Two examples of discrete PAC optimization

Emilie Kaufmann,

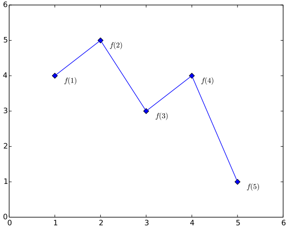
joint work with Aurélien Garivier and Wouter Koolen

Centre de Recherche en Informatique,
Signal et Automatique de Lille

GDR ISIS meeting
June 3rd, 2016

Generic discrete PAC optimization

$f : \{1, \ldots, K\} \to \mathbb{R}$



- a question $\mathcal{Q}(f)$, with (unknown) answer A^*
- find A^* using sequential noisy evaluations of f:

  querry i_t and observe $X_t : \mathbb{E}[X_t] = f(i_t)$.

PAC Learning framework: design a

sampling rule (i_t) / stopping rule τ / answering rule $\hat{A}$

such that $\mathbb{P}\left(\hat{A} = A^*\right) \geq 1 - \delta$, and $\mathbb{E}[\tau]$ as small as possible.

A particular example: (Bernoulli) bandit model

The querries of i are i.i.d. from a Bernoulli distribution of mean μ_i

$\mathcal{B}(\mu_1)$ $\mathcal{B}(\mu_2)$ $\mathcal{B}(\mu_3)$ $\mathcal{B}(\mu_4)$ $\mathcal{B}(\mu_5)$

$\Rightarrow$ Sequentially draw these "arms" to achieve a specific objective

- **Classical objective**: maximize the sum of "rewards" (reinforcement learning)
- **Alternative objective**: answer

$$\mathcal{Q}(\boldsymbol{\mu}) = \text{"which arm has highest mean?"}$$

(best arm identification)

- ... plenty of other objectives !

A particular example: (Bernoulli) bandit model

The querries of i are i.i.d. from a Bernoulli distribution of mean μ_i





$\mathcal{B}(\mu_1)$ $\mathcal{B}(\mu_2)$ $\mathcal{B}(\mu_3)$ $\mathcal{B}(\mu_4)$ $\mathcal{B}(\mu_5)$

$\Rightarrow$ Sequentially draw these "arms" to achieve a specific objective

- **Classical objective**: maximize the sum of "rewards" (reinforcement learning)
- **Alternative objective**: answer

$$\mathcal{Q}(\boldsymbol{\mu}) = \text{"which arm has highest mean?"}$$

(best arm identification)

- ... plenty of other objectives !

Optimal Best Arm Identification with Fixed Confidence

A. Garivier and E. Kaufmann
to appear in COLT 2016

The best arm identification problem

A Bernoulli bandit model is denoted by $\boldsymbol{\mu} = (\mu_1, \mu_2, \dots, \mu_K)$

$$\mathcal{S} = \left\{ \boldsymbol{\mu} \in [0,1]^K : \exists a \in \{1, \dots, K\} : \mu_a > \max_{i \neq a} \mu_i \right\}$$

A strategy is made of

- a sampling rule: which arm A_t is chosen at round t?
- a stopping rule τ: when should we stop sampling the arms?
- a recommendation rule $\hat{a}$: a guess for $a^* = \underset{a}{\text{argmax}}\ \mu_a$

The strategy should satisfy

- $\forall \boldsymbol{\mu} \in \mathcal{S}$, $\mathbb{P}_{\boldsymbol{\mu}}(\hat{a} = a^*) \geq 1 - \delta$ (δ-PAC strategy)
- for all $\boldsymbol{\mu} \in \mathcal{S}$, the sample complexity $\mathbb{E}_{\boldsymbol{\mu}}[\tau]$ is small.

The best arm identification problem

A Bernoulli bandit model is denoted by $\boldsymbol{\mu} = (\mu_1, \mu_2, \dots, \mu_K)$

$$\mathcal{S} = \left\{ \boldsymbol{\mu} \in [0,1]^K : \exists a \in \{1, \dots, K\} : \mu_a > \max_{i \neq a} \mu_i \right\}$$

A strategy is made of

- a sampling rule: which arm A_t is chosen at round t?
- a stopping rule τ: when should we stop sampling the arms?
- a recommendation rule $\hat{a}$: a guess for $a^* = \operatorname{argmax}_a \mu_a$

The strategy should satisfy

- $\forall \boldsymbol{\mu} \in \mathcal{S}$, $\mathbb{P}_{\boldsymbol{\mu}}(\hat{a} = a^*) \geq 1 - \delta$ (δ-PAC strategy)
- for all $\boldsymbol{\mu} \in \mathcal{S}$, the sample complexity $\mathbb{E}_{\boldsymbol{\mu}}[\tau]$ is small.

