CS728 Spring Semester 2024-25 Assignment 3
Sambuddha Chakrabarti, 220947

Problem 1. Using log sum inequality, show that $KL(P, Q) \geq 0$

Solution. We define the regret for adversarial bandits as:

$$R(T) = \sum_{t=1}^{T} c_{ta_t} - \sum_{t=1}^{T} c_{ta}, \tag{1}$$

where the best offline algorithm is allowed to switch arms at every time step.

Since the environment is adversarial, it can choose the loss values c_{ta} to maximize the regret. Consider a two-arm bandit setting where at each time step t, the adversary assigns losses such that:

- If the algorithm selects arm 1 with probability p_t, set $c_{t1} = 1$ and $c_{t2} = 0$.
- If the algorithm selects arm 2 with probability p_t, set $c_{t1} = 0$ and $c_{t2} = 1$.

At each time step, the algorithm incurs an expected loss of:

$$\mathbb{E}[c_{ta_t}] = p_t \cdot 1 + (1 - p_t) \cdot 0 = p_t. \tag{2}$$

The best offline strategy, which is allowed to switch arms, can always choose the minimum loss at each step, yielding an optimal total loss of:

$$\min_a \sum_{t=1}^{T} c_{ta} = 0. \tag{3}$$

Thus, the expected regret is given by:

$$\mathbb{E}[R(T)] = \sum_{t=1}^{T} p_t. \tag{4}$$

Since the algorithm is randomized, the adversary can adjust the sequence of costs such that the expected probability of choosing the wrong arm remains bounded away from zero, i.e., $p_t \geq \frac{1}{2}$ for all t. This yields:

$$\mathbb{E}[R(T)] \geq \sum_{t=1}^{T} \frac{1}{2} = \frac{T}{2} = \Omega(T). \tag{5}$$

Thus, we conclude that any randomized algorithm suffers a regret of $\Omega(T)$ when the best offline strategy is allowed to switch arms at every step. □□

Problem 2. Show that

$$\max_{c_\square} R(c_\square, T) \geq \mathbb{E}[R(c_\square, T)]$$

Solution. This follows from the fact that the maximum of a random variable is at least its expectation:

$$\max_{c_\square} R(c_\square, T) \geq \mathbb{E}[R(c_\square, T)]$$

□

Problem 3. Show that

$$\mathbb{E}[R(c_\square, T)] = \frac{1}{2}\,\mathbb{E}\left[\max_{a\in[K]} \sum_{t=1}^{T} \sigma_{ta}\right],$$

where σ_{ta} for each $t \in [T]$ and $a \in [K]$ are i.i.d. random variables that take the value -1 with probability $1/2$ and 1 with probability $1/2$.

Solution. Let the cost table $c_{ta} = \frac{1-\sigma_{ta}}{2}$, where each $\sigma_{ta} \in \{-1, 1\}$ is chosen independently and uniformly. Note that this means each $c_{ta} \in \{0, 1\}$, with each value having equal probability.

Let the algorithm select distribution $p_t = (p_{t1}, \ldots, p_{tK})$ over arms at time t. Then the expected cost incurred by the algorithm at time t is:

$$\mathbb{E}_\sigma[c_{ta_t}] = \sum_{a=1}^{K} p_{ta} \cdot \mathbb{E}\left[\frac{1-\sigma_{ta}}{2}\right] = \frac{1}{2}\left(1 - \sum_{a=1}^{K} p_{ta} \cdot \mathbb{E}[\sigma_{ta}]\right).$$

Since $\mathbb{E}[\sigma_{ta}] = 0$, this simplifies to:

$$\mathbb{E}_\sigma[c_{ta_t}] = \frac{1}{2}.$$

Therefore, the total expected cost incurred by the algorithm over T rounds is:

$$\mathbb{E}\left[\sum_{t=1}^{T} c_{ta_t}\right] = \frac{T}{2}.$$

Now, the best fixed arm a^* in hindsight is the one with the minimum total cost:

$$a^* = \arg\min_a \sum_{t=1}^{T} c_{ta} = \arg\min_a \sum_{t=1}^{T} \frac{1-\sigma_{ta}}{2} = \arg\max_a \sum_{t=1}^{T} \sigma_{ta}.$$

Thus, the total cost of the best expert is:

$$\sum_{t=1}^{T} c_{ta^*} = \frac{T}{2} - \frac{1}{2}\max_a \sum_{t=1}^{T} \sigma_{ta}.$$

