DATA 37200: Learning, Decisions, and Limits
(Winter 2025)

Lower Bounds for Stochastic MAB

Instructor: Haifeng Xu

Outline

- Technical Preparations
- Detour: Best-Arm Identification (BAI) Lower Bounds
- MAB Regret Lower Bounds
    - Instance-Independent Lower Bound
    - Instance-Dependent Lower Bounds

Lower Bounds: What and Why?

We look to derive results of form like

$$\text{Regret} \geq C\sqrt{KT} \qquad \text{for some constant } C$$

or equivalently, $\text{Regret} = \Omega(\sqrt{KT})$

This helps to understand what we *cannot* achieve – i.e., our limits

Lower Bounds: What and Why?

We look to derive results of form like

$$\text{Regret} \geq C\sqrt{KT} \qquad \text{for some constant } C$$

or equivalently, $\text{Regret} = \Omega(\sqrt{KT})$

This helps to understand what we *cannot* achieve – i.e., our limits

- If you have learned computational hardness (e.g., NP-hardness), that shares a similar spirit but very different flavor
- Computational lower bounds are mostly *conditional*
  - E.g., 3SAT takes at least exponential time to solve, *if P ≠ NP*
  - P ≠ NP is an assumption
- The bound we will show for MAB is *unconditional*
  - i.e., they are facts that do not require any assumption
  - Proofs here will mostly uses information theory

KL-Divergence

A useful quantity that measures "distance" between two distributions

Definition (KL-Divergence). For any two distributions p, q supported on discrete set X, their Kullback–Leibler (KL) divergence is defined as

$$KL(p, q) = \sum_{x \in X} p(x) \ln \left[\frac{p(x)}{q(x)}\right] = \mathbb{E}_{x \sim p} \ln \left[\frac{p(x)}{q(x)}\right]$$

Remarks

- Similarly defined for continuous domain, though not needed for now
- It is not symmetric: $KL(p, q) \neq KL(q, p)$
- Closely related to entropy; also named "relative entropy"
- Widely used in practice for measuring distribution distance (e.g., it is the default regularizer for fine-tuning LLMs)

KL-Divergence: An Example

Definition. (Biased Random Coins). For any $\epsilon \in [-\frac{1}{2}, \frac{1}{2}]$, let RC_ϵ be the binary random coin with $\epsilon/2$ bias -- i.e., it takes value 1/head with prob $\frac{1+\epsilon}{2}$, and 0/tail otherwise.

- RC_ϵ is a Bernoulli random variable with $p = (1+\epsilon)/2$
- Calculating KL-divergence

$$KL(RC_\epsilon, RC_0) = \frac{1+\epsilon}{2} \ln\left[\frac{(1+\epsilon)/2}{1/2}\right] + \frac{1-\epsilon}{2} \ln\left[\frac{(1-\epsilon)/2}{1/2}\right]$$

Claim: $KL(RC_\epsilon, RC_0) \le 2\epsilon^2$ and $KL(RC_0, RC_\epsilon) \le \epsilon^2$ for any $\epsilon \in (0, ½)$

Remark: this ϵ^2 term turns out to be the reason of the $\Omega(T^{\frac{1}{2}})$ lower bound

Claim's proof deferred to HW.

Properties of KL-Divergence

Theorem: KL-divergence satisfies the following properties

a. **Gibb's Inequality**: $KL(p,q) \geq 0$, with equality if and only if $p = q$

b. **Chain rule for product distributions**: For $i = 1, \cdots, n$, let p_i, q_i be two distributions supported on X_i. $p = p_1 \times p_2 \cdots \times p_n, q = q_1 \times q_2 \cdots \times q_n$ be their product distributions. Then $KL(p,q) = \sum_{i=1}^{n} KL(p_i, q_i)$.

c. **Pinsker's inequality**: For any event $A \subseteq X$, we have

$$2[p(A) - q(A)]^2 \leq KL(p,q)$$

Remarks.

- The probability difference of *any event* is upper bounded by $O(\sqrt{KL(p,q)})$
- Illustrates why it captures "divergence" between two distributions
- Pinsker's inequality implies $KL(RC_0, RC_\epsilon) \geq \epsilon^2/2$ (compare to previous claim $KL(RC_0, RC_\epsilon) \leq \epsilon^2$)

Properties of KL-Divergence

Theorem: KL-divergence satisfies the following properties

a. **Gibb's Inequality**: $KL(p, q) \geq 0$, with equality if and only if $p = q$

b. **Chain rule for product distributions**: For $i = 1, \cdots, n$, let p_i, q_i be two distributions supported on X_i. $p = p_1 \times p_2 \cdots \times p_n, q = q_1 \times q_2 \cdots \times q_n$ be their product distributions. Then $KL(p, q) = \sum_{i=1}^{n} KL(p_i, q_i)$.

c. **Pinsker's inequality**: For any event $A \subseteq X$, we have

$$2[p(A) - q(A)]^2 \leq KL(p, q)$$

Proofs deferred to HW1.

Exercise with KL & a Warm-up Lower-Bound Problem

How many coin flips are needed to confidently tell it is fair or not?

A Fair Coin or Not?

- You know a coin is either RC_0 or RC_ϵ
  - RC_0 is called a fair coin, and RC_ϵ has $\epsilon/2$ bias
- You can flip the coin T times
- Based on your observations, you have a (deterministic) decision rule to decide it is fair or biased:

$$\text{Rule: } \{0,1\}^T \rightarrow \{\text{fair, biased}\}$$

A Fair Coin or Not?

- You know a coin is either RC_0 or RC_ϵ
  - RC_0 is called a fair coin, and RC_ϵ has $\epsilon/2$ bias
- You can flip the coin T times
- Based on your observations, you have a (deterministic) decision rule to decide it is fair or biased:

$$\text{Rule: } \{0,1\}^T \rightarrow \{\text{fair, biased}\}$$

Question: how large your T needs to at least be for you to be correct with high prob in the following sense?

$$\Pr[\text{Rule}(observations) = \text{fair} \mid RC_0] \geq 3/4 \qquad (1)$$
$$\Pr[\text{Rule}(observations) = \text{biased} \mid RC_\epsilon] \geq 3/4 \qquad (2)$$

A Fair Coin or Not?

