CSCI 678: Theoretical Machine Learning
Lecture 10

Fall 2024, Instructor: Haipeng Luo

1 Partial Monitoring: Algorithms and Regret Upper Bounds

In the last lecture, we introduce the general partial monitoring problem parameterized by a loss matrix $\ell \in [0,1]^{K\times d}$ and a feedback matrix $\Phi \in \Sigma^{K\times d}$ (both known), where K is the number of actions for the learner, d is the number of outcomes for the environment, and Σ is an arbitrary set of alphabets containing all possible observations for the learner. Ahead of time, the environment decides n outcomes $z_1,\dots,z_n \in [d]$. Then, for each round $t = 1,\dots,n$, the learner selects an action $a_t \in [K]$, suffers loss $\ell(a_t, z_t)$, and only observes $\Phi(a_t, z_t)$. The goal of the learner is as usual to minimize regret against the best fixed action in hindsight:

$$\text{Reg}_n = \sum_{t=1}^n \ell(a_t, z_t) - \sum_{t=1}^n \ell(a^\star, z_t) \quad \text{where } a^\star \in \operatorname*{argmin}_{a\in[K]} \sum_{t=1}^n \ell(a, z_t).$$

We then discussed the classification theorem, which classifies all partial monitoring instances into exactly four categories, based on the corresponding minimax regret:

Theorem 1 (Classification Theorem)**.** *The minimax regret of a partial monitoring problem G is (ignoring dependence on all parameters but n):*

$$\inf_{\textit{learner}} \max_{z_{1:n}} \mathbb{E}\left[\text{Reg}_n\right] = \begin{cases} 0, & \textit{if } G \textit{ has only one Pareto-optimal action;} \\ \Theta(\sqrt{n}), & \textit{else if } G \textit{ is locally observable;} \\ \Theta(n^{\frac{2}{3}}), & \textit{else if } G \textit{ is globally observable;} \\ \Theta(n), & \textit{else.} \end{cases}$$

The goal of this lecture is to prove this theorem, starting from concrete algorithms whose regret matches the claimed upper bounds. The first case when there is only one Pareto-optimal action is trivial — the learner simply needs to stick with this unique Pareto-optimal action for every round to ensure 0 regret, since the benchmark in the regret definition is exactly the total loss of this Pareto-optimal action. The last case is also trivial from an upper bound perspective: any algorithm's regret is trivially $\mathcal{O}(n)$. In the rest of this section, we thus focus on the second and the third case.

1.1 Globally observable problems

First, we start with problems that are globally observable and aim at proposing an algorithm with regret $\mathcal{O}(n^{\frac{2}{3}})$. In light of our previous derivation of Exp3 for adversarial MAB, it is natural to consider applying Hedge again with some loss estimator. However, in partial monitoring, it is generally *impossible* to directly estimate the loss itself — to see this, just consider the following globally observable problem:

$$\ell = \begin{pmatrix} 0 & 0.5 & 0.5 & 1 \\ 0.5 & 0 & 1 & 0.5 \end{pmatrix} \qquad \text{and} \qquad \Phi = \begin{pmatrix} 1 & 2 & 1 & 2 \\ 2 & 1 & 2 & 1 \end{pmatrix};$$

for example, if the learner observes 1 after playing the first action, then she knows the outcome must be 1 or 3, but can never figure out which case it is (while the losses are drastically different in these two cases).

However, note the following simple observation: for any reference action b, we have

$$\mathrm{Reg}_n = \sum_{t=1}^{n}(\ell(a_t, z_t) - \ell(b, z_t)) - \sum_{t=1}^{n}(\ell(a^\star, z_t) - \ell(b, z_t)).$$

Therefore, it is in fact sufficient to estimate only the loss difference instead of the loss itself, which is now clearly possible in the previous example. More generally, recall that for a globally observable partial monitoring problem, for every pair of Pareto-optimal actions a and b, there exists a function $v_{ab} : [K] \times \Sigma \to \mathbb{R}$ such that

$$\ell(a, z) - \ell(b, z) = \sum_{k\in[K]} v_{ab}(k, \Phi(k, z)), \quad \forall z \in [d], \tag{1}$$

which suggests a natural importance-weighted estimator for $\ell(a, z) - \ell(b, z)$. Specifically, suppose that we sample action a_t from a distribution $p_t \in \Delta(K)$. Upon seeing observation $\Phi(a_t, z_t)$, we can construct the loss (difference) estimator $\widehat{\ell}_t = \frac{v(a_t,\Phi(a_t,z_t))}{p_t(a_t)} \in \mathbb{R}^K$, where we fix an arbitrary reference Pareto-optimal action b and let $v(\cdot, \cdot) \in \mathbb{R}^K$ be a vector such that its a-th coordinate, denoted as $v_a(\cdot, \cdot)$ (with a slight abuse of notation), is $v_{ab}(\cdot, \cdot)$ if a is Pareto-optimal, and arbitrary otherwise. The reason that we do not care about estimating the loss (difference) for non Pareto-optimal actions is because they can never be the optimal, which also suggests running the Hedge algorithm over only the set of Pareto-optimal actions, denoted by $\mathcal{A}$, to obtain $q_t \in \Delta(K)$ with $q_t(a) \propto \mathbf{1}\{a \in \mathcal{A}\} \exp\left(-\eta \sum_{\tau<t} \widehat{\ell}_\tau(a)\right)$.