All the results are stated for $\boldsymbol{\mu} \in \mathcal{S} : \mu_1 > \mu_2 \geq \dots \geq \mu_K$.

A Racing algorithm

Successive Elimination [Even Dar el al. 06]

- At start, all arms are active;
- Then, repeatedly cycle thru active arms until only one arm is still active
- At the end of a cycle, eliminate arms with statistical evidence of sub-optimality: desactivate a if

$$\max_i \hat{\mu}_i(t) - \hat{\mu}_a(t) \geq 2\sqrt{\frac{\log(Kt^2/\delta)}{t}}$$

Output: the single active arm $\hat{a}$

Theorem

Successive Elimination is δ-PAC and with probability $1-\delta$,

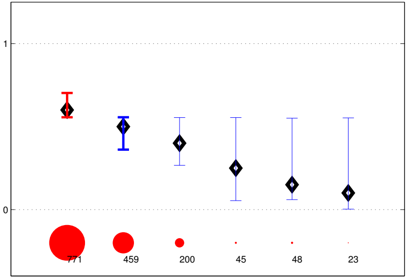
$$\tau_\delta = O\left(\sum_{a=2}^{K} \frac{\log \frac{K}{\delta \Delta_a}}{\Delta_a^2}\right),$$

with $\Delta_a = \mu_1 - \mu_a$.

The LUCB algorithm

An algorithm based on confidence intervals

$$\mathcal{I}_a(t) = [\mathrm{LCB}_a(t), \mathrm{UCB}_a(t)].$$



- At round t, draw

$$L_t = \underset{a}{\arg\max}\ \hat{\mu}_a(t)$$
$$C_t = \underset{a \neq L_t}{\arg\max}\ \mathrm{UCB}_a(t)$$

- Stop at round t if

$$\mathrm{LCB}_{L_t}(t) > \mathrm{UCB}_{C_t}(t)$$

Theorem [Kalyanakrishan et al.]

For well-chosen confidence intervals, LUCB is δ-PAC and

$$\mathbb{E}[\tau_\delta] = O\left(\left[\frac{1}{\Delta_2^2} + \sum_{a=2}^{K} \frac{1}{\Delta_a^2}\right] \log\left(\frac{1}{\delta}\right)\right)$$

A new lower bound

Notation: Kullback-Leibler divergence

$$\begin{aligned} d(\mu, \mu') &:= \mathrm{KL}\left(\mathcal{B}(\mu), \mathcal{B}(\mu')\right) \\ &= \mu \log(\mu/\mu') + (1-\mu)\log((1-\mu)/(1-\mu')) \end{aligned}$$

From Pinsker inequality, $d(\mu_a, \mu_1) > 2\Delta_a^2$.

A new lower bound

Notation: Kullback-Leibler divergence

$$
\begin{aligned}
d(\mu, \mu') &:= \mathrm{KL}\left(\mathcal{B}(\mu), \mathcal{B}(\mu')\right) \\
&= \mu \log(\mu/\mu') + (1-\mu)\log((1-\mu)/(1-\mu'))
\end{aligned}
$$

From Pinsker inequality, $d(\mu_a, \mu_1) > 2\Delta_a^2$.

Theorem

For any δ-PAC algorithm,

$$
\mathbb{E}_{\boldsymbol{\mu}}[\tau] \geq T^*(\boldsymbol{\mu}) \log\left(\frac{1}{2.4\delta}\right),
$$

where

$$
T^*(\boldsymbol{\mu})^{-1} = \sup_{w \in \Sigma_K} \inf_{\boldsymbol{\lambda} \in \mathrm{Alt}(\boldsymbol{\mu})} \left(\sum_{a=1}^{K} w_a d(\mu_a, \lambda_a) \right)
$$

with $\Sigma_K = \{w \in [0,1]^K : \sum_{i=1}^K w_i = 1\}$ and

$$
\mathrm{Alt}(\boldsymbol{\mu}) = \{\boldsymbol{\lambda} \in \mathcal{S} : a^*(\boldsymbol{\lambda}) \neq a^*(\boldsymbol{\mu})\}.
$$

Where does it come from ?