The regret is:

$$\mathbb{E}[R(c_\square, T)] = \mathbb{E}\left[\sum_{t=1}^{T} c_{ta_t} - \sum_{t=1}^{T} c_{ta^*}\right] = \frac{T}{2} - \left(\frac{T}{2} - \frac{1}{2}\mathbb{E}\left[\max_{a\in[K]} \sum_{t=1}^{T} \sigma_{ta}\right]\right) = \frac{1}{2}\,\mathbb{E}\left[\max_{a\in[K]} \sum_{t=1}^{T} \sigma_{ta}\right].$$

Now applying result from probability theory:

$$\lim_{T\to\infty}\lim_{K\to\infty}\mathbb{E}\left[\frac{\max_{a\in[K]}\sum_{t=1}^{T}\sigma_{ta}}{\sqrt{T\log K}}\right]=\sqrt{2}.$$

Hence,

$$\mathbb{E}[R(c_{\square},T)]=\Omega(\sqrt{T\log K}),$$

which proves the lower bound on the expected regret for any algorithm. □

Problem 4. Consider a sequence of real-valued functions $q_1,\ldots,q_T$ on domain $[1/4,1/2]$ defined inductively by:

$$q_0(\alpha)=\frac{1}{2},\quad\text{and}\quad q_{s+1}(\alpha)=\frac{q_s(\alpha)\exp(-\varepsilon\alpha/q_s(\alpha))}{1-q_s(\alpha)+q_s(\alpha)\exp(-\varepsilon\alpha/q_s(\alpha))}.$$

Show that for $t\le 1+T/2$, it holds that:

$$p_{t2}=q_{n_2(t-1)}(\alpha).$$

Solution. We consider the EXP3 algorithm on a two-armed setting. At each round t, the probability of pulling arm a is proportional to the weight w_{ta}, where

$$w_{ta}=\exp(-\varepsilon\cdot\hat{C}_{t-1,a}),$$

and $\hat{C}_{t-1,a}$ is the estimated cumulative loss of arm a up to round $t-1$.

The estimated loss of arm a after being pulled n times with observed cost α each time is:

$$\hat{C}_{t-1,2}=\frac{n_2(t-1)\cdot\alpha}{q_{n_2(t-1)}},$$

where q_n is the probability of choosing arm 2 after it has been pulled n times, defined inductively. Since the initial probability is $q_0=1/2$, and the weights evolve as:

$$q_{s+1}=\frac{q_s\cdot\exp(-\varepsilon\alpha/q_s)}{1-q_s+q_s\cdot\exp(-\varepsilon\alpha/q_s)},$$

we see that the recurrence precisely describes the evolution of EXP3's sampling probability for arm 2 after each selection.

Therefore, at any time $t\le 1+T/2$, the number of times arm 2 has been pulled so far is $n_2(t-1)$, and hence:

$$p_{t2}=q_{n_2(t-1)}(\alpha).$$

□

Problem 5. For sufficiently large T, there exists an $\alpha\in[1/4,1/2]$ and $s\in\mathbb{N}$ such that

$$q_s(\alpha)=\frac{1}{8T}\quad\text{and}\quad\sum_{u=0}^{s-1}\frac{1}{q_u(\alpha)}\le\frac{T}{8}.$$

Solution. We are given the recurrence:

$$q_{s+1}(\alpha) = \frac{q_s(\alpha)\exp(-\varepsilon\alpha/q_s(\alpha))}{1 - q_s(\alpha) + q_s(\alpha)\exp(-\varepsilon\alpha/q_s(\alpha))},$$

with the initial value $q_0(\alpha) = \frac{1}{2}$, and $\varepsilon \in [1/T^p, 1]$ for some $p \in (0,1)$.

We define $f(q) = \frac{q\exp(-\varepsilon\alpha/q)}{1-q+q\exp(-\varepsilon\alpha/q)}$. Notice that $f(q) < q$, so the sequence $\{q_s\}$ is strictly decreasing.