Claim: Fix a decision rule that satisfies (1) and (2). Then $T \geq \frac{1}{2\epsilon^2}$.

Proof

- The decision rule is deterministic, so there is a subset $A_0 \subseteq \{0,1\}^T$ of events such that

  Rule(x) = fair   for any $x \in A_0$

  Rule(x) = biased   for any $x \notin A_0$

- Following accuracy requirement implies A_0 happens with probability $\geq 3/4$ under RC_0, but happens with prob $\leq 1/4$ under RC_ϵ

Question: how large your T needs to at least be for you to be correct with high prob in the following sense?

$$\Pr[\text{Rule}(observations) = \text{fair} \mid RC_0] \geq 3/4 \qquad (1)$$
$$\Pr[\text{Rule}(observations) = \text{biased} \mid RC_\epsilon] \geq 3/4 \qquad (2)$$

A Fair Coin or Not?

Claim: Fix a decision rule that satisfies (1) and (2). Then $T \geq \frac{1}{2\epsilon^2}$.

Proof

- The decision rule is deterministic, so there is a subset $A_0 \subseteq \{0,1\}^T$ of events such that

$$\text{Rule}(x) = \text{fair} \quad \text{for any } x \in A_0$$

$$\text{Rule}(x) = \text{biased} \quad \text{for any } x \notin A_0$$

- Following accuracy requirement implies A_0 happens with probability $\geq 3/4$ under RC_0, but happens with prob $\leq 1/4$ under RC_ϵ

That is, $\Pr(A_0|RC_0) - \Pr(A_0|RC_\epsilon) \geq 3/4 - 1/4 = 1/2$

Next we employ properties of KL to show T has to be large to achieve the above inequality

A Fair Coin or Not?

Claim: Fix a decision rule that satisfies (1) and (2). Then $T \geq \frac{1}{2\epsilon^2}$.

Proof (con'd)

That is, $\Pr(A_0|RC_0) - \Pr(A_0|RC_\epsilon) \geq 3/4 - 1/4 = 1/2$

➢Let $p_i = RC_0$, $q_i = RC_\epsilon$; consider product distributions $p = \Pi_{i=1}^{T} p_i$, $q = \Pi_{i=1}^{T} q_i$

➢p, q are measures over $\{0,1\}^T \supseteq A_0$, so Pinsker's inequality told us

$$KL(p,q) \geq 2|\Pr(A_0|RC_0) - \Pr(A_0|RC_\epsilon)|^2 \ \geq 1/2$$

➢Employ chain rule to upper bound KL:

$$KL(q,q) = \ \sum_{i=1}^{T} KL(p_i, q_i) \leq T\epsilon^2$$

➢Combing these two inequalities we have

$$T \geq \frac{KL(p,q)}{\epsilon^2} \geq \frac{1}{2\epsilon^2}$$

A Fair Coin or Not?

Claim: Fix a decision rule that satisfies (1) and (2). Then $T \geq \frac{1}{2\epsilon^2}$.

Proof (con'd)

That is, $\Pr(A_0|RC_0) - \Pr(A_0|RC_\epsilon) \geq 3/4 - 1/4 = 1/2$

- Let $p_i = RC_0$, $q_i = RC_\epsilon$; consider product distributions $p = \Pi_{i=1}^{T} p_i$, $q = \Pi_{i=1}^{T} q_i$
- p, q are measures over $\{0,1\}^T \supseteq A_0$, so Pinsker's inequality told us

$$KL(p, q) \geq 2|\Pr(A_0|RC_0) - \Pr(A_0|RC_\epsilon)|^2 \geq 1/2$$

- Employ chain rule to upper bound KL:

Remarkably, the proof applies to any decision rule; fundamentally, it is because Pinsker's inequality holds for any A_0

$$T \geq \frac{KL(p, q)}{\epsilon^2} \geq \frac{1}{2\epsilon^2}$$

Outline

- Technical Preparations
- Detour: Best-Arm Identification (BAI) Lower Bounds
- MAB Regret Lower Bounds
  - Instance-Independent Lower Bound
  - Instance-Dependent Lower Bounds

A Variant of MAB: Best-Arm Identification (BAI)

$1 : r_1 \sim D_1$

μ_1

. . .

$a : r_a \sim D_2$

μ_a

. . .

$k : r_k \sim D_k$

μ_k

- Same setup as MAB, but task is to identify best arm $i^* (= \arg\max_{i \in [k]} \mu_i)$
- Same strategy process of pulling arms $i^1, i^2, \dots, i^t, \dots, i^T$
- Given T rounds of opportunities, performance is measured by *probability of success* $\Pr(I^T = i^*)$

A Variant of MAB: Best-Arm Identification (BAI)

$1 : r_1 \sim D_1$

μ_1

. . .

$a\ : r_a \sim D_2$

μ_a

. . .

$k : r_k \sim D_k$

μ_k

- Same setup as MAB, but task is to identify best arm $i^*(= \arg\max_{i \in [k]} \mu_i)$
- Same strategy process of pulling arms $i^1, i^2, \dots, i^t, \dots, i^T$
- Given T rounds of opportunities, performance is measured by *probability of success* $\Pr(I^T = i^*)$

- Clearly, if T is very large, we can easily succeed with high prob.
- **Goal Next**: understand how large T needs to be in order to guarantee reasonable success on *any problem instance*

  By proving a statement of form "*if $T \le ??$, then for any algorithm will have at least constant probability of failing to find optimal arm on some instance*"

Imaging the Difficult Instances…

$1 : r_1 \sim D_1$

μ_1

. . .

$a\ :\ r_a \sim D_2$

μ_a

. . .

$k : r_k \sim D_k$

μ_k

What instance would be difficult for BAI?