However, recall that only playing Pareto-optimal actions is generally a bad idea, since playing non Pareto-optimal actions might be the only way to obtain useful information (recall the example of binary classification with query cost $c > 1/2$). This means that p_t cannot be simply q_t. In fact, for the estimator to be unbiased for any $a \in \mathcal{A}$:

$$\mathbb{E}_t\left[\widehat{\ell}_t(a)\right] = \sum_{k=1}^{K} p_t(k)\frac{v_a(k, \Phi(k, z_t))}{p_t(k)} = \ell(a, z_t) - \ell(b, z_t), \tag{2}$$

we generally require p_t to have a full support. On the other hand, we certainly also do not want to deviate from q_t significantly and play non optimal actions too often. This motivate us to choose p_t to be a mix of q_t and some small amount of uniform exploration: $p_t = (1 - \gamma)q_t + \gamma\frac{\mathbf{1}}{K}$ (where $\mathbf{1}$ is the all-one vector), so that every action is selected with probability at least γ/K. The complete algorithm is now shown below.

Algorithm 1: An algorithm for globally observable problems

Let $b \in \mathcal{A}$ be an arbitrary Pareto-optimal action and $v(\cdot, \cdot) \in \mathbb{R}^K$ be the vector with $v_{ab}(\cdot, \cdot)$ being its a-th coordinate if a is Pareto-optimal and arbitrary otherwise.
Let $\gamma \le 1$ be the exploration parameter and $\eta > 0$ be the learning rate.
For $t = 1, \ldots, n$:

1. Compute $q_t \in \Delta(K)$ such that $q_t(a) \propto \mathbf{1}\{a \in \mathcal{A}\} \exp\left(-\eta \sum_{\tau=1}^{t-1} \widehat{\ell}_\tau(a)\right)$.
2. Sample $a_t \sim p_t = (1 - \gamma)q_t + \gamma\frac{\mathbf{1}}{K}$ and receive feedback $\Phi(a_t, z_t)$.
3. Construct loss difference estimator $\widehat{\ell}_t = \frac{v(a_t,\Phi(a_t,z_t))}{p_t(a_t)}$.

To analyze this algorithm, we first make use of the key Hedge lemma from Lecture 6 to derive the following general regret bound that in fact holds for any p_t:

Lemma 1. *As long as $\eta\widehat{\ell}_t(a) \ge -1$ for all t and a, Algorithm 1 ensures:*

$$\mathbb{E}\left[\mathrm{Reg}_n\right] \le \frac{\ln K}{\eta} + \mathbb{E}\left[\sum_{t=1}^{n}\left(\textsc{Dev}_t + \textsc{Var}_t\right)\right],$$

where $\textsc{Dev}_t = (p_t - q_t)^\top \ell e_{z_t}$ measures the derivation of p_t from q_t, and $\textsc{Var}_t = \eta \sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q_t(a)}{p_t(k)} v_a^2(k, \Phi(k, z_t))$ measures the variance of the loss estimator.

Proof. Using the Hedge lemma and the fact $a^\star \in \mathcal{A}$, we have (as long as $\eta\widehat{\ell}_t(a) \geq -1$)

$$\sum_{t=1}^{n} \left\langle q_t, \widehat{\ell}_t \right\rangle - \sum_{t=1}^{n} \widehat{\ell}_t(a^\star) \leq \frac{\ln|\mathcal{A}|}{\eta} + \eta \sum_{t=1}^{n} \sum_{a\in\mathcal{A}} q_t(a)\widehat{\ell}_t^2(a). \tag{3}$$

We have already shown the unbiasedness of $\widehat{\ell}_t$ in Equation (2), and similarly its variance is bounded as

$$\mathbb{E}_t\left[\widehat{\ell}_t^2(a)\right] = \sum_{k=1}^{K} p_t(k) \frac{v_a^2(k,\Phi(k,z_t))}{p_t^2(k)} = \sum_{k=1}^{K} \frac{v_a^2(k,\Phi(k,z_t))}{p_t(k)}.$$

Therefore, taking expectation on both sides of Equation (3) and noting that $\mathbb{E}\left[\mathrm{Reg}_n\right] = \mathbb{E}\left[\sum_{t=1}^{n} p_t^\top \ell e_{z_t} - \sum_{t=1}^{n} \ell(a^\star, z_t)\right]$ finishes the proof. □

Next, we plug in the definition of p_t to show the final regret bound.

Theorem 2. *For any globally observable problems, Algorithm 1 ensures*

$$\mathbb{E}\left[\mathrm{Reg}_n\right] \leq \frac{\ln K}{\eta} + \gamma n + \frac{\eta n K^2 V^2}{\gamma}$$

as long as $\eta \leq \frac{\gamma}{KV}$ *where* $V = \max_{k\in[K],\sigma\in\Sigma} \|v(k,\sigma)\|_\infty$. *Picking* $\gamma = \sqrt{\eta}KV$ *and* $\eta = \min\left\{1, \left(\frac{\ln K}{nKV}\right)^{\frac{2}{3}}\right\}$ *gives* $\mathbb{E}\left[\mathrm{Reg}_n\right] = \mathcal{O}\left((nKV)^{\frac{2}{3}}(\ln K)^{\frac{1}{3}}\right)$.