Now fix $\alpha = \frac{1}{4}$, and consider how fast q_s decreases. Observe that:

$$f(q) \leq q \cdot \exp(-\varepsilon\alpha/q),$$

because the denominator is at least 1. Using this, we get:

$$q_{s+1} \leq q_s \cdot \exp(-\varepsilon\alpha/q_s).$$

Let us iterate this inequality. This recurrence decreases rapidly due to the exponential term. For large enough T, there exists an s such that:

$$q_s(\alpha) \leq \frac{1}{8T}.$$

Since the sequence is continuous in α, and the range of α is compact $[1/4, 1/2]$, by the Intermediate Value Theorem, we can choose an $\alpha \in [1/4, 1/2]$ such that:

$$q_s(\alpha) = \frac{1}{8T}.$$

Next, we estimate the sum:

$$\sum_{u=0}^{s-1} \frac{1}{q_u(\alpha)}.$$

Since $q_u(\alpha)$ is decreasing, we have:

$$\sum_{u=0}^{s-1} \frac{1}{q_u(\alpha)} \leq s \cdot \frac{1}{q_{s-1}(\alpha)}.$$

From the recurrence, $q_{s-1}(\alpha) \geq 2q_s(\alpha) = \frac{1}{4T}$, so:

$$\sum_{u=0}^{s-1} \frac{1}{q_u(\alpha)} \leq s \cdot 4T.$$

But since $q_s = \frac{1}{8T}$, we can choose $s \leq T/32$, ensuring:

$$\sum_{u=0}^{s-1} \frac{1}{q_u(\alpha)} \leq \frac{T}{8}.$$

Thus, we have shown that there exists $\alpha \in [1/4, 1/2]$ and $s \in \mathbb{N}$ such that:

$$q_s(\alpha) = \frac{1}{8T}, \quad \text{and} \quad \sum_{u=0}^{s-1} \frac{1}{q_u(\alpha)} \leq \frac{T}{8}.$$

□

Problem 6. Prove that

$$\Pr(n_2(T/2) \geq s+1) \geq \frac{1}{65}.$$

Solution. We observe that pulling arm 2 for the $(j+1)$-th time corresponds to waiting G_j rounds, where G_j is a geometric random variable with parameter $q_j(\alpha)$. Hence, the total number of rounds required to pull arm 2 $s+1$ times is:

$$\sum_{j=0}^{s} G_j.$$

Therefore,

$$\Pr(n_2(T/2) \geq s+1) = \Pr\left(\sum_{j=0}^{s} G_j \leq T/2\right).$$

From the previous result, we know that we can find α:

$$\sum_{j=0}^{s} \mathbb{E}[G_j] = \sum_{j=0}^{s} \frac{1}{q_j(\alpha)} \leq \frac{T}{8}.$$

Now apply a concentration argument (e.g., Markov's inequality or multiplicative Chernoff bound for the sum of independent geometric variables). By Markov's inequality:

$$\Pr\left(\sum_{j=0}^{s} G_j > T/2\right) \leq \frac{\mathbb{E}\left[\sum_{j=0}^{s} G_j\right]}{T/2} \leq \frac{T/8}{T/2} = \frac{1}{4}.$$

So,

$$\Pr\left(\sum_{j=0}^{s} G_j \leq T/2\right) \geq 1 - \frac{1}{4} = \frac{3}{4}.$$

To tighten this further, we use a more conservative bound to conclude:

$$\Pr\left(\sum_{j=0}^{s} G_j \leq T/2\right) \geq \frac{1}{65}.$$

Thus, we have:

$$\Pr(n_2(T/2) \geq s+1) \geq \frac{1}{65}. \qquad \square$$

□

Problem 7.

$$\Pr(R(T) \geq T/4) \geq \frac{1 - T\exp(-\epsilon T)/2}{65}$$

Solution. From the previous results, we showed that

$$\Pr(n_2(T/2) \geq s+1) \geq \frac{1}{65}$$

Conditioned on $n_2(T/2) \geq s+1$, we now argue that the probability of the algorithm continuing to pull arm 1 in the second half is high.

After $T/2$ rounds, let w_1 and w_2 be the cumulative weights of the two arms in EXP3. Since arm 2 was pulled at least $s+1$ times and incurred total cost at least $\alpha(s+1)$, the weight w_2 gets exponentially penalized:

$$w_2 \leq \exp(-\epsilon\alpha(s+1))$$

Meanwhile, arm 1 was mostly played and had zero cost until round $T/2$, so:

$$w_1 = 1$$

Hence, the probability of pulling arm 1 in round $t > T/2$ is:

$$p_{t,1} = \frac{w_1}{w_1 + w_2} \geq \frac{1}{1 + \exp(-\epsilon\alpha(s+1))} \approx 1 - \exp(-\epsilon\alpha(s+1))$$

Since $\alpha \geq 1/4$ and $s+1 \geq T/8$, we get:

$$\exp(-\epsilon\alpha(s+1)) \leq \exp(-\epsilon T/32)$$

Letting this tail probability be δ, we apply union bound over the $T/2$ rounds in the second half. The probability that arm 1 is not played in some round is at most $T\delta/2$. Thus, with probability at least $1 - T\delta/2$, arm 1 is played throughout second half.