- All arms have equal mean, except one of them that is slightly higher
  - Difficult since every sub-optimal arm is equally confusing
- Hopefully, each arm has large variance so rewards are random enough to “hide” the true mean
  - Interestingly, Bernoulli distributions (i.e., biased coins) turn out to already be sufficiently hard

Construction of Lower Bound Instances

$1: r_1 \sim D_1$

$\mu_1 = 1/2$

$a: r_a \sim D_2$

$\mu_a = (1+\epsilon)/2$

$k: r_k \sim D_k$

$\mu_k = 1/2$

- Each D_i is Bernoulli
- All of them are RC_0, except one arm a is RC_ϵ

Construction of Lower Bound Instances

$1 : r_1 \sim D_1$

$\mu_1 = 1/2$

. . .

$a : r_a \sim D_2$

$\mu_a = (1 + \epsilon)/2$

. . .

$k : r_k \sim D_k$

$\mu_k = 1/2$

- Each D_i is Bernoulli
- All of them are RC_0, except one arm a is RC_ϵ

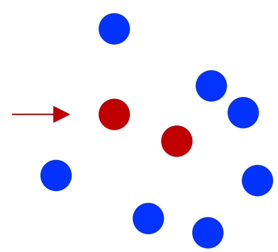
Remark.

This is not a single instance, but rather a set of k instances – each $a \in [k]$ correspond to one *problem instance* P_a

Formally, $P_a = \{k \text{ bandits with } D_a = RC_\epsilon, \text{all other } D_i = RC_0\}$

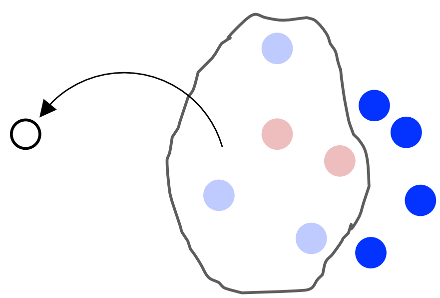
A Note on Lower Bound Proof Approaches

- Generally, two approaches to show an algorithm can perform bad on some instance
  1. Show that an algorithm does bad on some instance



A Note on Lower Bound Proof Approaches

- Generally, two approaches to show an algorithm can perform bad on some instance
  1. Show that an algorithm does bad on some instance
  2. Craft a set of instance, then randomly sample one for the algorithm to solve; show that the algo's expected performance is bad



A Note on Lower Bound Proof Approaches

- Generally, two approaches to show an algorithm can perform bad on some instance
  1. Show that an algorithm does bad on some instance
  2. Craft a set of instance, then randomly sample one for the algorithm to solve; show that the algo's expected performance is bad
- (2)⇒ (1) because if an algorithm perform bad in expectation, it must have performed bad in at least one instance on the support
- A stronger version of (1) ⇒ (2):

  if an algo does bad on a constant fraction of instances, then it has constant probability to perform bad on a randomly sampled instance
- (1) suffices for a lower bound proof, but we use (2) often due to proof convenience
- For our problem, we will use the set of instances $\{P_a\}_{a\in[k]}$

Lower Bounds for BAI

Theorem 0: Consider BAI with $T \leq \frac{ck}{\epsilon^2}$ on instances from set $\{P_a\}_{a \in [k]}$, where c is a small enough absolute constant.

For any deterministic algorithm for this problem, there exists at least $\lceil k/3 \rceil$ P_a instances such that

$$\Pr(I^T \neq a | P_a) \geq 1/2$$

$$P_a = \{k \text{ bandits with } D_a = RC_\epsilon, \text{all other } D_i = RC_0\}$$

Lower Bounds for BAI

Theorem 0: Consider BAI with $T \leq \frac{ck}{\epsilon^2}$ on instances from set $\{P_a\}_{a\in[k]}$, where c is a small enough absolute constant.

For any deterministic algorithm for this problem, there exists at least $\lceil k/3 \rceil$ P_a instances such that

$$\Pr(I^T \neq a | P_a) \geq 1/2$$

Corollary: Consider any BAI algorithm (possibly randomized) running on a uniformly randomly sampled instance from set $\{P_a\}_{a\in[k]}$ with $T \leq \frac{ck}{\epsilon^2}$.

Then $\Pr(I^T \neq i^*) \geq \frac{1}{6}$ where probability is over random choice of instance P_a, randomness of rewards and the algorithm.

- For deterministic algo, we have $\Pr(I^T \neq a | P_a) \geq 1/2$ for at least 1/3 of instances in $\{P_a\}_{a\in[k]}$ $\Rightarrow \Pr(I^T \neq i^*) \geq \frac{1}{2} \times \frac{1}{3} = \frac{1}{6}$ on a sampled instance
- Any randomized algorithm is a distribution over deterministic algorithm

  $\Rightarrow \Pr(I^T \neq i^*) \geq \frac{1}{6}$ by taking expectation over algo' randomness

Next: Proof of Theorem 0 in 3 Steps

Step 1: Converting the question to an instance testing problem by introducing a benchmark scenario

Introduce instance P_0, where all k arms are independent RC_0 (i.e., non-biased coins)

➢ Intuitions for remaining proofs

- We say an arm $j \in [k]$ is "neglected" by the algorithm if *(1) it was not played too often; (2) it has low probability to be the final output* I^T

- Will show under any deterministic algorithm to P_0, a constant fraction of arms are neglected

  because not all arms can be played a lot, simply by counting

Step 1: Converting the question to an instance testing problem by introducing a benchmark scenario

Introduce instance P_0, where all k arms are independent RC_0 (i.e., non-biased coins)

➢ Intuitions for remaining proofs

- We say an arm $j \in [k]$ is "neglected" by the algorithm if *(1) it was not played too often; (2) it has low probability to be the final output* I^T

- Will show under any deterministic algorithm to P_0, a constant fraction of arms are neglected

  because not all arms can be played a lot, simply by counting

- Now consider any neglected arm under the same algorithm in P_j

  $KL(P_j, P_0)$ is likely small since they only slightly differ on arm j,

  Pinsker's Inequality told us $\Pr\left(I^T \neq j \middle| P_j\right) - \Pr(I^T \neq j | P_0)$ must be small

Tricky part is to figure out how small this could *tightly* be!