Proof. First note that $\widehat{\ell}_t(a) \geq -KV/\gamma$ and thus the condition $\eta\widehat{\ell}_t(a) \geq -1$ of Lemma 1 holds when $\eta \leq \frac{\gamma}{KV}$. It thus suffices to bound $\textsc{Dev}_t$ and $\textsc{Var}_t$: $\textsc{Dev}_t = (p_t - q_t)^\top \ell e_{z_t} \leq \frac{\gamma}{K}\mathbf{1}^\top \ell e_{z_t} \leq \gamma$,

$$\textsc{Var}_t = \eta \sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q_t(a)}{p_t(k)} v_a^2(k,\Phi(k,z_t)) \leq \eta K V^2 \sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q_t(a)}{\gamma} = \frac{\eta K^2 V^2}{\gamma}.$$

Combining the bounds proves the theorem. □

We have thus achieved our goal of showing $\mathcal{O}(n^{2/3})$ regret for any globally observable problems. We conclude by pointing out that V is a problem-dependent constant and could in fact be exponentially large in d, but for all examples discussed in the last lecture, V is simply $\mathcal{O}(1)$.

1.2 Locally observable problems

Our next goal is to achieve $\mathcal{O}(\sqrt{n})$ regret for any locally observable problems. Recall that in such problems, for every pair of neighboring actions a and b, we can find a function v_{ab} such that Equation (1) holds and additionally $v_{ab}(k,\cdot)$ is zero for all $k \notin \{a,b\}$. The hope is that using this property, the variance term $\textsc{Var}_t$ in Lemma 1 can be improved from $\mathcal{O}(\frac{\eta}{\gamma})$ to just $\mathcal{O}(\eta)$ (which is then enough to achieve $\mathcal{O}(\sqrt{n})$ regret). However, it is unclear how to use this property directly since the idea of Algorithm 1 requires using v_{ab} even when a and b are *not* neighbor. To get an idea of how to modify Algorithm 1 to solve this problem, we will consider two examples below to convey three important messages.

Message One: a better v is needed. The first example is a bandit problem with $\Phi = \ell$, which, as discussed in the last lecture, is locally observable, since for any $a, b \in \mathcal{A}$ (that are not even necessarily neighbors), we can set $v_{ab}(a,\sigma) = \sigma$, $v_{ab}(b,\sigma) = -\sigma$, and $v_{ab}(k,\sigma) = 0$ for all $k \notin \{a,b\}$ so that Equation (1) holds. If we use such v_{ab} in Algorithm 1, notice that the variance term is still not improved since

$$\textsc{Var}_t = \eta \sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q_t(a)}{p_t(k)} v_a^2(k,\Phi(k,z_t)) = \eta \sum_{a\in\mathcal{A}} \sum_{k\in\{a,b\}} \frac{q_t(a)}{p_t(k)} v_a^2(k,\Phi(k,z_t))$$

can still be of order η/γ due to the term $q_t(a)/p_t(b)$. However, we know that the problem can be solved by Exp3, which basically also fits into the framework of Algorithm 1 with $v(k,\sigma) = \sigma \cdot e_k$.

The estimator is unbiased in estimating the loss itself (instead of loss difference): $\mathbb{E}_t[\widehat{\ell}_t(a)] = \ell(a, z_t)$, and the variance is now drastically improved:

$$\textsc{Var}_t = \eta \sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q_t(a)}{p_t(k)} v_a^2(k, \Phi(k, z_t)) = \eta \sum_{a\in\mathcal{A}} \frac{q_t(a)}{p_t(a)} \Phi^2(a, z_t) \le \frac{K\eta}{1-\gamma} = \mathcal{O}(K\eta).$$

This example shows that it is not enough to only consider $v(\cdot,\cdot)$ that estimates loss difference between two actions. Instead, let us generalize the idea and look for v that estimates the loss of an action compared to any constant, that is, a v from the following set

$$\mathcal{H} = \left\{ v : [K] \times \Sigma \to \mathbb{R}^K \;\middle|\; \forall z \in [d], \exists \xi_z \in \mathbb{R}, \text{ s.t. } \ell(a,z) - \xi_z = \sum_{k=1}^{K} v_a(k, \Phi(k,z)), \forall a \in \mathcal{A} \right\}. \tag{4}$$

In Algorithm 1, we have used any $v \in \mathcal{H}$ such that $\xi_z = \ell(b, z)$ for some reference action b, while in bandit, we used a $v \in \mathcal{H}$ such that $\xi_z = 0$. Note that if we equivalently see v as a vector in space $\mathbb{R}^{K\times|\Sigma|\times K}$, then the set $\mathcal{H}$ is in fact a convex set defined by $\mathcal{O}(dK)$ linear constraints (more specifically, for each z, we have $|\mathcal{A}| - 1$ constraints specified by $\ell(a,z) - \sum_{k=1}^{K} v_a(k, \Phi(k,z)) = \ell(b,z) - \sum_{k=1}^{K} v_b(k, \Phi(k,z)) = \ell(c,z) - \sum_{k=1}^{K} v_c(k, \Phi(k,z)) = \cdots$, for $a, b, c, \ldots \in \mathcal{A}$.)