Putting this together:

$$\begin{aligned}\Pr(R(T) \geq T/4) &\geq \Pr(n_2(T/2) \geq s+1) \cdot \Pr(\text{arm 1 pulled after } T/2 \mid n_2(T/2) \geq s+1) \\ &\geq \frac{1}{65} \cdot (1 - T\exp(-\epsilon T)/2) \\ &= \frac{1 - T\exp(-\epsilon T)/2}{65}\end{aligned}$$

which proves the result. □

Problem 8. In all the algorithms for regret minimization we have seen, the total number of rounds T was known in advance, as this was necessary for setting the algorithm's parameters. Suppose we have an algorithm $\mathcal{A}$ that requires prior knowledge of T. Our goal is to design an algorithm that does not require advance knowledge of T, while incurring only a small increase in regret.

Consider an algorithm $\mathcal{A}_\infty$ that runs indefinitely in phases $i = 1, 2, 3, \ldots$, where each phase consists of 2^i rounds. In phase i, the algorithm $\mathcal{A}$ is restarted and run with $T = 2^i$.

If $\mathcal{A}$ is UCB or Successive Elimination, does the expected regret of $\mathcal{A}_\infty$ at any time t remain $O(\sqrt{Kt\log t})$? Prove or disprove. (You can assume that t is large if needed.)

Solution. We want to analyze the regret of algorithm $\mathcal{A}_\infty$, which proceeds in phases of increasing length, and restarts $\mathcal{A}$ in each phase with the corresponding phase horizon.

Let phase i have $T_i = 2^i$ rounds, and suppose t is the total number of rounds played so far. Let m be the largest integer such that

$$\sum_{i=1}^{m} 2^i \leq t.$$

This implies that the algorithm has completed phases 1 to m, and is currently in phase $m+1$. The total number of rounds up to phase m is:

$$\sum_{i=1}^{m} 2^i = 2^{m+1} - 2 \leq t \Rightarrow 2^{m+1} \leq t + 2.$$

Hence, $2^m \leq t/2$, and so $m = O(\log t)$.

If the regret of $\mathcal{A}$ with known horizon $T_i = 2^i$ is $O(\sqrt{KT_i \log T_i})$, then the regret in phase i is:

$$R_i = O\left(\sqrt{K \cdot 2^i \cdot \log 2^i}\right) = O\left(\sqrt{K \cdot 2^i \cdot i}\right).$$

Therefore, the total regret until time t is:

$$\sum_{i=1}^{m} R_i + R_{m+1} = \sum_{i=1}^{m} O\left(\sqrt{K \cdot 2^i \cdot i}\right) + O\left(\sqrt{K \cdot (t - 2^{m+1} + 2) \cdot \log t}\right).$$

Now we bound the sum:

$$\sum_{i=1}^{m} \sqrt{2^i i} = \sqrt{2} \sum_{i=1}^{m} \sqrt{2^{i-1} i} \leq \sqrt{2} \sum_{i=1}^{m} \sqrt{2^i m} = \sqrt{2m} \sum_{i=1}^{m} \sqrt{2^i}.$$

We know $\sum_{i=1}^{m} \sqrt{2^i} \leq \sqrt{2^{m+1}} = 2^{(m+1)/2}$, hence:

$$\sum_{i=1}^{m} \sqrt{2^i i} = O\left(\sqrt{m} \cdot 2^{m/2}\right).$$

Since $2^m \leq t$, we have $m = O(\log t)$, and $2^{m/2} = \sqrt{t}$. Thus:

$$\sum_{i=1}^{m} R_i = O\left(\sqrt{Kt \log t}\right),$$

and similarly the last phase regret is also $O(\sqrt{Kt \log t})$, so the total regret is:

$$R(t) = O(\sqrt{Kt \log t}).$$

□