frameworks for proving optimal, sublinear regret bounds in the full or gradient-feedback models. We now turn toward a weaker feedback model of online learning: bandit feedback. In this lecture we introduce the bandit convex optimization setting, the special case of (adversarial) Multi-Armed Bandits (MAB), and we show that a certain instantiation of FTRL adapted to the bandit feedback model can achieve near-optimal regret bounds in the MAB setting.

1 Bandit Convex Optimization

We start by describing a variant of the Online Convex Optimization (OCO) setting where the learner only has access to *bandit* or *zero-order* feedback about the loss functions.

Bandit Feedback Model. Consider the following online learning setup over a decision space $\mathcal{X}$: at time $t = 1, \dots, T$

(1) The learner chooses $x_t \in \mathcal{X}$.
(2) An adversary selects convex $f_t : \mathcal{X} \to \mathbb{R}$.
(3) The learner incurs and observes the cost $f_t(x_t) \in \mathbb{R}$.

Notice that the only difference in this setup compared to the standard OCO setting presented in Lecture 01 is the feedback: in the bandit feedback model, the learner only observes its incurred cost $f_t(x_t)$, as opposed to the full function f_t or the gradient $\nabla f_t(x_t) \in \mathbb{R}^n$. Thus in general, the learner has less information to use when choosing its next action choice, and we should intuitively expect weaker regret bounds compared to the full feedback setting.

Special Case: Multi-Armed Bandits. A special case we will be interested in is the *Multi-Armed Bandit* (MAB) setting. Here, we now consider a *finite* action space $\mathcal{X} = [n] = \{1, \dots, n\}$ of n arms/actions. At each round, the learner chooses an action $i_t \in \mathcal{X} = [n]$, and an adversary chooses a loss vector $\ell_t \in \mathbb{R}^n$ that assigns the loss $\ell_t(i)$ to each action $i \in [n]$. The learner incurs the cost $\ell_t(i_t) \in \mathbb{R}^n$ and observes this cost value as feedback.

In this setting, it is clear that an adversary can always ensure the learner has a large loss if the vector ℓ_t can depend on the choice i_t. Thus, the learner must use randomization to choose i_t. In particular, we will assume the following setup:

Definition 1 (Multi-Armed Bandits)**.** *Let $\mathcal{X} = [n]$. At time $t = 1, \dots, T$*

(1) The learner chooses a distribution $x_t \in \Delta_n$ and samples $i_t \sim x_t$.
(2) An adversary selects $\ell_t \in \mathbb{R}^n$.

(3) The learner incurs and observes the cost $\ell_t(i_t) \in \mathbb{R}$.

In this setting, given the randomized nature of the model, we will measure the performance of an algorithm $\mathcal{A}$ using the following *expected* regret:

Definition 2 (Expected Regret)**.** *Let $\mathcal{A}$ be an algorithm for the MAB setting that generates distributions $\{x_t\}$ on the sequence of loss vectors $\{\ell_t\}$. Then the expected regret $\mathbf{E}[\mathrm{Reg}_{\mathcal{A}}(T)]$ is*

$$\mathbf{E}[\mathrm{Reg}_{\mathcal{A}}(T)] = \mathbf{E}\left[\sum_{t=1}^{T} \ell_t(i_t) - \min_{i^\star \in [n]} \sum_{t=1}^{T} \ell_t(i^\star)\right],$$

where the expectation is taken over the randomness of the samples $\{i_t \sim x_t\}$.

Remark 3. We make several remarks about the MAB setting and the expected regret definition:

- **Adversarial Choices of Losses**: First, in step (2) of the model, we assume the adversary can select ℓ_t depending on the distribution x_t but not on the realization $i_t \sim x_t$. For simplicity, we will also assume throughout that $\ell_t \in [0,1]^n$ has bounded and non-negative entries.
- **Expected Cost vs. Random Cost of the Learner**: Under the realization of the samples $\{i_t \sim x_t\}$, the learner's *random incurred cost* is the quantity $\sum_{t=1}^{T} \ell_t(i_t)$. For each t, fixing $x_t \in \Delta_n$ and taking expectation over the sample $i_t \sim x_t$, observe that $\mathbf{E}[\ell_t(i_t)] = \sum_{i=1}^{n} x_t(i)\ell_t(i) = \langle x_t, \ell_t \rangle$. This is exactly the incurred cost that we measure in the (deterministic) experts setting with full feedback. Without taking expectation and considering the random incurred cost leads to a notion of *random regret*.