Message Two: a better p_t is needed. Next, consider the example of binary classification with query cost $c < 1/2$ (which is locally observable as discussed last time):

$$\ell = \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ c & c \end{pmatrix} \qquad \text{and} \qquad \Phi = \begin{pmatrix} \perp & \perp \\ \perp & \perp \\ \text{☺} & \text{☹} \end{pmatrix}. \tag{5}$$

An obvious estimation function $v \in \mathcal{H}$ is: $v(k, \sigma) = \mathbf{1}\{k = 3\}(0, 1, c)$ if $\sigma =$ ☺ and $v(k, \sigma) = \mathbf{1}\{k = 3\}(1, 0, c)$ if $\sigma =$ ☹ (that is, output 0 when playing the first two actions that give no information, and the true loss when playing the third action that reveals all information). The variance term would then be

$$\textsc{Var}_t = \eta \sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q_t(a)}{p_t(k)} v_a^2(k, \Phi(k, z_t)) = \eta \sum_{a\in\mathcal{A}} \frac{q_t(a)}{p_t(3)} v_a^2(3, \Phi(3, z_t)) \le \frac{\eta}{p_t(3)}, \tag{6}$$

which intuitively makes sense since action 3 is the only action that reveals information and thus the less likely we pick this action, the higher the variance. Therefore, if we still let p_t be simply a mix of q_t and a small amount of uniform exploration, the variance would be of order η/γ again. However, now consider a different p_t that is obtained from q_t by drastically increasing $q_t(3)$: move $\min\{q_t(1), q_t(2)\}$ weight from actions 1 and 2 to 3. For example, in the case where $q_t(2) \le q_t(1)$, p_t is defined as: $p_t(1) = q_t(1) - q_t(2), p_t(2) = 0$, and $p_t(3) = q_t(3) + 2q_t(2)$; see Figure 1 for an illustration. In this case, the variance term is improved, and more importantly, the deviation term $\textsc{Dev}_t = (p_t - q_t)^\top \ell e_{z_t}$ is actually always negative since

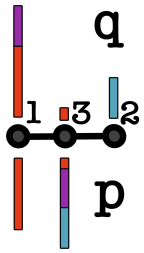

Figure 1

$$(p_t - q_t)^\top \ell = (-q_t(2), -q_t(2), 2q_t(2))\ell = ((2c-1)q_t(2), (2c-1)q_t(2))$$

and $c < 1/2$, meaning that playing p_t is guaranteed to suffer less loss than q_t! This example shows that by coming up with a more sophisticated p_t (instead of simply mixing q_t with uniform exploration), it is possible to reduce both the deviation $\textsc{Dev}_t$ and the variance $\textsc{Var}_t$ simultaneously.

Message Three: a better combination of v and p_t is needed. Even with this new p_t, however, the variance term $\eta/p_t(3)$ from Equation (6) can still be large if $p_t(3) = q_t(3) + 2q_t(2)$ is too small. To address this, consider a slightly different estimation function $v \in \mathcal{H}$: $v(k, \sigma) = \mathbf{1}\{k = 3\}(0, 1, c)$ if $\sigma =$ ☺ and $v(k, \sigma) = \mathbf{1}\{k = 3\}(0, -1, c-1)$ if $\sigma =$ ☹, which is only different from the previous choice by shifting the second case by -1 (and thus still in $\mathcal{H}$ by definition). The variance term then becomes (using $v_1^2(3, \Phi(3, z_t)) = 0$ always)

$$\textsc{Var}_t = \eta \sum_{a\in\mathcal{A}} \frac{q_t(a)}{p_t(3)} v_a^2(3, \Phi(3, z_t)) \le \frac{\eta(q_t(2) + q_t(3))}{p_t(3)} = \frac{\eta(q_t(2) + q_t(3))}{2q_t(2) + q_t(3)} \le \eta,$$

which is much better than η/γ and leads to $\mathcal{O}(\sqrt{n})$ regret in the end. Note that we have been assuming $q_t(2) \le q_t(1)$, and in the case $q_t(2) > q_t(1)$, we will have to change v accordingly (that is, shifting the first case by -1 now) to ensure $\textsc{Var}_t \le \eta$. This conveys the important message that we need to choose a good combination of v and p_t *depending on the value of* q_t.

Exploration by optimization. With these examples in mind, it remains to figure out how to pick a good combination of $v \in \mathcal{H}$ and p_t based on q_t. In fact, let us take this idea to the extreme and ask: what is the best possible combination of v and p_t? The quality of a combination can be simply measured by $\textsc{Dev}_t + \textsc{Var}_t$, and thus the best combination is the solution to the following min-max optimization problem (the $\max_{z_t}$ part is due to the fact that z_t is unknown before deciding p_t):

$$\begin{aligned}\textsc{Opt}_\eta(q_t) &= \min_{(v,p_t)\in\Omega} \max_{z_t\in[d]} (\textsc{Dev}_t + \textsc{Var}_t) \\ &= \min_{(v,p_t)\in\Omega} \max_{z_t\in[d]} \left((p_t - q_t)^\top \ell e_{z_t} + \eta \sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q_t(a)}{p_t(k)} v_a^2(k, \Phi(k, z_t)) \right),\end{aligned} \tag{7}$$