Step 2: Characterizing "neglected arms" under any deterministic algorithm on benchmark instance P_0

Lemma 1: For any deterministic algorithm on P_0, there is a subset $J \subset [k]$ of arms such that

1) $|J| \geq k/3$
2) For any $j \in J$, $\mathbb{E}\left(N_j^T | P_0\right) \leq \frac{3T}{k}$
3) For any $j \in J$, $\Pr(I^T = j | P_0) \leq \frac{3}{k}$

Recall: I^T is the (random) arm pulled at last round T
N_j^T is the number of times arm j is pulled until round T

- That is, J contains all arms that are "neglected" in the sense of property 2) and 3)
- Property 1) says that J has size at least $k/3$

Step 2: Characterizing "neglected arms" under any deterministic algorithm on benchmark instance P_0

Lemma 1: For any deterministic algorithm on P_0, there is a subset $J \subset [k]$ of arms such that

1) $|J| \geq k/3$

2) For any $j \in J$, $\mathbb{E}\left(N_j^T | P_0\right) \leq \frac{3T}{k}$

3) For any $j \in J$, $\Pr(I^T = j | \, P_0) \leq \frac{3}{k}$

Intuition of the proof

- Follows from counting argument:
  - At least $2k/3$ arms satisfy property 2) since $\sum_{j=1}^{T} N_j^T = T$ is always true
  - At least $2k/3$ arms satisfy property 3) since $\sum_{j=1}^{T} \Pr\left(I_j^T = j\right) = 1$

Formal proof left to HW1!

Step 2: Characterizing "neglected arms" under any deterministic algorithm on benchmark instance P_0

Lemma 1: For any deterministic algorithm on P_0, there is a subset $J \subset [k]$ of arms such that

1) $|J| \geq k/3$

2) For any $j \in J$, $\Pr\left(N_j^T \leq \frac{24T}{k} \mid P_0\right) \geq \frac{7}{8}$

3) For any $j \in J$, $\Pr(I^T = j \mid P_0) \leq \frac{3}{k}$

Corollary: Property 2) above implies $\Pr\left(N_j^T \leq \frac{24T}{k} \mid P_0\right) \geq \frac{7}{8}$

Proof.

$$\Pr\left(N_j^T > \frac{24T}{k} \mid P_0\right) \leq \frac{\mathbb{E}\left(N_j^T \mid P_0\right)}{24T/k}$$

By Markov's inequality

$$\Pr(N > x) \leq \frac{\mathbb{E}(N)}{x}$$

$$\leq 1/8$$

By plugging in property 2)

This implies the corollary

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

The intuitive idea is straightforward

– Want to show KL divergence $KL(P_0, P_j)$ is upper bounded

– Pinsker's Inequality then implies if j is neglected under P_0, it will be under P_j as well

Technical argument needs careful treatment

– Simple argument yields $T \leq \frac{c}{\epsilon^2}$

– To get the stronger $T \leq \frac{ck}{\epsilon^2}$ bound, we need to carefully define the (random) events that determine a BAI algorithm's behavior

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- A deterministic BAI algorithm maps any observed reward sequence thus far to the next to-be-pulled arm

$$\text{Alg: } \{0,1\}^t \to [k]$$

- Such an Alg can be viewed as an adaptive way to open exactly T cells of a random reward table

$\longrightarrow N_i^t$

Arms

1							
2							
…							
j							
…							
k							

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- A deterministic BAI algorithm maps any observed reward sequence thus far to the next to-be-pulled arm

$$\text{Alg: } \{0,1\}^t \rightarrow [k]$$

- Such an Alg can be viewed as an adaptive way to open exactly T cells of a random reward table

N_i^t

Arms

1	0						
2	1	0	…				
…							
j	0	1	1	…			
…							
k	0	1	…				

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- We only care about event $\Pr(I^T = j)$
- Randomness purely comes from this random reward table
  - Since Alg is deterministic – it maps a sequence of T rewards to a deterministic choice of I^T

$\longrightarrow N_i^t$

Arms

1	0						
2	1	0	…				
…							
j	0	1	1	…			
…							
k	0	1	…				

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- We only care about event $\Pr(I^T = j)$
- Randomness purely comes from this random reward table
  - Since Alg is deterministic – it maps a sequence of T rewards to a deterministic choice of I^T
  - These T rewards can be from different rows/arms

$\xrightarrow{\qquad\qquad} N_i^t$

Arms

1	0						
2	1	0	…				
…							
j	0	1	1	…			
…							
k	0	1	…				

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- **Bad news**: generally, every reward cell below can possibly affect the algorithm
  - We particularly do not like that all T cells in j's row can affect Alg

$\longrightarrow N_i^t$ $\qquad\qquad$ T

Arms

1	0						
2	1	0	…				
…							
j	0	1	1	…			
…							
k	0	1	…				

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Bad news: generally, every reward cell below can possibly affect the algorithm
    - We particularly do not like that all T cells in j's row can affect Alg