2 The EXP3 Algorithm for Adversarial Multi-Armed Bandits

In this section we describe the EXP3 algorithm for the (adversarial) Multi-Armed Bandit setting. This algorithm was introduced by Auer et al. (2002) and titled *exponential weights for exploration and exploitation*. Unsurprisingly, the algorithm is based on the exponential weights method (MWU), and we attempt to develop some intuition before formally stating the algorithm and its guarantee.

2.1 Intuition for EXP3

The key challenge of the bandit feedback model is that, after sampling an action $i_t \sim x_t$, the learner only observes *a single coordinate* of the loss vector ℓ_t. In contrast, in the full-feedback experts setting, the learner chooses a distribution x_t and can see subsequently observe the entire vector ℓ_t.

In the full feedback setting, we've seen in Lecture 02 that the FTRL strategy uses the previously observed loss vectors to keep an estimate of the cost of *all* actions over the entire history of the process. When instantiated with the negative entropy regularizer, FTRL becomes the MWU method, which chooses x_{t+1} by applying exponentially-weighted updates to each coordinate i of x_t depending on the magnitude of the feedback $\ell_t(i)$.

In the MAB setting, the learner does not have enough feedback to make these updates at every coordinate of the distribution x_{t+1}. However, one natural idea is to instead construct an estimator $\hat{\ell}_t \in \mathbb{R}^n$ of the vector ℓ_t, and to apply the multiplicative updates to x_t using the coordinates of this estimated loss vector.

The EXP3 algorithm precisely employs this strategy using the following estimator $\hat{\ell}_t \in \mathbb{R}^n$ with coordinates:

$$\hat{\ell}_t(i) = \begin{cases} \ell_t(i)/x_t(i) & \text{if } i_t = i \\ 0 & \text{otherwise} \end{cases}. \tag{1}$$

A key property of $\hat{\ell}_t$ is that it is an *unbiased* estimator of ℓ_t:

Proposition 4. *Fix $x_t \in \Delta_n$, and suppose $i_t \sim x_t \in [n]$. Let $\hat{\ell}_t \in \mathbb{R}^n$ be the estimator whose coordinates are given in (1). Then $\mathbf{E}[\hat{\ell}_t] = \ell_t$.*

Proof. Fix $i \in [n]$. Then by definition of ℓ_t, we can compute over the randomness of $i_t \sim x_t$:

$$\mathbf{E}[\hat{\ell}_t(i)] = 0 \cdot \Pr[i_t = i] + \frac{\ell_t(i)}{x_t(i)} \cdot \Pr[i_t = i] = \frac{\ell_t(i)}{x_t(i)} \cdot x_t(i) = \ell_t(i) \,.$$

□

2.2 Algorithm Description and Regret Guarantee

We now formally state the EXP3 algorithm for the Adversarial Multi-Armed Bandit setting:

Algorithm 1 EXP3 for Multi Armed Bandits

Input: Stepsize parameter $\eta > 0$; Initialize $w_1 = (1, \ldots, 1) \in \mathbb{R}^n$ and $x_1 = (\frac{1}{n}, \ldots, \frac{1}{n}) \in \Delta_n$.
for $t = 1, \ldots, T$ **do**:

1. Sample $i_t \sim x_t$, and incur cost $\ell_t(i_t)$. Observe bandit feedback $\ell_t(i_t)$.
2. Construct estimator $\hat{\ell}_t \in \mathbb{R}^n$ with coordinate $i \in [n]$ given by

$$\hat{\ell}_t(i) = \begin{cases} \ell_t(i_t)/x_t(i_t) & \text{if } i = i_t \\ 0 & \text{otherwise} \end{cases} \,. \tag{2}$$

3. Update $w_{t+1} \in \mathbb{R}^n$ with coordinates $i \in [n]$ given by

$$w_{t+1}(i) = w_t(i) \cdot \exp(-\eta \hat{\ell}_t(i_t)) \,. \tag{3}$$

4. Set $x_{t+1} \in \Delta_n$ to be

$$x_{t+1} = \frac{w_{t+1}}{\|w_{t+1}\|_1} \,. \tag{4}$$

end for

Remark 5. We make several remarks about the EXP3 algorithm:

- **EXP3 as MWU on Estimated Loss Vectors**: As mentioned above, EXP3 is exactly the result of running the Multiplicative Weights Update (MWU) algorithm for the experts setting on the sequence of *estimated* loss vectors $\{\hat{\ell}_t\}$. Indeed, the regret guarantee we establish for EXP3 will be proven using this perspective.
- **EXP3 with Uniform Mixing**: The original description of the EXP3 algorithm of Auer et al. (2002) includes a *uniform mixing* term of the form $\hat{x}_{t+1} = (1-\gamma)x_{t+1} + \gamma/n$ for some $\gamma \in [0,1]$ (and each $i_{t+1} \sim \hat{x}_{t+1}$). Such additional mixing helps obtain *high probability* bounds on regret, as opposed to bounds on expected regret (see the remarks below for more discussion).

We now state the regret guarantee for EXP3:

Theorem 6. *Let $\{i_t\}$ be the sequence of actions played by EXP3 (Algorithm 1) with setpsize $\eta > 0$ on a sequence of loss vectors $\{\ell_t\}$ in the MAB setting with each $\ell_t \in [0,1]^n$. Then for any $T \geq 1$, setting $\eta := \sqrt{\frac{\log n}{nT}}$, we have over the randomness of the algorithm that*

$$\mathbf{E}\left[\mathrm{Reg}_{EXP3}(T)\right] \leq 2\sqrt{T \cdot n \log n} \,.$$

Remark 7. We make several remarks about the regret guarantee for EXP3:

- **Comparison with full-feedback regret bounds**: Recall that in the full-feedback experts setting, MWU (an instantiation of FTRL) obtains regret $\text{Reg}_{MWU}(T) \leq 2\sqrt{T \log n}$. In the bandit setting, Theorem 6 shows that one can obtain an *expected* regret with the same (optimal) dependence on T, and only an additional $\sqrt{n}$ multiplicative dependence on the size of the action space. In general, this $\sqrt{n}$ dependence is tight (see Lattimore and Szepesvári (2020, Chapter 13)).

  Variance of EXP3 can be large: The theorem proves a bound on the expected regret, but it is well known that the *variance* of EXP3 can be large. In particular, the *random regret* of EXP3 can scale linearly in T with at least constant probability (see Lattimore and Szepesvári (2020, Note 1, Section 11.5)). The uniform mixing component (Bullet 2 of Remark 5) can be used to obtain tighter, probabilistic bounds on the random regret.

2.3 Proof of Expected Regret Bound

We now develop the proof of Theorem 6. As mentioned, we will analyze EXP3 through the lens of running the MWU (in the experts setting) on the sequence of estimated loss vectors $\{\hat{\ell}t\}$. We make this reduction precise as follows:

Reduction to MWU on Estimated Losses. By using the property that each vectors ℓ_t is an unbiased estimator of ℓ_t (e.g., Proposition 4), we establish the following reduction:

Proposition 8. *Let $\{x_t\}$ and $\{i_t\}$ be the sequence of distributions and sampled actions generated by the EXP3 algorithm on loss vectors $\{\ell_t\}$. Then*

$$\mathbf{E}\left[\text{Reg}_{EXP3}(T)\right] = \mathbf{E}\left[\sum_{t=1}^{T}\langle x_t, \hat{\ell}_t\rangle - \min_{x\in\Delta_n}\sum_{t=1}^{T}\langle x, \hat{\ell}_t\rangle\right].$$

Proof. First, let $i^\star \in \text{argmax}_{i\in[n]}\sum_{t=1}^{T}\ell_t(i)$, and recall for the MAB setting that the expected regret is given by

$$\mathbf{E}\left[\text{Reg}_{EXP3}(T)\right] = \mathbf{E}\left[\sum_{t=1}^{T}\ell_t(i_t) - \sum_{t=1}^{T}\ell_t(i^\star)\right] = \mathbf{E}\left[\sum_{t=1}^{T}\ell_t(i_t)\right] - \sum_{t=1}^{T}\ell_t(i^\star), \tag{5}$$

where the second equality follows by the linearity of expectation. We will simplify the two terms of (5) separately:

- **Simplifying the first term of (5)**: First, observe by the linearity of expectation that

$$\mathbf{E}\left[\sum_{t=1}^{T}\ell_t(i_t)\right] = \sum_{t=1}^{T}\mathbf{E}\left[\ell_t(i_t)\right] = \sum_{t=1}^{T}\mathbf{E}\left[\mathbf{E}_{t-1}\left[\ell_t(i_t)\right]\right]. \tag{6}$$

  Here, we write $\mathbf{E}_{t-1}[\cdot]$ to denote a *conditional expectation* on the history of the algorithm through round $t-1$ and use the towering property of conditional expectation[1]. Then by definition of the algorithm, we have

$$\mathbf{E}_{t-1}\left[\ell_t(i_t)\right] = \sum_{i=1}^{n}x_t(i)\cdot\ell_t(i) = \sum_{i=1}^{n}x_t(i)\cdot\mathbf{E}_{t-1}[\hat{\ell}_t(i)] \tag{7}$$

$$= \sum_{i=1}^{n}\mathbf{E}_{t-1}[x_t(i)\cdot\hat{\ell}_t(i)] = \mathbf{E}_{t-1}\left[\langle x_t, \hat{\ell}_t\rangle\right]. \tag{8}$$

[1]For a random variable X, we have $\mathbf{E}[\mathbf{E}_{t-1}[X]] = \mathbf{E}[X]$. See Lattimore and Szepesvári (2020, Section 11.4).

Here, in the second equality we apply the ubiased estimator property of Proposition 4, in the third equality we use the fact that, conditioned on the randomness through round $t-1$, the distribution $x_t \in \Delta_n$ is fixed and each $x_t(i)$ is a constant, and in the final equality we use the linearity of expectation. Again using the towering property of conditional expectation, we have

$$\mathbf{E}\left[\,\mathbf{E}_{t-1}[\ell_t(i_t)]\right] = \mathbf{E}\left[\,\mathbf{E}_{t-1}[\langle x_t, \hat{\ell}_t\rangle]\right] = \mathbf{E}[\langle x_t, \hat{\ell}_t\rangle]\,. \tag{9}$$

Substituting this into (6) and using the linearity of expectation, we have

$$\mathbf{E}\left[\sum_{t=1}^{T} \ell_t(i_t)\right] = \mathbf{E}\left[\sum_{t=1}^{T}\langle x_t, \hat{\ell}_t\rangle\right]. \tag{10}$$

- **Simplifying the second term of** (5): For the second term, we have again by Proposition 4 and the linearity of expectation that

$$\sum_{t=1}^{T} \ell_t(i^\star) = \sum_{t=1}^{T} \mathbf{E}[\hat{\ell}_t(i^\star)] = \mathbf{E}\left[\sum_{t=1}^{T} \hat{\ell}_t(i^\star)\right] = \mathbf{E}\left[\min_{x\in\Delta_n}\sum_{t=1}^{T}\langle x, \hat{\ell}_t\rangle\right], \tag{11}$$

where the last equality follows by definition of $i^\star$.

Combining expressions (10) and (11), we conclude

$$\mathbf{E}\left[\mathrm{Reg}_{EXP3}(T)\right] = \mathbf{E}\left[\sum_{t=1}^{T}\langle x_t, \hat{\ell}_t\rangle - \min_{x\in\Delta_n}\sum_{t=1}^{T}\langle x, \hat{\ell}_t\rangle\right]. \qquad \square$$

Running MWU on Estimated Losses. The punchline of Proposition 8 is that the expected regret of EXP3 exactly reduces to the (expected) regret of running MWU on the estimated loss sequence $\{\hat{\ell}t\}$. We will denote this quantity by $\widehat{R}_{MWU}(T)$, where

$$\widehat{R}_{MWU}(T) = \sum_{t=1}^{T}\langle x_t, \hat{\ell}_t\rangle - \min_{x\in\Delta_n}\sum_{t=1}^{T}\langle x, \hat{\ell}_t\rangle\,,$$

and thus Proposition 8 says $\mathbf{E}[\mathrm{Reg}_{EXP3}(T)] = \mathbf{E}[\widehat{R}_{MWU}(T)]$.