where the decision space Ω is

$$\Omega = \left\{ (v,p) \in \mathcal{H} \times \Delta(K) \;\middle|\; \eta v_a(k,\sigma) + p(k) \geq 0, \forall a \in \mathcal{A}, k \in [K], \sigma \in \Sigma \right\}, \tag{8}$$

which is convex and ensures that the condition $\eta\widehat{\ell}_t(a) \geq -1$ (from Lemma 1) always holds. While seemingly complex, $\textsc{Opt}_\eta(q_t)$ is in fact a convex optimization problem and can be solved using any off-the-shelf convex solvers. This idea, called *exploration by optimization*, is first introduced by Lattimore and Szepesvári [2020] (and turns out to be powerful for other problems as well). The complete algorithm is now shown below.

Algorithm 2: Exploration by optimization for locally observable problems

Let $\eta > 0$ be the learning rate.
For $t = 1, \ldots, n$:

1. Compute $q_t \in \Delta(K)$ such that $q_t(a) \propto \mathbf{1}\{a \in \mathcal{A}\} \exp\left(-\eta \sum_{\tau=1}^{t-1} \widehat{\ell}_\tau(a)\right)$.
2. Solve $\textsc{Opt}_\eta(q_t)$ from Equation (7) to obtain v and p_t.
3. Sample $a_t \sim p_t$ and receive feedback $\Phi(a_t, z_t)$.
4. Construct loss difference estimator $\widehat{\ell}_t = \frac{v(a_t, \Phi(a_t, z_t))}{p_t(a_t)}$.

It turns out that we have $\textsc{Opt}_\eta(q) = \mathcal{O}(\eta)$ for all locally observable problems:

Lemma 2. *For any locally observable problem and $q \in \Delta(\mathcal{A})$, we have $\textsc{Opt}_\eta(q) = \mathcal{O}(\eta K^3 |\Sigma|^2)$.*[1]

This immediately implies the following regret bound using Lemma 1.

Theorem 3. *For any locally observable problem, Algorithm 2 ensures $\mathbb{E}[\mathrm{Reg}_n] \leq \frac{\ln K}{\eta} + \mathcal{O}(\eta n K^3 |\Sigma|^2)$, which is $\mathcal{O}(\sqrt{n K^3 |\Sigma|^2 \ln K})$ after picking the optimal η.*

Note that the bound has no dependence at all on the number of outcomes d. It also has no problem-dependent constant such as V in Theorem 2 (that could be exponentially large). The proof of Lemma 2 is somewhat involved, and we only discuss the key ideas below.

Proof sketch of Lemma 2. We drop all the subscripts t in this proof for conciseness. The first step to bound $\textsc{Opt}_\eta(q)$ is to linearize the $\max_z$ part in order to apply the minimax theorem (similarly to what we did in Lecture 5 to analyze $\mathcal{V}^{\text{seq}}(\mathcal{F}, n)$):

$$\begin{aligned}\textsc{Opt}_\eta(q) &= \min_{(v,p)\in\Omega} \max_{z\in[d]} \left((p-q)^\top \ell e_z + \eta \sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q(a)}{p(k)} v_a^2(k, \Phi(k,z)) \right) \\ &= \min_{(v,p)\in\Omega} \max_{u\in\Delta(d)} \left((p-q)^\top \ell u + \eta \mathbb{E}_{i\sim u} \left[\sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q(a)}{p(k)} v_a^2(k, \Phi(k,i)) \right] \right) \\ &= \max_{u\in\Delta(d)} \min_{(v,p)\in\Omega} \Bigg(\underbrace{(p-q)^\top \ell u}_{\textsc{Dev}} + \underbrace{\eta \mathbb{E}_{i\sim u} \left[\sum_{a\in\mathcal{A}} \sum_{k=1}^{K} \frac{q(a)}{p(k)} v_a^2(k, \Phi(k,i)) \right]}_{\textsc{Var}} \Bigg).\end{aligned}$$

[1] More precisely, this bound holds for any locally observable problems that have no degenerate actions (otherwise, $\textsc{Opt}_\eta(q)$ is still of order $\mathcal{O}(\eta)$ but with other problem-dependence constants).

The advantage of swapping the role of the learner and the environment via the minimax theorem is that it is now intuitively much clearer how to select a good combination of v and p. Indeed, given that the environment is going to sample the outcome according to $u \in \Delta(d)$, we should naturally assign more weight to the optimal action under u, that is, $a_u \in \operatorname{argmin}_a \langle \ell_a, u \rangle$. However, we should not move too much weight from other actions to a_u either as it would likely increase the variance. To achieve a good balance, we will define a tree with a_u as the root and other Pareto-optimal actions in $\mathcal{A}$ as the rest of the nodes, and let the weight "flow" from the leaves to the root in a particular way.