      $\Rightarrow$ too much randomness that makes $KL(P_0, P_j)$ too large

      $\Rightarrow$ a non-tight bound c/ϵ^2

T

$\longrightarrow N_i^t$

Arms

1	0						
2	1	0	…				
…							
j	0	1	1	…			
…							
k	0	1	…				

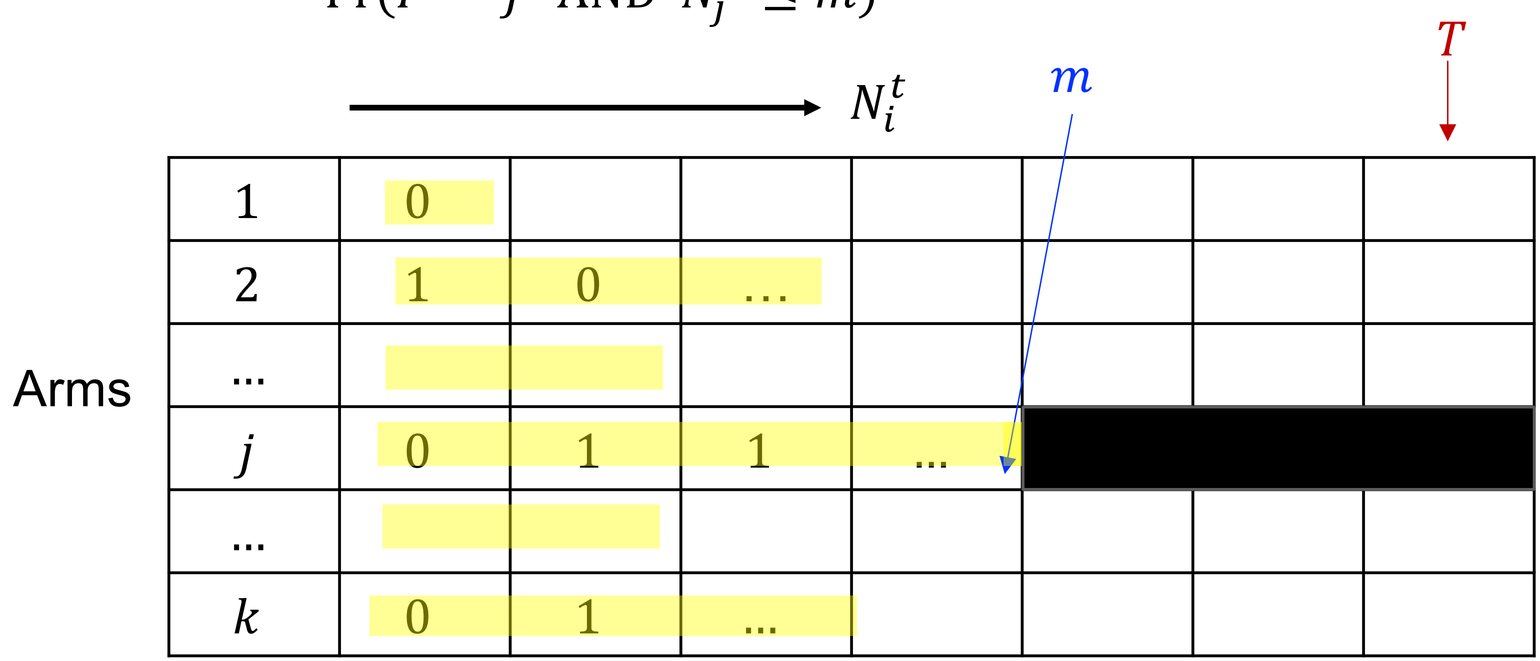
Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Key idea: only consider first $m = \min\{\frac{24T}{k},\ T\}$ cells in j'th row, though allow other rows' all random rewards (since they are equal under P_0, P_j)

Formally, consider

From Lemma 1, these are precisely the condition of "neglected arms"

$$\Pr(I^t = j \ \ \text{AND} \ \ N_j^T \le m)$$



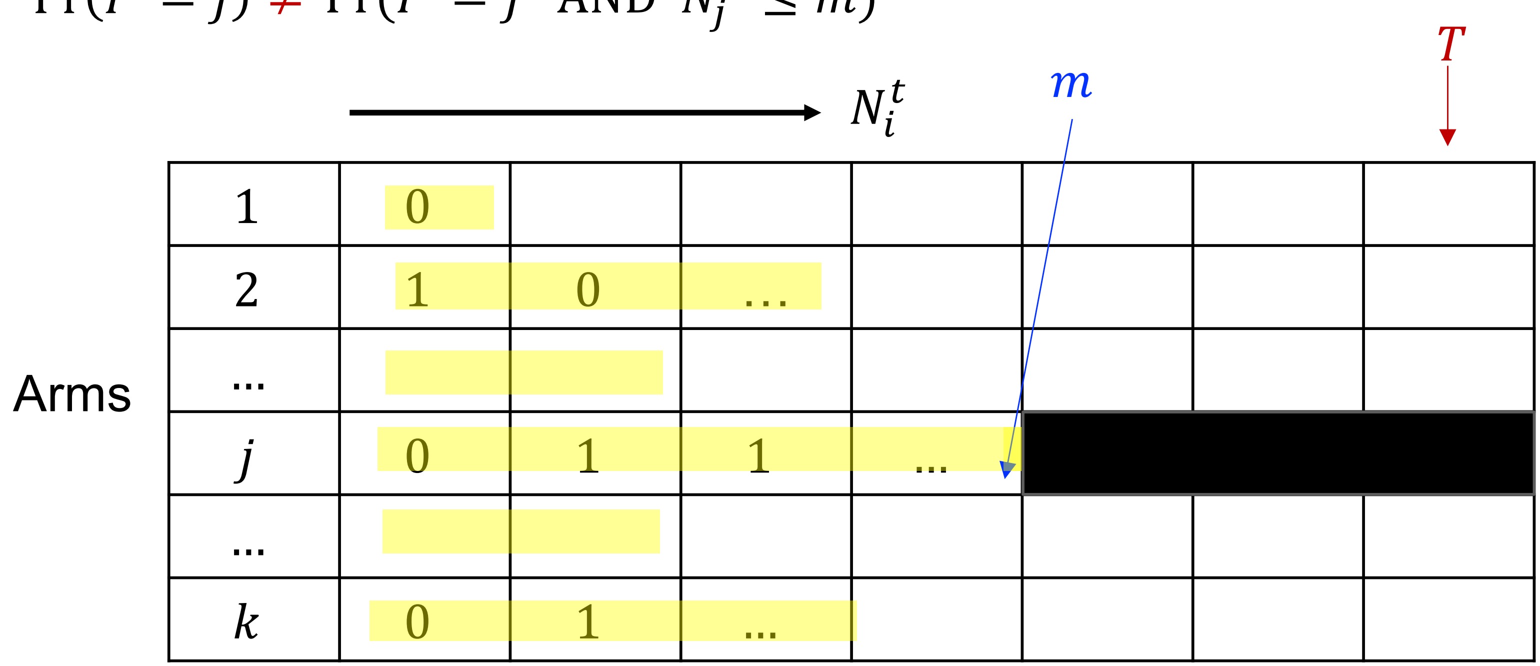
Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Key idea: only consider first $m = \min\{\frac{24T}{k},\ T\}$ cells in j'th row, though allow other rows' all random rewards (since they are equal under P_0, P_j)

