CS272 - Theoretical Foundations of Learning, Decisions, and Games

Lecture 11: Bandits and Intro to Zero-sum Games

October 2, 2025

Lecturer: Nika Haghtalab *Readings: Section 18.3, 20 Lectures on AGT*
Scribe: Hengyu Fu, Hangxin Gan

1 EXP3 algorithm

1.1 Recap: Standard EXP3 (adversarial bandits)

The EXP3 (Exponential-weight for Exploration and Exploitation) algorithm addresses adversarial multi-armed bandits. Suppose there are n actions and at each round $t = 1, \ldots, T$ the learner picks an action i_t and incurs a loss $c^t(i_t) \in [0, 1]$ assigned by an adversary. The algorithm is shown in Algorithm 1.

Algorithm 1 Old School EXP3 Algorithm

Input: Number of experts n, number of rounds T, exploration parameter γ
Initialize: $w_i^1 = 1$ for all $i \in [n]$
Set learning rate $\eta = \gamma/n$
for $t = 1, \ldots, T$ **do**
    Form normalized weights:

$$p^t(i) = \frac{w_i^t}{\sum_{j=1}^n w_j^t}, \qquad i = 1, \ldots, n$$

    Form sampling probabilities:

$$q^t(i) = (1 - \gamma)\, p^t(i) + \frac{\gamma}{n}, \qquad i = 1, \ldots, n$$

    Sample $i_t \sim q^t$ and observe cost $c^t(i_t) \in [0, 1]$
    Form unbiased loss estimate:

$$\hat{c}^t(i) = \begin{cases} \dfrac{c^t(i_t)}{q^t(i_t)}, & i = i_t, \\ 0, & \text{otherwise} \end{cases}$$

    Update weights:

$$w_i^{t+1} = w_i^t \exp\Big(-\eta\, \hat{c}^t(i)\Big) = w_i^t \exp\Big(-\tfrac{\gamma}{n}\, \hat{c}^t(i)\Big)$$

end for

The following theorem provides the regret guarantee for the standard EXP3 Algorithm.

Theorem 1.1 (Regret Guarantee of Algorithm 1)**.** *With a common tuning $\gamma = \Theta\big(\sqrt{\frac{n \ln n}{T}}\big)$, the expected regret of the EXP3 Algorithm (Algorithm 1) satisfies*

$$\mathbb{E}\Big[\sum_{t=1}^{T} c^t(i_t)\Big] \;-\; \min_i \sum_{t=1}^{T} c^t(i) \;=\; O\Big(\sqrt{Tn \ln n}\Big).$$

A variant of EXP3 (e.g., EXP3.P / EXP3-IX) uses slightly different scaling and a small bias in the estimates to obtain high-probability bounds; in expectation the same $O(\sqrt{nT \ln n})$ rate is preserved.

We now present the reward/utility-oriented variant of EXP3, which we will regularly use in game-theory contexts (Algorithm 2).

Algorithm 2 Old School EXP3 — Reward Version

Input: Number of arms n, number of rounds T, exploration parameter $\gamma \in (0, 1)$
Initialize: $w_i^1 = 1$ for all $i \in [n]$
for $t = 1, \ldots, T$ **do**
    Compute normalized weights: $p^t(i) = w_i^t / \sum_{j=1}^n w_j^t$
    Set exploration probabilities:

$$q^t(i) = (1 - \gamma)\, p^t(i) + \frac{\gamma}{n}$$

    Sample $i^t \sim q^t$ and observe utility $u^t(i^t) \in [0, 1]$
    Form importance-weighted estimator:

$$\hat{u}^t(i) = \begin{cases} \dfrac{u^t(i^t)}{q^t(i^t)}, & i = i^t, \\ 0, & i \neq i^t \end{cases}$$

    Update weights:

$$w_i^{t+1} = w_i^t \exp\Big(\gamma \hat{u}^t(i)/n\Big)$$

end for

Theorem 1.2 (Regret Guarantee for Reward Variant)**.** *Algorithm 2 guarantees*

$$\mathbb{E}\Big[\sum_{t=1}^{T} u^t(i^t)\Big] \geq (1 - \gamma) \max_j \sum_{t=1}^{T} u^t(j) - \frac{n}{\gamma} \ln n.$$

It follows that $\textsc{Regret}\big(\text{EXP3}\big) \leq O\left(\sqrt{Tn \ln n}\right)$ *if* $\gamma = \sqrt{\frac{n \ln n}{T}}$.