Change of distribution Lemma [K., Cappé, Garivier 15]

If $a^*(\boldsymbol{\mu}) \neq a^*(\boldsymbol{\lambda})$, any δ-PAC algorithm satisfies

$$\sum_{a=1}^{K} \mathbb{E}_{\boldsymbol{\mu}}[N_a(\tau)] d(\mu_a, \lambda_a) \geq \log\left(\frac{1}{2.4\delta}\right).$$

Letting $\mathrm{Alt}(\boldsymbol{\mu}) = \{\boldsymbol{\lambda} : a^*(\boldsymbol{\lambda}) \neq a^*(\boldsymbol{\mu})\}$,

$$\begin{aligned}
\inf_{\boldsymbol{\lambda} \in \mathrm{Alt}(\boldsymbol{\mu})} \sum_{a=1}^{K} \mathbb{E}_{\boldsymbol{\mu}}[N_a(\tau)] d(\mu_a, \lambda_a) &\geq \log\left(\frac{1}{2.4\delta}\right) \\
\mathbb{E}_{\boldsymbol{\mu}}[\tau] \times \inf_{\boldsymbol{\lambda} \in \mathrm{Alt}(\boldsymbol{\mu})} \sum_{a=1}^{K} \frac{\mathbb{E}_{\boldsymbol{\mu}}[N_a(\tau)]}{\mathbb{E}_{\boldsymbol{\mu}}[\tau]} d(\mu_a, \lambda_a) &\geq \log\left(\frac{1}{2.4\delta}\right) \\
\mathbb{E}_{\boldsymbol{\mu}}[\tau] \times \left(\sup_{w \in \Sigma_K} \inf_{\boldsymbol{\lambda} \in \mathrm{Alt}(\boldsymbol{\mu})} \sum_{a=1}^{K} w_a d(\mu_a, \lambda_a) \right) &\geq \log\left(\frac{1}{2.4\delta}\right)
\end{aligned}$$

Optimal proportion of draws

The vector

$$w^*(\boldsymbol{\mu}) = \underset{w \in \Sigma_K}{\text{argmax}} \inf_{\lambda \in \mathrm{Alt}(\boldsymbol{\mu})} \left(\sum_{a=1}^{K} w_a d(\mu_a, \lambda_a) \right)$$

contains the optimal proportions of draws of the arms, i.e. an algorithm matching the lower bound should satisfy

$$\forall a \in \{1, \dots, K\}, \quad \frac{\mathbb{E}_{\boldsymbol{\mu}}[N_a(\tau)]}{\mathbb{E}_{\boldsymbol{\mu}}[\tau]} \simeq w_a^*(\boldsymbol{\mu}).$$

We show that:

- ➜ $w^*(\boldsymbol{\mu})$ is well-defined (unique maximizer)
- ➜ $w^*(\boldsymbol{\mu})$ can be computed efficiently for all $\boldsymbol{\mu}$

Sampling rule: Tracking the optimal proportions

$\hat{\boldsymbol{\mu}}(t) = (\hat{\mu}_1(t), \ldots, \hat{\mu}_K(t))$: vector of empirical means

- Introducing

$$U_t = \{a : N_a(t) < \sqrt{t}\},$$

the arm sampled at round $t+1$ is

$$A_{t+1} \in \begin{cases} \underset{a \in U_t}{\operatorname{argmin}} \; N_a(t) \text{ if } U_t \neq \emptyset & (\textit{forced exploration}) \\ \underset{1 \leq a \leq K}{\operatorname{argmax}} \left[t \; w_a^*(\hat{\boldsymbol{\mu}}(t)) - N_a(t)\right] & (\textit{tracking}) \end{cases}$$

Lemma

Under the Tracking sampling rule,

$$\mathbb{P}_{\boldsymbol{\mu}}\left(\lim_{t\to\infty} \frac{N_a(t)}{t} = w_a^*(\boldsymbol{\mu})\right) = 1.$$