Formally, consider

From Lemma 1, these are precisely the condition of "neglected arms"

$$\Pr(I^t = j) \neq \Pr(I^t = j \ \text{ AND } \ N_j^T \leq m)$$



Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Key idea: only consider first $m = \min\{\frac{24T}{k},\ T\}$ cells in j'th row, though allow other rows' all random rewards (since they are equal under P_0, P_j)

Formally, consider

$$\Pr(I^t = j) = \Pr(I^t = j \ \text{ AND } \ N_j^T \le m) + \Pr(I^t = j \ \text{ AND } \ N_j^T > m)$$

N_i^t $\quad m \quad T$

1	0						
2	1	0	…				
…							
j	0	1	1	…			
…							
k	0	1	…				

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

Both events depend only on first m rewards of row j

$$\le \overbrace{\Pr(I^t = j \ \ \text{AND} \ \ N_j^T \le m)}^{\text{Event } A_1} + \overbrace{\Pr(N_j^T > m)}^{\text{Event } A_2}$$

$$\Pr(I^t = j) = \Pr(I^t = j \ \ \text{AND} \ \ N_j^T \le m) + \Pr(I^t = j \ \ \text{AND} \ \ N_j^T > m)$$

$\longrightarrow N_i^t$ $\quad m$

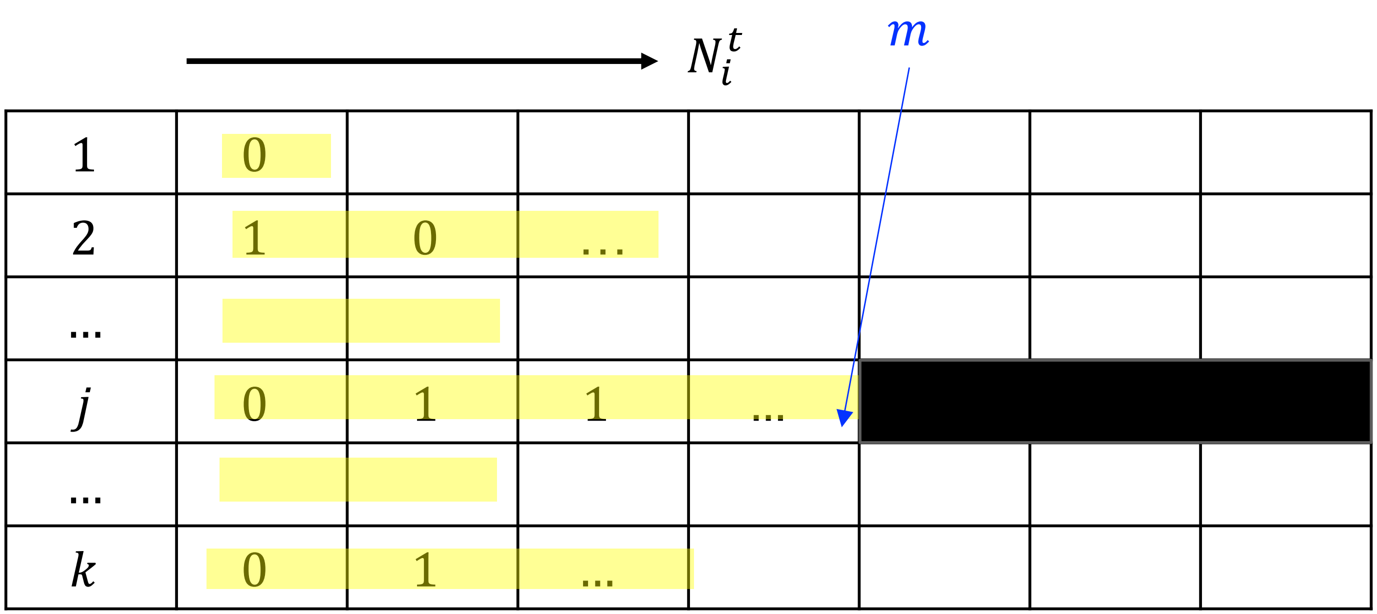
1	0						
2	1	0	…				
…							
j	0	1	1	…			
…							
k	0	1	…				

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Let $p_j^t = RC_0$ for $t = 1, 2, \dots, m$ and $p_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \dots, T$
- Let $q_j^t = RC_\epsilon$ for $t = 1, 2, \dots, m$ and $q_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \dots, T$
- Both event A_1, A_2 are in support of $p = \Pi_{i \neq j, t \in [T]} \mathrm{p}_i^t \cdot \Pi_{t \in [m]} p_j^t$ and a similarly defined q

$$\leq \overbrace{\Pr(I^t = j \ \text{ AND } \ N_j^T \leq m)}^{\text{Event } A_1} + \overbrace{\Pr(N_j^T > m)}^{\text{Event } A_2}$$

$$\Pr(I^t = j) = \Pr(I^t = j \ \text{ AND } \ N_j^T \leq m) + \Pr(I^t = j \ \text{ AND } \ N_j^T > m)$$



Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Let $p_j^t = RC_0$ for $t = 1, 2, \dots, m$ and $p_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \dots, T$
- Let $q_j^t = RC_\epsilon$ for $t = 1, 2, \dots, m$ and $q_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \dots, T$
- Both event A_1, A_2 are in support of $p = \Pi_{i \neq j, t \in [T]} \mathrm{p}_i^t \cdot \Pi_{t \in [m]} p_j^t$ and a similarly defined q

$$\Pr(I^t = j) \leq \Pr(\overbrace{I^t = j \ \text{ AND } \ N_j^T \leq m}^{\text{Event } A_1}) + \Pr(\overbrace{N_j^T > m}^{\text{Event } A_2})$$