More specifically, for each $a_u \neq a \in \mathcal{A}$, define its parent as $\mathrm{par}(a) = \operatorname{argmin}_{k \in N_a} \langle \ell_k, u \rangle$ where N_a contains a and all its neighbors. It can be verified that this leads to a well-defined tree (by using the concavity of the function $u \to \min_{a \in \mathcal{A}} \langle \ell_a, u \rangle$; try it yourself for the case with $d = 2$), where for each node $a_u \neq a \in \mathcal{A}$, its parent $\mathrm{par}(a)$ must suffer less loss under u, that is, $\langle \ell_a, u \rangle > \langle \ell_{\mathrm{par}(a)}, u \rangle$. This means that if we transfer weight only from a node to its parent, the resulting new distribution will only suffer less loss. In particular, starting from the given distribution $q \in \Delta(\mathcal{A})$, we will transfer the weight in the following manner: for each note $a \in \mathcal{A}$, suppose that it has m ancestors; divide its weight $q(a)$ equally into $m+1$ shares and move one share to each of its ancestor (keeping one share for itself). This is best illustrated in the following figure, where the left picture represents the original q supported on five actions, with the arrows below them pointing to their parents, and the right picture represents the new distribution.

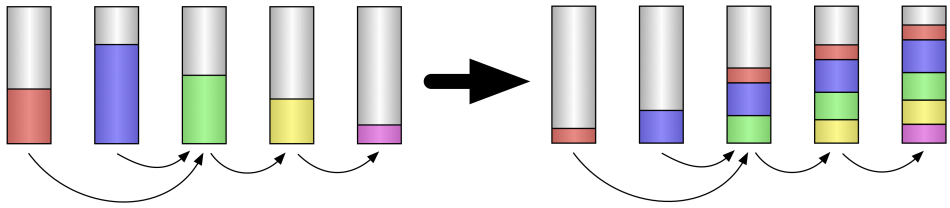

Figure 2 : An illustration of how to obtain r from q

Denote the new distribution as $r \in \Delta(\mathcal{A})$ after this process. The final distribution p is defined as $p = (1-\gamma)r + \gamma \frac{\mathbf{1}}{K}$ for some $\gamma \in (0, 1/2]$, which again mixes r with a small amount of exploration. This step is to ensure that the constraint in the definition of Ω (Equation (8)) is satisfied (see details below). Since r is strictly better than q: $(r-q)^\top \ell u < 0$, the final distribution is only γ worse: $\textsc{Dev} = (p-q)^\top \ell u = (1-\gamma)(r-q)^\top \ell u + \gamma(\frac{\mathbf{1}}{K} - q)^\top \ell u \leq \frac{\gamma}{K} \mathbf{1}^\top \ell u \leq \gamma$.

It remains to pick v and show that $\textsc{Var}$ is also well controlled. Similarly to Algorithm 1, we pick v such that v_a is v_{ab} defined in Equation (1) where the reference function b is now set to the root a_u. Additionally, we require this v_{ab} to be such that $v_{ab}(k, \sigma) = 0$ whenever k is not in $\mathrm{path}(a)$, the set of nodes on the path from a to the root a_u. This is possible by the local observability and the fact that every node and its parent are neighbors. In fact, it is also possible to make sure that the magnitude of each $v_{ab}(k, \sigma)$ is at most $2|\Sigma|$ (details omitted). With such an estimation function v, we calculate the variance as

$$\sum_{a \in \mathcal{A}} \sum_{k=1}^{K} \frac{q(a)}{p(k)} v_a^2(k, \Phi(k,i)) \leq 4|\Sigma|^2 \sum_{a \in \mathcal{A}} \sum_{k \in \mathrm{path}(a)} \frac{q(a)}{p(k)} \leq 8|\Sigma|^2 \sum_{a \in \mathcal{A}} \sum_{k \in \mathrm{path}(a)} \frac{q(a)}{r(k)},$$

where the last step is due to $p(k) \geq (1-\gamma) r(k) \geq \frac{1}{2} r(k)$. To continue, we use two important facts based on the construction of r (*c.f.* Figure 2 again): 1) for any node $k \in \mathrm{path}(a)$, $r(k) \geq r(a)$ (since every node gives an equal share to its ancestor); 2) for any node $a \in \mathcal{A}$, $r(a) \geq q(a)/K$ (since every node at least keeps one share of its original weight). Combining these, we continue with

$$\sum_{a \in \mathcal{A}} \sum_{k=1}^{K} \frac{q(a)}{p(k)} v_a^2(k, \Phi(k,i)) \leq 8|\Sigma|^2 \sum_{a \in \mathcal{A}} \sum_{k \in \mathrm{path}(a)} \frac{q(a)}{r(a)} \leq 8|\Sigma|^2 \sum_{a \in \mathcal{A}} \sum_{k \in \mathrm{path}(a)} K \leq 4K^3 |\Sigma|^2,$$

which means $\textsc{Var} \leq 8\eta K^3 |\Sigma|^2$.

Combining the bounds for both $\textsc{Dev}$ and $\textsc{Var}$, we conclude $\textsc{Opt}_\eta(q) \leq \gamma + 8\eta K^3 |\Sigma|^2$. It remains to pick γ, which, as mentioned earlier, is used to ensure the condition $\eta v_a(k, \sigma) + p(k) \geq 0$ in

Equation (8). Since $|v_a(k,\sigma)| \le 2|\Sigma|$ and $p(k) \ge \gamma/K$, it suffices to pick $\gamma = 2\eta K|\Sigma|$. This shows $\textsc{Opt}_\eta(q) \le 2\eta K|\Sigma| + 8\eta K^3|\Sigma|^2 = \mathcal{O}(\eta K^3|\Sigma|^2)$ and completes the entire proof. □

We point out that the way we construct v and p in the proof above does not correspond to an actual algorithm, since this is based on a known $u \in \Delta(d)$ (enabled by the minimax theorem). Instead, to find v and p_t in the algorithm, as mentioned, we need to solve the convex program Equation (7).