Stopping rule: performing statistical tests

High values of the Generalized Likelihood Ratio

$$Z_{a,b}(t) := \log \frac{\max_{\{\boldsymbol{\lambda}:\lambda_a \geq \lambda_b\}} \ell(X_1,\dots,X_t;\boldsymbol{\lambda})}{\max_{\{\boldsymbol{\lambda}:\lambda_a \leq \lambda_b\}} \ell(X_1,\dots,X_t;\boldsymbol{\lambda})},$$

reject the hypothesis that $(\mu_a < \mu_b)$.

We stop when one arm is accessed to be significantly larger than all other arms, according to a SGLR Test:

$$\begin{aligned}\tau_\delta &= \inf\{t \in \mathbb{N} : \exists a \in \{1,\dots,K\}, \forall b \neq a, Z_{a,b}(t) > \beta(t,\delta)\} \\ &= \inf\left\{t \in \mathbb{N} : \max_{a\in\{1,\dots,K\}} \min_{b\neq a} Z_{a,b}(t) > \beta(t,\delta)\right\}\end{aligned}$$

Chernoff stopping rule [Chernoff 59]

A δ-PAC stopping rule

One has $Z_{a,b}(t) = -Z_{b,a}(t)$ and, if $\hat{\mu}_a(t) \geq \hat{\mu}_b(t)$,

$$Z_{a,b}(t) = N_a(t)\, d\left(\hat{\mu}_a(t), \hat{\mu}_{a,b}(t)\right) + N_b(t)\, d\left(\hat{\mu}_b(t), \hat{\mu}_{a,b}(t)\right),$$

where $\hat{\mu}_{a,b}(t) := \frac{N_a(t)}{N_a(t)+N_b(t)}\hat{\mu}_a(t) + \frac{N_b(t)}{N_a(t)+N_b(t)}\hat{\mu}_b(t)$.

A link with the lower bound

$$\max_a \min_{b \neq a} Z_{a,b}(t) = t \times \left(\inf_{\boldsymbol{\lambda} \in \mathrm{Alt}(\hat{\mu}(t))} \sum_{a=1}^{K} \frac{N_a(t)}{t} d(\hat{\mu}_a(t), \lambda_a) \right) \simeq \frac{t}{T^*(\boldsymbol{\mu})}$$

under a "good" sampling strategy (for t large)

Lemma

If $\mu_a < \mu_b$, for any sampling rule it holds that

$$\mathbb{P}_{\boldsymbol{\mu}}\left(\exists t \in \mathbb{N} : Z_{a,b}(t) > \log(2t/\delta)\right) \leq \delta.$$

An asymptotically optimal algorithm

Theorem [K. and Garivier, 2016]

The Track-and-Stop strategy, that uses

- the Tracking sampling rule
- the Chernoff stopping rule with $\beta(t,\delta) = \log\left(\frac{2(K-1)t}{\delta}\right)$
- and recommends $\hat{a} = \underset{a=1\dots K}{\operatorname{argmax}}\ \hat{\mu}_a(\tau)$

is δ-PAC for every $\delta \in]0,1[$ and satisfies

$$\limsup_{\delta\to 0} \frac{\mathbb{E}_{\boldsymbol{\mu}}[\tau_\delta]}{\log(1/\delta)} = T^*(\boldsymbol{\mu}).$$

Numerical experiments

Experiments on two Bernoulli bandit models:

- $\boldsymbol{\mu}_1 = [0.5\ 0.45\ 0.43\ 0.4]$, such that
$$w^*(\boldsymbol{\mu}_1) = [0.417\ 0.390\ 0.136\ 0.057]$$
- $\boldsymbol{\mu}_2 = [0.3\ 0.21\ 0.2\ 0.19\ 0.18]$, such that
$$w^*(\boldsymbol{\mu}_2) = [0.336\ 0.251\ 0.177\ 0.132\ 0.104]$$

In practice, set the threshold to $\beta(t,\delta) = \log\left(\frac{\log(t)+1}{\delta}\right)$.