By chain rule

$$KL(p, q) = \textstyle\sum_{i \neq j, t \in [T]} KL(p_i^t, q_i^t) + \sum_{t \in [m]} KL(p_j^t, q_j^t)$$

$$= \textstyle\sum_{i \neq j, t \in [T]} KL(RC_0, RC_0) + \sum_{t \in [m]} KL(RC_0, RC_\epsilon)$$

$$= m\, KL(RC_0, RC_\epsilon)$$

$$\leq \frac{24T}{k} \epsilon^2 \qquad \text{Since } m = \min\{\tfrac{24T}{k}, T\}$$

Theorem 0 assumed $T \leq \frac{ck}{\epsilon^2}$ for a small constant $c \Rightarrow KL(p, q) \leq \frac{1}{32}$

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Let $p_j^t = RC_0$ for $t = 1, 2, \dots, m$ and $p_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \dots, T$
- Let $q_j^t = RC_\epsilon$ for $t = 1, 2, \dots, m$ and $q_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \dots, T$
- Both event A_1, A_2 are in support of $p = \Pi_{i\neq j, t\in[T]} \mathrm{p}_i^t \cdot \Pi_{t\in[m]} p_j^t$ and a similarly defined q

$$\Pr(I^t = j) \leq \underbrace{\Pr(I^t = j \ \text{ AND } \ N_j^T \leq m)}_{\text{Event } A_1} + \underbrace{\Pr(N_j^T > m)}_{\text{Event } A_2}$$

Theorem 0 assumed $T \leq \frac{ck}{\epsilon^2}$ for a small constant $c \Rightarrow KL(p, q) \leq \frac{1}{32}$

$$\Rightarrow |\Pr(A|P_0) - \Pr(A|P_j)| \leq \sqrt{KL(p,q)/2} \leq \frac{1}{8}$$

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Let $p_j^t = RC_0$ for $t = 1, 2, \ldots, m$ and $p_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \ldots, T$
- Let $q_j^t = RC_\epsilon$ for $t = 1, 2, \ldots, m$ and $q_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \ldots, T$
- Both event A_1, A_2 are in support of $p = \Pi_{i \neq j, t \in [T]} \mathrm{p}_i^t \cdot \Pi_{t \in [m]} p_j^t$ and a similarly defined q

$$\Pr(I^t = j) \le \overbrace{\Pr(I^t = j \ \text{ AND } \ N_j^T \le m)}^{\text{Event } A_1} + \overbrace{\Pr(N_j^T > m)}^{\text{Event } A_2}$$

Theorem 0 assumed $T \le \frac{ck}{\epsilon^2}$ for a small constant $c \Rightarrow KL(p, q) \le \frac{1}{32}$

$$\Rightarrow |\Pr(A|P_0) - \Pr(A|P_j)| \le \sqrt{KL(p,q)/2} \le \frac{1}{8}$$

$$\begin{aligned}\Pr(A_1|P_j) &\le \Pr(A_1|P_0) + \frac{1}{8} \\ &\le \frac{3}{k} + \frac{1}{8} \quad \text{(by lemma 1)} \\ &\le \frac{1}{4}\end{aligned}$$

$$\begin{aligned}\Pr(A_2|P_j) &\le \Pr(A_2|P_0) + \frac{1}{8} \\ &\le \frac{1}{8} + \frac{1}{8} \quad \text{(by lemma 1)} \\ &\le \frac{1}{4}\end{aligned}$$

By considering instances with large k

Step 3: Upper bounding $\Pr(I^T = j)$ under instance P_j for any neglected arm j

- Let $p_j^t = RC_0$ for $t = 1, 2, \dots, m$ and $p_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \dots, T$
- Let $q_j^t = RC_\epsilon$ for $t = 1, 2, \dots, m$ and $q_i^t = RC_0$ for $i \neq j$ and $t = 1, 2, \dots, T$
- Both event A_1, A_2 are in support of $p = \Pi_{i \neq j, t \in [T]} \mathrm{p}_i^t \cdot \Pi_{t \in [m]} p_j^t$ and a similarly defined q

$$\Pr(I^t = j) \leq \overbrace{\Pr(I^t = j \ \ \mathrm{AND} \ \ N_j^T \leq m)}^{\text{Event } A_1} + \overbrace{\Pr(N_j^T > m)}^{\text{Event } A_2} \leq \frac{1}{2} \text{ on instance } P_j$$

Theorem 0 assumed $T \leq \frac{ck}{\epsilon^2}$ for a small constant $c \Rightarrow KL(p, q) \leq \frac{1}{32}$

$$\Rightarrow |\Pr(A|P_0) - \Pr(A|P_j)| \leq \sqrt{KL(p,q)/2} \leq \frac{1}{8}$$

$$\Pr(A_1|P_j) \leq \Pr(A_1|P_0) + \frac{1}{8}$$
$$\leq \frac{3}{k} + \frac{1}{8} \quad \text{(by lemma 1)}$$
$$\leq \frac{1}{4}$$

$$\Pr(A_2|P_j) \leq \Pr(A_2|P_0) + \frac{1}{8}$$
$$\leq \frac{1}{8} + \frac{1}{8} \quad \text{(by lemma 1)}$$
$$\leq \frac{1}{4}$$

By considering instances with large k

To Summarize

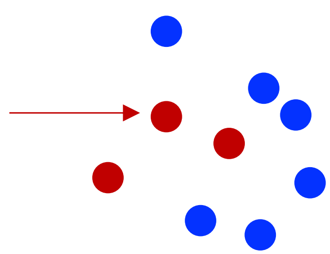
We proved

Theorem 0: Consider BAI with $T \leq \frac{ck}{\epsilon^2}$ on instances from set $\{P_a\}_{a\in[k]}$, where c is a small enough absolute constant.