2 Partial Monitoring: Lower Bounds

Now that all upper bounds in the classification theorem have been proven, we move on to prove the lower bounds. The first case is again trivial since the minimax regret of any problem is at least 0 (the adversary just needs to stick with one particular outcome all the time). For the other three cases, it is not difficult to find *one* instance where the corresponding lower bound holds. Indeed,

- for locally observable problems, just note that MAB is one such instance, where we proved $\Omega(\sqrt{nK})$ regret in the last lecture;
- for problems that are not even globally observable, we have discussed the hopeless game last time where one needs to do binary classification under absolutely no feedback at all, so $\Omega(n)$ regret is obvious;
- finally, for globally observable problems that are not locally observable, we can consider the example of classification with query cost again, that is, Equation (5), this time with $c > 1/2$. A simple informal argument below indicates the $\Omega(n^{2/3})$ lower bound: if the environment selects one of the outcomes with probability $0.5+\epsilon$ each time (and the other one with probability $0.5-\epsilon$), then the learner either needs to query $1/\epsilon^2$ times to figure out which outcome appears more often, in which case she suffers $(c-(0.5-\epsilon))/\epsilon^2 = \Omega(1/\epsilon^2)$ regret, or she does not query enough and never figures out which one is better, in which case she suffers $\Omega(n\epsilon)$ regret. By setting $\epsilon = n^{-\frac{1}{3}}$, we thus know that in either case the regret is at least $\Omega(n^{\frac{2}{3}})$.

However, the classification theorem says something more — it says that for *every* (not just one particular) problem in each category, the corresponding regret lower bound holds. To formally prove this stronger statement, we utilize similar ideas from the MAB lower bound proof.

Proof for lower bounds of Theorem 1. The idea is still to first construct a stochastic environment where two actions a and b are equally good, identify the one that is selected less often by the algorithm, and then construct another stochastic environment where this action becomes slightly better, but it is hard for the algorithm to realize the change. Again, it is sufficient to consider deterministic algorithms. Below, we first describe the common part of the proof for all three categories.

Let a and b be a pair of neighboring actions (we will specify which pair later for each category). Consider an environment where the outcomes $z_1, \dots, z_n$ are i.i.d. samples from a distribution $u \in \Delta(d)$ that lies in the (relative) interior of $C_a \cap C_b$ (so by the definition of cells, a and b are both optimal actions in this environment). For any fixed algorithm, let $m_k = \mathbb{E}\left[\sum_{t=1}^n \mathbf{1}\{a_t = k\}\right]$ be the expected total number of times action k is selected in this environment. Without loss of generality, assume $m_b \le n/2$ (note that one of m_a and m_b must be no more than $n/2$).

Next, consider a different environment where the outcomes $z_1, \dots, z_n$ are i.i.d. samples from a distribution $u' = u + \delta$ where δ satisfies $\sum_{i=1}^d \delta(i) = 0$ and $\langle \ell_a - \ell_b, \delta\rangle = \epsilon$ for some small enough $\epsilon > 0$ such that $u' \in C_b$ (that is, action b is optimal under this new environment). Note that this is always possible since the constraint $\sum_i \delta(i) = 0$ defines a space that is orthogonal to the all-one vector, and $\ell_a - \ell_b$ cannot be in the same direction as the all-one vector for otherwise one of them strictly dominates the other.

It remains to argue that the regret of the same algorithm under this environment has to be large. Note that every time the algorithm selects action a, it incurs regret $\langle \ell_a - \ell_b, u'\rangle = \langle \ell_a - \ell_b, \delta\rangle = \epsilon$; and every time it selects an action $k \notin N_{ab} = \{a, b\}$, it incurs some constant regret which can be assumed to be larger than $c + \epsilon$ for some constant c as long as ϵ is small enough. Therefore, the

regret is

$$\mathbb{E}'\left[\mathrm{Reg}_n\right] \geq \mathbb{E}\left[\sum_{t=1}^{n} \ell(a_t, z_t) - \sum_{t=1}^{n} \ell(b, z_t)\right] \geq (n - m'_b)\epsilon + c\bar{m}'$$

where $\mathbb{E}'$ denotes the expectation in environment u', $m'_k = \mathbb{E}'\left[\sum_{t=1}^{n} \mathbf{1}\left\{a_t = k\right\}\right]$, and $\bar{m}' = \sum_{k \notin N_{ab}} m'_k$.[2] Now we relate m_b and m'_b in a similar way as in the MAB lower bound proof:

$$m'_b \leq m_b + n\left\|\mathbb{P} - \mathbb{P}'\right\|_1 \leq m_b + n\sqrt{2\mathrm{KL}(\mathbb{P}' \parallel \mathbb{P})}$$

where $\mathbb{P}$ and $\mathbb{P}'$ are the distributions of the observation sequence $\Phi(a_1, z_1), \ldots, \Phi(a_n, z_n)$ in environment u and u' respectively. Generalizing the divergence decomposition lemma (Lemma 3 of Lecture 9), one can verify the following

$$\mathrm{KL}(\mathbb{P}' \parallel \mathbb{P}) = \sum_{k=1}^{K} m'(k)\mathrm{KL}(\mathbb{P}'_k \parallel \mathbb{P}_k)$$

where $\mathbb{P}_k$ and $\mathbb{P}'_k$ are the distributions of $\Phi(k, z)$ when z is drawn from u and u' respectively, which, with our notation of signal matrix S_k, can be conveniently written as $S_k u$ and $S_k u'$. We next discuss the three categories separately.