	Track-and-Stop	Chernoff-Racing	KL-LUCB	KL-Racing
$\boldsymbol{\mu}_1$	4052	4516	8437	9590
$\boldsymbol{\mu}_2$	1406	3078	2716	3334

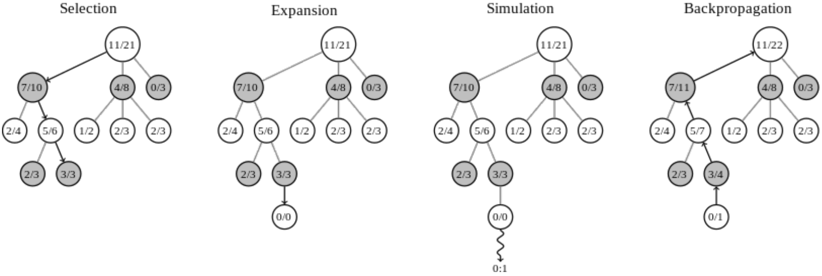
Table : Expected number of draws $\mathbb{E}_{\boldsymbol{\mu}}[\tau_\delta]$ for $\delta = 0.1$, averaged over $N = 3000$ experiments.

Maximin Action Identification: A New Bandit Framework for Games

A. Garivier, E. Kaufmann, W. Koolen,

to appear in COLT 2016

Monte-Carlo Tree Search for games



We introduce an idealized model:

- depth-two complete tree
- perfect rollouts

and give sample complexity guarantees in a PAC framework.

Towards another discrete PAC optimization problem

Consider a two-player game in which

- when A chooses action $i \in \{1, \ldots, K\}$
- and then player B choose action $j \in \{1, \ldots, K_i\}$,

the probability that A wins is $\mu_{i,j}$.

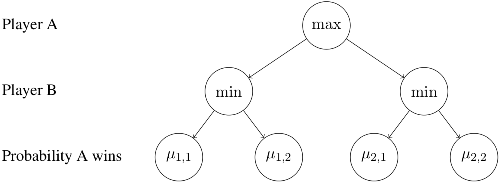


Figure 1: Game tree when there are two actions per player ($K = K_1 = K_2 = 2$).

Best action for A given that B is strategic:

$$i^* \in \underset{i \in \{1,\ldots,K\}}{\operatorname{argmax}} \ \min_{j \in \{1,\ldots,K_i\}} \mu_{i,j} \quad \text{(maximin action)}$$

Goal: Learn i^* by sequentially choosing pairs of actions $P = (i, j)$ and observing samples from $\mathcal{B}(\mu_{i,j})$ ("rollouts")

Towards another discrete PAC optimization problem

Consider a two-player game in which

- when A chooses action $i \in \{1, \ldots, K\}$
- and then player B choose action $j \in \{1, \ldots, K_i\}$,

the probability that A wins is $\mu_{i,j}$.

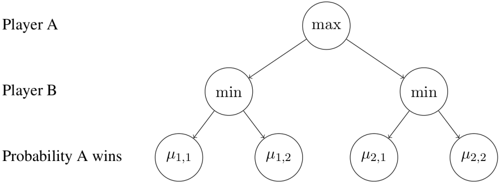


Figure 1: Game tree when there are two actions per player ($K = K_1 = K_2 = 2$).

Best action for A given that B is strategic:

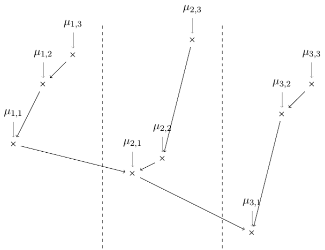
$$i^* \in \underset{i \in \{1,\ldots,K\}}{\text{argmax}} \ \min_{j \in \{1,\ldots,K_i\}} \mu_{i,j} \quad \text{(maximin action)}$$

Goal: Learn i^* by sequentially choosing pairs of actions $P = (i, j)$ and observing samples from $\mathcal{B}(\mu_{i,j})$ ("rollouts") $\Rightarrow$ Depth 2 MCTS

Maximin action identification

A bandit model parametrized by $\boldsymbol{\mu} = (\mu_{i,j})_{\substack{1 \leq i \leq K, \\ 1 \leq j \leq K_i}}$

$\mathcal{Q}(\boldsymbol{\mu})$: What is the maximin action? i.e. find $i^* = \underset{i}{\arg\max} \underset{j}{\min} \ \mu_{i,j}$