For any deterministic algorithm for this problem, there exists at least $[k/3]$ P_a instances such that

$$\Pr(I^T \neq a | P_a) \geq 1/2$$

Notably, this theorem does not hold for randomized algorithm since the $[k/3]$ P_a instances may be different under different algorithm randomness



Any deterministic algorithm "fails" at a constant fraction of constructed instances

To Summarize

We proved

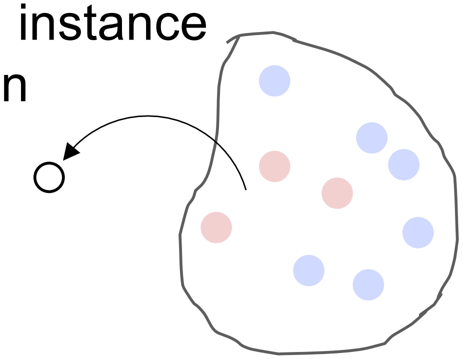
Theorem 0: Consider BAI with $T \le \frac{ck}{\epsilon^2}$ on instances from set $\{P_a\}_{a\in[k]}$, where c is a small enough absolute constant.

For any deterministic algorithm for this problem, there exists at least $\lceil k/3 \rceil$ P_a instances such that

$$\Pr(I^T \neq a | P_a) \ge 1/2$$

Notably, this theorem does not hold for randomized algorithm since the $\lceil k/3 \rceil$ P_a instances may be different under different algorithm randomness

Randomly sample an instance removes this limitation



Any deterministic algorithm "fails" at a constant fraction of constructed instances

To Summarize

We proved

Theorem 0: Consider BAI with $T \leq \frac{ck}{\epsilon^2}$ on instances from set $\{P_a\}_{a\in[k]}$, where c is a small enough absolute constant.

For any deterministic algorithm for this problem, there exists at least $\lceil k/3 \rceil$ P_a instances such that

$$\Pr(I^T \neq a | P_a) \geq 1/2$$

Notably, this theorem does not hold for randomized algorithm since the $\lceil k/3 \rceil$ P_a instances may be different under different algorithm randomness

Corollary: Consider any BAI algorithm (possibly randomized) running on a uniformly randomly sampled instance from set $\{P_a\}_{a\in[k]}$ with $T \leq \frac{ck}{\epsilon^2}$.

Then $\Pr(I^T \neq i^*) \geq \frac{1}{6}$ where probability is over random choice of instance P_a, randomness of rewards and the algorithm.

Outline

- Technical Preparations
- Detour: Best-Arm Identification (BAI) Lower Bounds
- MAB Regret Lower Bounds
  - Instance-Independent Lower Bound
  - Instance-Dependent Lower Bounds

Regret Lower Bound for MAB

Theorem 1: Fixed time horizon T and number of arms k.

For any bandit algorithm, running on a uniformly randomly sampled instance from $\{P_a\}_{a\in[k]}$ with $\epsilon = \sqrt{ck/T}$ for a sufficiently small constant c, we have

$$\mathbb{E}(R^T) \geq \Omega(\sqrt{kT})$$

where expectation is over choice of instance P_a, randomness in rewards and Algo.

Proof.

- Note that $T = ck/\epsilon^2$ by our choice of ϵ
- Previous corollary says any algorithm running on the stated random instance satisfies $\Pr(I^t \neq i^*) \geq \frac{1}{6}$ for any $t \leq ck/\epsilon^2 (= T)$
- This means we suffer expected regret $\geq \frac{1}{6} \times \frac{\epsilon}{2}$ at each round $t \leq T$ since in the constructed instance, any sub-optimal arm has $\Delta = \epsilon/2$
- In total, we have

$$\mathbb{E}(R^T) \geq \frac{\epsilon}{12} \times T = \Omega(\sqrt{kT})$$

Regret Lower Bound for MAB

Theorem 1: Fixed time horizon T and number of arms k.

For any bandit algorithm, running on a uniformly randomly sampled instance from $\{P_a\}_{a \in [k]}$ with $\epsilon = \sqrt{ck/T}$ for a sufficiently small constant c, we have

$$\mathbb{E}(R^T) \geq \Omega(\sqrt{kT})$$

where expectation is over choice of instance P_a, randomness in rewards and Algo.

Remarks.

- This is called "worst-case lower bound"
  - You designed an algorithm;
  - Someone tries to "stress test" your algorithm by trying to feeding in the most challenging instance
  - The bound captures the best you can do under such challenge
- Also known as "minimax lower bound"

$$\min_{\text{Algorithm}} \max_{\text{Instance}} Regret(\text{Algo}|\text{Ins})$$

Outline

- Technical Preparations
- Detour: Best-Arm Identification (BAI) Lower Bounds
- MAB Regret Lower Bounds
  - Instance-Independent Lower Bound
  - Instance-Dependent Lower Bounds

That is, remove that “max” in “minimax lower bound”, and derive a bound for every instance

Instance-Dependent Regret Lower Bound

Rough format of the statement

"*For any MAB problem instance* P *and time horizon* T, *no algorithm can achieve regret* $\mathbb{E}(R^T) = \mathrm{o}(\ Time_{P,T}\ )$"

- However, this claim is clearly not true → why?
  - Consider a trivial algorithm Alg_a which always pulls arm a
  - One of $\{\mathrm{Alg}_a\}_{a \in [k]}$ has 0 regret
- To have a meaningful result, we need to rule out such "pure luck" algorithms that fail miserably in general, but do well occasionally

Instance-Dependent Regret Lower Bound

Theorem 2. Consider any MAB algorithm that satisfies

$$\mathbb{E}(R^T) \leq O(C_{P,\alpha}\, T^\wedge\alpha) \qquad \text{for any } \alpha > 0 \text{ and any instance } P.$$

Then for any problem instance P, there exists a time T_0 such that for any $T \geq T_0$, we have

$$\mathbb{E}(R^T) \geq \mu^*(1-\mu^*) \sum_{i \neq i^*} \frac{\ln T}{\Delta_i}$$

- This is the restriction on the algorithms that we consider
  - That is, these are reasonable algorithms that attempted to solve all instances
- This bound shows that UCB's gap-dependent regret bound is tight order-wise

Thank You

Haifeng Xu
University of Chicago
haifengxu@uchicago.edu