2 A Simple EXP3-Style Algorithm with Weaker Guarantees

In homework, you are free to use the above two theorems without proving them from scratch. Their proofs are very similar to the proof of MWU, so we will not repeat them here. You can refer to Auer et al. [2002] for a detailed analysis.

Below, we present a similar algorithm—one that differs only in the parameterization of the update step. We prove that such an algorithm achieves a $T^{2/3}(n \ln n)^{1/3}$ regret guarantee. While this is certainly not as good as the regret in the theorems above, presenting this algorithm gives us an opportunity to review an important algorithm-design approach: reductions!

In what follows, we show that any no-regret algorithm for the full-information setting can be adapted to the bandit setting as well, albeit with slightly worse regret guarantees. For ease of presentation, we use the Randomized Weighted Majority (RWM) algorithm, which is no-regret in the full-information setting. The high-level idea for this algorithm is as follows:

At iteration t,

- Use probabilities suggested by a no-regret algorithm in the full-information setting (such as RWM). Denote these probabilities by the vector p^t.

  Sample an expert i^t from a distribution q^t, where $q^t = (1-\gamma)\,p^t + \gamma(\frac{1}{n}, \ldots, \frac{1}{n})$. This is equivalent to taking $i^t \sim p^t$ with probability $(1-\gamma)$ and taking i^t uniformly from $[n]$ with probability γ.

- Let the full-information algorithm update the experts' weights. To do so, the algorithm needs to see a utility of every expert. So, we construct a utility vector as follows:

$$\hat{u}^t = \left(0, \ldots, 0, \frac{u^t(i^t)}{q^t(i^t)}, 0, \ldots, 0\right).$$

  Here $u^t(i^t)$ is the utility of the expert i^t chosen at time t and $q^t(i^t)$ is the probability of choosing i^t. The vector $\hat{u}^t$ is then passed to the full-information algorithm (such as RWM) to update the weights accordingly. As we will show below, these man-made utilities actually make sense! That is, they are unbiased estimators for the utility of each expert. The version of RWM described earlier assumes utilities in the range $[0, 1]$. Here, the constructed utility is $\frac{u^t(i^t)}{q^t(i^t)} \in [0, n/\gamma]$. Therefore, we adapt the regret bound of RWM to scale with this larger range. This results in the update rule of $w_i^t(1 + \epsilon\gamma\hat{u}^t(i)/n)$ — as opposed to $w_i^t(1 + \epsilon\hat{u}^t(i))$ for $[0, 1]$ utilities.)

This high level idea is summarized in the following figure

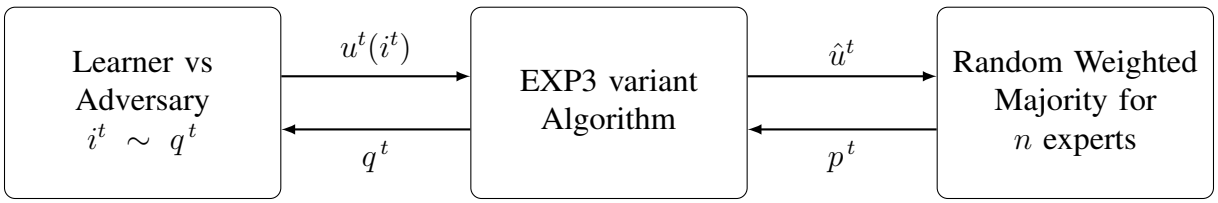

Figure 1: High level idea for EXP3 variant

We have the following regret guarantee for the variant of EXP3.

Algorithm 3 Reduction to MWU

Input: Number of arms n, number of rounds T, exploration parameter $\gamma \in (0, 1)$, and parameter $\epsilon > 0$.