Goal: Build a strategy $(P_t, \tau, \hat{\imath})$ such that

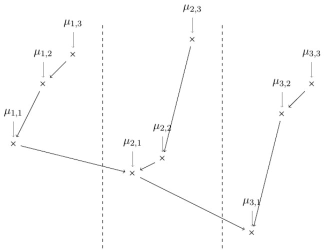
$$\forall \boldsymbol{\mu}, \ \mathbb{P}_{\boldsymbol{\mu}}\left(\min_{j \in \{1 \dots K_{i^*}\}} \mu_{i^*,j} - \min_{j \in \{1 \dots K_{\hat{\imath}}\}} \mu_{\hat{\imath},j} \leq \epsilon \right) \geq 1 - \delta,$$

and $\mathbb{E}_{\boldsymbol{\mu}}[\tau]$ is as small as possible.

Maximin action identification

A bandit model parametrized by $\boldsymbol{\mu} = (\mu_{i,j})_{\substack{1\leq i\leq K, \\ 1\leq j\leq K_i}}$

$\mathcal{Q}(\boldsymbol{\mu})$: What is the maximin action? i.e. find $i^* = \operatorname{arg\,max}_i \min_j \; \mu_{i,j}$

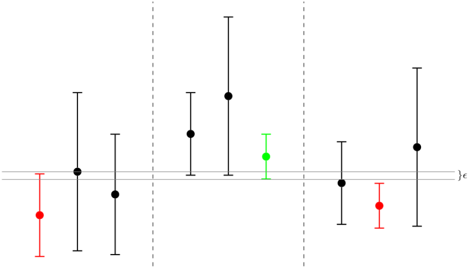


Goal: Build a strategy $(P_t, \tau, \hat{\imath})$ such that

$$\forall \boldsymbol{\mu}, \; \mathbb{P}_{\boldsymbol{\mu}}\Big(\mu_{1,1} - \mu_{\hat{\imath},1} \leq \epsilon\Big) \geq 1 - \delta,$$

and $\mathbb{E}_{\boldsymbol{\mu}}[\tau]$ is as small as possible.

The Maximin-LUCB algorithm



- Pick one representative per action $P_i = (i, j_i)$,

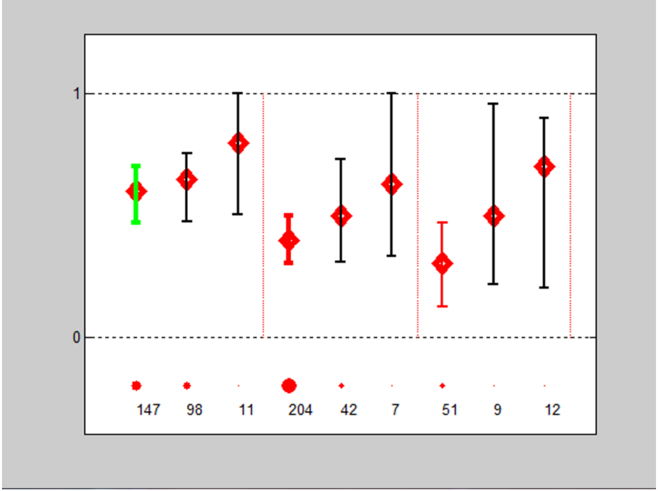
$$j_i = \underset{j}{\arg\max}\, \mathrm{LCB}_{(i,j)}(t)$$

- Letting $\hat{\imath}(t) = \arg\max_i \min_j \hat{\mu}_{(i,j)}(t)$, draw

$$L_t = (\hat{\imath}(t), j_{\hat{\imath}(t)}) \quad \text{and} \quad C_t = \underset{P \in \{(i,j_i)\}_{i \neq \hat{\imath}(t)}}{\arg\max} \mathrm{UCB}_P(t)$$

- Stop if $\mathrm{LCB}_{L_t}(t) > \mathrm{UCB}_{C_t}(t) - \epsilon$

M-LUCB in action !