Now recall from Lecture 03 that MWU is an instantiation of FTRL for the full-feedback experts setting using the negative entropy regularizer (Proposition 17 from Lecture Notes 02). Moreover, we thus have from Theorems 11 and 18 of Lecture Notes 02 that

$$\widehat{R}_{MWU}(T) \le \eta\sum_{t=1}^{T}\|\hat{\ell}_t\|_\infty^2 + \frac{\log n}{\eta}\,. \tag{12}$$

In the MAB setting, we assume the true loss vectors $\ell_t \in [0,1]^n$ have bounded coordinates, and thus $\|\ell_t\|_\infty^2 \le 1$. On the other hand, the estimators $\hat{\ell}_t$ in general can have unbounded coordinates. For example, for $i := i_t$, we have $\hat{\ell}_t(i) = \ell_t(i)/x_t(i) \le 1/x_t(i)$, which explodes as x_t grows small. Thus, the generic regret bound for MWU in (12) (stemming from the FTRL analysis) is somewhat unsatisfactory when applying the result to the sequence $\{\ell_t\}$.

However, it turns out that a more direct analysis of MWU via a potential function argument yields the following regret bound that is more amenable for this task:

Theorem 9. *Let $\{x_t\}$ be the iterates of MWU (Algorithm 1 from Lecture Notes 02) with stepsize $\eta > 0$ on a sequence of loss vectors $\{\hat{\ell}_t\}$ such that $\ell_t(i) \ge 0$ for all $i \in [n]$ and $t \ge 1$. Then:*

$$\mathrm{Reg}_{MWU}(T) \le \eta\cdot\sum_{t=1}^{T}\langle x_t, \hat{\ell}_t^2\rangle + \frac{\log n}{\eta}\,, \tag{13}$$

where $\hat{\ell}_t^2 = (\hat{\ell}_t(1)^2, \ldots \hat{\ell}_t(n)^2) \in \mathbb{R}^n$.

Equipped with this result, we can now prove the regret bound for EXP3 from Theorem 6:

Proof of Theorem 6. Combining Proposition 8 and Theorem 9, we have

$$\mathbf{E}\left[\mathrm{Reg}_{EXP3}(T)\right] = \mathbf{E}\left[\widehat{R}_{MWU}(T)\right] \leq \mathbf{E}\left[\eta \cdot \sum_{t=1}^{T} \langle x_t, \hat{\ell}_t^2 \rangle + \frac{\log n}{\eta}\right]. \tag{14}$$

Now by definition of $\hat{\ell}_t$, we have by applying Proposition 4 and the towering property that

$$\mathbf{E}\left[\langle x_t, \hat{\ell}_t^2 \rangle\right] = \mathbf{E}\left[\sum_{i=1}^{n} x_t(i) \cdot \hat{\ell}_t(i)^2\right] = \sum_{i=1}^{n} \mathbf{E}\left[x_t(i) \cdot \mathbf{E}_{t-1}\left[\hat{\ell}_t(i)^2\right]\right]. \tag{15}$$

For each $i \in [n]$, we can further compute by construction of $\hat{\ell}_t$ that

$$\mathbf{E}_{t-1}\left[\hat{\ell}_t(i)^2\right] = \Pr_{i_t \sim x_t}[i_t = i] \cdot (0)^2 + \Pr_{i_t \sim x_t}[i_t = i] \cdot (\ell_t(i)/x_t(i))^2 = \ell_t(i)^2 / x_t(i).$$

Substituting this back into (15), we find

$$\mathbf{E}\left[\langle x_t, \hat{\ell}_t^2 \rangle\right] = \sum_{i=1}^{n} \mathbf{E}[x_t(i) \cdot (\ell_t(i)^2 / x_t(i))] = \sum_{i=1}^{n} \mathbf{E}[\ell_t(i)^2] \leq n, \tag{16}$$

where the final inequality comes from the assumption that each $\ell_t \in [0,1]^n$. Thus the bound in (14) becomes

$$\mathbf{E}\left[\mathrm{Reg}_{EXP3}(T)\right] \leq \eta \cdot Tn + \frac{\log n}{\eta}.$$

Setting $\eta := \sqrt{\frac{\log n}{nT}}$ then yields $\mathbf{E}\left[\mathrm{Reg}_{EXP3}(T)\right] \leq 2\sqrt{T \cdot n \log n}$. □

References

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.