Initialize: $w_i^1 = 1$ for all $i \in [n]$

for $t = 1, \ldots, T$ **do**

Compute normalized weights: $p^t(i) = w_i^t / \sum_{j=1}^n w_j^t$

Set exploration probabilities:

$$q^t(i) = (1 - \gamma)\, p^t(i) + \frac{\gamma}{n}$$

Sample $i^t \sim q^t$ and observe utility $u^t(i^t) \in [0, 1]$

Form importance-weighted estimator:

$$\hat{u}^t(i) = \begin{cases} \dfrac{u^t(i^t)}{q^t(i^t)}, & i = i^t, \\ 0, & i \neq i^t \end{cases}$$

Update weights multiplicatively:

$$w_i^{t+1} = w_i^t \Big(1 + \epsilon\gamma \hat{u}^t(i)/n\Big)$$

end for

Theorem 2.1 (Regret guarantee for Algorithm 3)**.** *For $\epsilon = \gamma$, Algorithm 3 guarantees that*

$$\mathbb{E}\left[\sum_{t=1}^T u^t(i^t)\right] \;\geq\; (1-\gamma)^2 \max_j \sum_{t=1}^T u^t(j) \;-\; \frac{n}{\gamma^2} \ln n.$$

It follows that this algorithm has regret $\big(T^{2/3}(n \ln n)^{1/3}\big)$ when $\gamma = (n \ln n / T)^{1/3}$.

2.1 Proof of Theorem 2.1

We prove the theorem by combining the following facts.

Fact 2.2 (Unbiased estimation)**.** *For all $j \in [n]$, $\hat{u}^t(j)$ is an unbiased estimator of $u^t(j)$:*

$$\mathop{\mathbb{E}}_{i^t \sim q^t}\big[\hat{u}^t(j)\big] = u^t(j).$$

Proof.

$$\mathop{\mathbb{E}}_{i^t \sim q^t}\big[\hat{u}^t(j)\big] = q^t(j) \cdot \frac{u^t(j)}{q^t(j)} + \big(1 - q^t(j)\big) \cdot 0 = u^t(j).$$

□

Let $\mathrm{OPT}_{\mathrm{RWM}} = \max_j \sum_{t=1}^T \hat{u}^t(j)$ be the best utility in hindsight according to RWM, and let $\mathrm{OPT} = \max_j \sum_{t=1}^T u^t(j)$ be the true best-in-hindsight utility. Next we show that $\mathrm{OPT}_{\mathrm{RWM}}$ is at least as large in expectation.

Fact 2.3 (OPT in RWM is more competitive than OPT)**.**

$$\mathop{\mathbb{E}}_{i^t \sim q^t}[\text{OPT}_{\text{RWM}}] \geq \text{OPT}.$$

Proof. By Jensen's inequality and Fact 2.2,

$$\mathop{\mathbb{E}}_{i^t \sim q^t}[\text{OPT}_{\text{RWM}}] = \mathop{\mathbb{E}}_{i^t \sim q^t}\left[\max_j \sum_t \hat{u}^t(j)\right] \geq \max_j \mathop{\mathbb{E}}_{i^t \sim q^t}\left[\sum_t \hat{u}^t(j)\right] = \max_j \sum_t u^t(j) = \text{OPT}.$$

□

Fact 2.4 (Regret of RWM under scaling)**.**

$$\sum_{t=1}^{T}\left(p^t \cdot \hat{u}^t\right) \;\geq\; (1-\epsilon)\,\text{OPT}_{\text{RWM}} \;-\; \frac{n}{\epsilon\,\gamma}\,\ln n.$$

Proof. This follows from the standard RWM analysis (with utilities instead of costs). The constructed utilities lie in $\left[0, \frac{n}{\gamma}\right]$ rather than $[0,1]$, leading to the multiplicative scaling in the last term. □

Fact 2.5 (True reward vs. RWM's expected reward)**.** *For each t, if $i = i^t$,*

$$u^t(i) \;=\; \hat{u}^t(i)\,q^t(i) \;=\; p^t(i)\,\hat{u}^t(i)\,\frac{q^t(i)}{p^t(i)} \;\geq\; (1-\gamma)\,p^t(i)\,\hat{u}^t(i).$$

Proof. Since $q^t(i) = (1-\gamma)\,p^t(i) + \gamma/n$, we have $q^t(i)/p^t(i) \geq 1-\gamma$. □

Proof of Theorem 2.1. Combining the facts, the bandit algorithm's reward satisfies

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} u^t(i^t)\right] &\geq (1-\gamma)\;\mathbb{E}\left[\sum_{t=1}^{T} p^t(i^t)\hat{u}^t(i^t)\right] && \text{(Fact 2.5)}\\
&\geq (1-\gamma)(1-\epsilon)\;\mathbb{E}\left[\text{OPT}_{\text{RWM}}\right] \;-\; (1-\gamma)\,\frac{n}{\epsilon\,\gamma}\,\ln n && \text{(Fact 2.4)}\\
&\geq (1-\gamma)^2\,\text{OPT} \;-\; \frac{n}{\gamma^2}\,\ln n, && \text{(Fact 2.3), set } \epsilon = \gamma
\end{aligned}$$

which proves the claim. □

3 Game Theory

3.1 Preliminaries

We introduce the setting of a game of m players.