Sample complexity analysis

$$\mathrm{LCB}_P(t) = \hat{\mu}_P(t) - \sqrt{\frac{\beta(t,\delta)}{2N_P(t)}}, \quad \mathrm{UCB}_P(t) = \hat{\mu}_P(t) + \sqrt{\frac{\beta(t,\delta)}{2N_P(t)}}$$

Theorem

Let $\alpha > 1$. There exists $C > 0$ such that for the choice

$$\beta(t,\delta) = \log(Ct^{1+\alpha}/\delta),$$

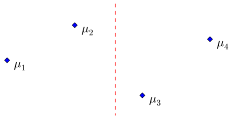
M-LUCB is δ-PAC and

$$\limsup_{\delta\to 0} \frac{\mathbb{E}_{\boldsymbol{\mu}}[\tau_\delta]}{\log(1/\delta)} \leq 8(1+\alpha)H^*(\boldsymbol{\mu})$$

$$H^*(\boldsymbol{\mu}) := \sum_{(1,j)\in\mathcal{P}_1} \frac{1}{(\mu_{1,j}-\mu_{2,1})^2} + \sum_{(i,j)\in\mathcal{P}\backslash\mathcal{P}_1} \frac{1}{(\mu_{1,1}-\mu_{i,1})^2 \vee (\mu_{i,j}-\mu_{i,1})^2}.$$

Lower bound and optimal algorithm ?

2 actions by player:



Theorem

Any δ-PAC algorithm satisfies

$$\mathbb{E}_{\boldsymbol{\mu}}[\tau] \geq P^*(\boldsymbol{\mu}) \log(1/(2.4\delta)),$$

where

$$P_*^{-1}(\boldsymbol{\mu}) = \max_{w \in \Sigma_4} \inf_{\boldsymbol{\mu}' : \mu_1' \wedge \mu_2' < \mu_3' \wedge \mu_4'} \left(\sum_{a=1}^{4} w_a \, d(\mu_a, \mu_a') \right)$$

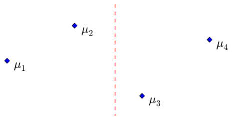
<u>Particular case:</u> if $\mu_4 > \mu_2$,

$$w^*(\boldsymbol{\mu}) = \operatorname*{argmax}_{w \in \Sigma_4} \inf_{\boldsymbol{\mu}' : \mu_1' \wedge \mu_2' < \mu_3' \wedge \mu_4'} \left(\sum_{a=1}^{4} w_a \, d(\mu_a, \mu_a') \right)$$

can be computed and $w_4^*(\boldsymbol{\mu}) = 0$!

Lower bound and optimal algorithm ?

2 actions by player:



$$w^*(\boldsymbol{\mu}) = \underset{w \in \Sigma_4}{\operatorname{argmax}} \inf_{\boldsymbol{\mu}' \in \mathrm{Alt}(\boldsymbol{\mu})} \left(\sum_{a=1}^{4} w_a \, d(\mu_a, \mu'_a) \right)$$

Assuming, in general, that $w^*(\boldsymbol{\mu})$ is unique and well-behaved, with

$$\hat{Z}(t) = \inf_{\boldsymbol{\mu}' \in \mathrm{Alt}(\hat{\boldsymbol{\mu}}(t))} \sum_{a=1}^{4} N_a(t) d(\hat{\mu}_a(t), \mu'_a),$$

a strategy such that $\frac{N_a(t)}{t} \to w_a^*(\boldsymbol{\mu})$ and

$$\tau = \inf\{t \in \mathbb{N} : \hat{Z}(t) \geq \log(Ct/\delta)\},$$

would satisfy $\tau_\delta \leq P^*(\boldsymbol{\mu}) \log(1/\delta) + o(\log(1/\delta))$, a.s.

Conclusion and perspectives

For the best arm identification problem:

- we exhibit a (non-explicit) characteristic time $T^*(\boldsymbol{\mu})$
- we propose an (efficient !) asymptotically optimal algorithm
- ... finite-time analysis of strategies inspired by other successful heuristics? (UCB/Thompson Sampling)

Remark: BAI $\neq$ regret minimization $(w^*(\boldsymbol{\mu}) \neq \mathbb{1}_{a^*})$

For depth-two MCTS:

- we devise efficient algorithms building on BAI tools, with sample complexity guarantees
- optimal strategies remain to be characterized
- ... we need to go deeper !