Locally observable problems. We bound $\mathrm{KL}(\mathbb{P}'_k \parallel \mathbb{P}_k)$ for any k as follows:

$$\mathrm{KL}(\mathbb{P}'_k \parallel \mathbb{P}_k) = \mathrm{KL}(S_k u' \parallel S_k u) \leq \mathrm{KL}(u' \parallel u) \leq \sum_{i=1}^{d} \frac{(u(i)' - u(i))^2}{u(i)} = \sum_{i=1}^{d} \frac{\delta(i)^2}{u(i)} = \mathcal{O}(\epsilon^2),$$

where the first inequality is by the data processing inequality and the second inequality is because KL divergence is bounded by Chi-square distance. Therefore, we have $m'_b \leq m_b + \mathcal{O}\left(\epsilon n\sqrt{n}\right) \leq \frac{n}{2} + \mathcal{O}\left(\epsilon n\sqrt{n}\right)$, and thus

$$\mathbb{E}'\left[\mathrm{Reg}_n\right] \geq (n - m'_b)\epsilon \geq \left(\frac{n}{2} - \mathcal{O}\left(\epsilon n\sqrt{n}\right)\right)\epsilon,$$

which is $\Omega(\sqrt{n})$ by setting $\epsilon = \beta/\sqrt{n}$ for some small enough constant β.

Non globally observable problems. In this case, we let a and b be a neighboring pair that is not globally observable, and also let δ be orthogonal to rowspace$(S_{[K]})$. Note that this is always possible: the condition $\langle \ell_a - \ell_b, \delta \rangle = \epsilon$ can be satisfied since by definition $\ell_a - \ell_b \notin$ rowspace$(S_{[K]})$, and the condition $\sum_i \delta(i) = 0$ holds automatically since the all-one vector is in rowspace$(S_{[K]})$. In this construction, for every k we have

$$\mathrm{KL}(\mathbb{P}'_k \parallel \mathbb{P}_k) = \mathrm{KL}(S_k u' \parallel S_k u) = \mathrm{KL}(S_k u + S_k \delta \parallel S_k u) = \mathrm{KL}(S_k u \parallel S_k u) = 0,$$

meaning that the observation distributions are exactly the same in the two environments, and thus $m'_b \leq m_b \leq n/2$. Therefore, $\mathbb{E}'\left[\mathrm{Reg}_n\right] \geq (n - m'_b)\epsilon \geq \frac{n\epsilon}{2} = \Omega(n)$.

Globally (but not locally) observable problems. In this case, we let a and b be a neighboring pair that is not locally observable, and let δ be orthogonal to rowspace$(S_{N_{ab}})$, which is also always possible by the fact $\ell_a - \ell_b \notin$ rowspace$(S_{N_{ab}})$ and that the all-one vector is in $S_{N_{ab}}$. Therefore, for $k \in N_{ab}$, again we have $\mathrm{KL}(\mathbb{P}'_k \parallel \mathbb{P}_k) = \mathrm{KL}(S_k u + S_k \delta \parallel S_k u) = \mathrm{KL}(S_k u \parallel S_k u) = 0$ (so playing a or b is not able to distinguish between the two environments), and for $k \notin N_{ab}$, we use the previous bound $\mathrm{KL}(\mathbb{P}'_k \parallel \mathbb{P}_k) = \mathcal{O}\left(\epsilon^2\right)$. Combing everything we have $m'_b \leq n/2 + \mathcal{O}\left(n\epsilon\sqrt{2\bar{m}'}\right)$ and

$$\mathbb{E}'\left[\mathrm{Reg}_n\right] \geq (n - m'_b)\epsilon + c\bar{m}' \geq \frac{n\epsilon}{2} - \mathcal{O}\left(n\epsilon^2\sqrt{2\bar{m}'}\right) + c\bar{m}' \geq \frac{n\epsilon}{2} - \mathcal{O}\left(n^2\epsilon^4\right),$$

where the last step is by lower bounding the quadratic (in terms of $\sqrt{\bar{m}'}$) by its minimum. Finally setting $\epsilon = \beta n^{-\frac{1}{3}}$ for some small enough constant β proves $\mathbb{E}'\left[\mathrm{Reg}_n\right] = \Omega(n^{\frac{2}{3}})$. □

References

Tor Lattimore and Csaba Szepesvári. Exploration by optimisation in partial monitoring. In *Conference on Learning Theory*, pages 2488–2515. PMLR, 2020.

[2] We have assumed that there are no degenerate/duplicate actions. However, it is not hard to verify that the same statement holds up to some constant in the general case.