- An m-player normal-form game has players $1, \ldots, m$ with action sets $A_1, \ldots, A_m$.
- For pure profile $(a_1, \ldots, a_m)$, player i receives utility $u_i(a_1, \ldots, a_m)$.
- A mixed strategy for player i is a distribution $p_i \in \Delta(A_i)$. The expected utility under mixed strategies $(p_1, \ldots, p_m)$ is

$$U_i(p_1, \ldots, p_m) \;=\; \mathbb{E}_{a_j \sim p_j}\big[u_i(a_1, \ldots, a_m)\big].$$

3.2 Two-player Zero-sum Games

For two players, the game is zero-sum if $u_1(a_1, a_2) = -u_2(a_1, a_2)$ for all outcomes. Represent the row player's payoffs by a matrix $U = [u_{ij}]$. If the row player uses distribution p and the column player uses q, the expected payoff to the row player is

$$p^\top U q,$$

and the column player's payoff is $-p^\top U q$.

Example: Rock–Paper–Scissors

Below is the payoff matrix of Rock–Paper–Scissors (row player payoff). Rows are row player's actions; columns are column player's actions.

Table 1: Rock–Paper–Scissors payoff matrix (row player).

	Rock	Paper	Scissors
Rock	0	-1	1
Paper	1	0	-1
Scissors	-1	1	0

This is a zero-sum matrix: each entry is the negative of the corresponding column-player payoff.

3.3 The Minimax Theorem

We introduce the minimax theorem in zero-sum games.

Theorem 3.1 (The Minimax Theorem)**.** *For every finite two-player zero-sum game with payoff matrix U,*

$$\max_p \min_q \; p^\top U q \;=\; \min_q \max_p \; p^\top U q.$$

Proof via No-Regret Learning. The inequality "$\leq$" is immediate: The player going second does a better job of achieving their objective since they can adapt to the strategy of the first player.

For the reverse inequality, we will prove this via online learning. We consider a T-step interactions, where at time t, the row player player strategy p_t and the column player plays q_t. The row player uses a no-regret algorithm, i.e., give the history of distributions $q_1, \ldots, q_{t-1}$ that the column player has played so far, the row player runs a no-regret algorithm to choose p^t in order to maximize utilities it receives $U(p^t, q^t)$ in a no-regret manner. Then the column player chooses $q^t = \arg\min_q U(p^t, q^t)$. Let $\bar{p} = (1/T)\sum_{t=1}^T p^t$ and let $\bar{q} = (1/T)\sum_{t=1}^T q^t$. Then we have

$$\max_p \min_q U(p, q) \geq \min_q U(\bar{p}, q) \tag{1}$$

$$= \min_q \frac{1}{T} \sum_{t=1}^T U(p^t, q) \tag{2}$$

$$\geq \frac{1}{T}\sum_{t=1}^{T}\min_{q} U(p^t, q) \tag{3}$$

$$= \frac{1}{T}\sum_{t=1}^{T} U(p^t, q^t) \qquad \text{(By Def of } q^t\text{)} \tag{4}$$

$$\geq \frac{1}{T}\max_{p}\sum_{t=1}^{T} U(p, q^t) - \frac{Regret}{T} \qquad \text{(By } p^t\text{s being no-regret)} \tag{5}$$

$$\geq \max_{p} U(p, \bar{q}) - \frac{Regret}{T} \tag{6}$$

$$\geq \max_{p}\min_{q} U(p, q) - \frac{Regret}{T} \tag{7}$$

$$\tag{8}$$

Note that because the game is finite, the row player indeed has a no-regret algorithm with regret $O\left(\sqrt{T\ln(n)}\right)$. So, as $T \to \infty$, $\frac{\text{Regret}}{T} \to 0$. This yields the proof. □

References

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