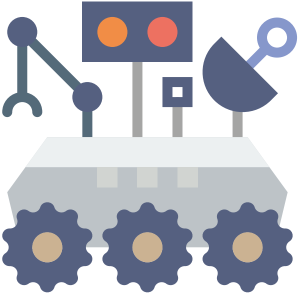
Exploration and Exploitation in Structured Stochastic Bandits



Wouter M. Koolen

CWI

Centrum Wiskunde & Informatica

Collaborators

Rémy Degenne

Han Shao (邵涵)

Emilie Kaufmann

Pierre Ménard

Aurélien Garivier

Part I

Saddle Points Intermezzo

Objective function

$$g(x, y)$$

convex in x, concave in y.

Games

Objective function

$$g(x, y)$$

convex in x, concave in y.
The game **value** is

$$V^* = \inf_x \sup_y g(x, y) = \sup_y \inf_x g(x, y).$$

Games

Objective function

$$g(x, y)$$

convex in x, concave in y.
The game **value** is

$$V^* = \inf_x \sup_y g(x, y) = \sup_y \inf_x g(x, y).$$

Definition

An ϵ-**saddle point** $(\bar{x}, \bar{y})$ satisfies

$$V^* - \epsilon \leq \inf_x g(x, \bar{y}) \leq V^* \leq \sup_y g(\bar{x}, y) \leq V^* + \epsilon.$$

Question: how to find ϵ-saddle point?

Iterative Algorithm

Idea [Freund and Schapire, 1999]: play a regret minimisation algorithm for x against one for y.

- Players play x_t and y_t.
- Players see loss functions

$$\begin{aligned} x &\mapsto +g(x, y_t), \\ y &\mapsto -g(x_t, y). \end{aligned}$$

Output pair of **iterate averages**: $\left(\frac{1}{T}\sum_{t=1}^T x_t, \frac{1}{T}\sum_{t=1}^T y_t\right)$.

Saddle point

Assume the players have regret (bounds) R_T^x and R_T^y, i.e.

$$\begin{aligned}\sum_{t=1}^T +g(x_t, y_t) - \inf_x \sum_{t=1}^T +g(x, y_t) &\leq R_T^x \\ \sum_{t=1}^T -g(x_t, y_t) - \inf_y \sum_{t=1}^T -g(x_t, y) &\leq R_T^y\end{aligned}$$

Theorem

The **iterate averages** $\bar{x}_T = \frac{1}{T} \sum_{t=1}^T x_t$ *and* $\bar{y}_T = \frac{1}{T} \sum_{t=1}^T y_t$ *form an* $\frac{R_T^x + R_T^y}{T}$*-saddle point.*

Analysis

$$
\begin{aligned}
V^* &= \inf_x \sup_y g(x,y) \\
&\le \sup_y g(\bar{x}_T, y) && \text{(suboptimal choice } \bar{x}_T\text{)} \\
&\le \sup_y \frac{1}{T} \sum_{t=1}^T g(x_t, y) && \text{(convexity in 1st argument)} \\
&\le \frac{1}{T} \sum_{t=1}^T g(x_t, y_t) + \frac{R_T^y}{T} && \text{(}y\text{ player regret guarantee)} \\
&\le \inf_x \frac{1}{T} \sum_{t=1}^T g(x, y_t) + \frac{R_T^x + R_T^y}{T} && \text{(}x\text{ player regret guarantee)} \\
&\le \inf_x g(x, \bar{y}_T) + \frac{R_T^x + R_T^y}{T} && \text{(concavity in 2nd argument)} \\
&\le \inf_x \sup_y g(x,y) + \frac{R_T^x + R_T^y}{T} && \text{(suboptimal choice } \bar{y}_T\text{)} \\
&= V^* + \frac{R_T^x + R_T^y}{T}
\end{aligned}
$$

Applications

Choice of online learning algorithm governed by domain of $g(x, y)$.

Applications

Choice of online learning algorithm governed by domain of $g(x, y)$.

For either player, we can use e.g.

- OGD, Hedge, FTRL, FTPL, ... $\sqrt{T}$ **regret** $\Rightarrow \epsilon = 1/\sqrt{T}$

Applications

Choice of online learning algorithm governed by domain of $g(x, y)$.

For either player, we can use e.g.

- OGD, Hedge, FTRL, FTPL, ... $\sqrt{T}$ **regret** $\Rightarrow \epsilon = 1/\sqrt{T}$

while the **second** player can ensure **negative regret** with

- Be-The-Leader
- Best Response **no memory**

Applications

Choice of online learning algorithm governed by domain of $g(x, y)$.

For either player, we can use e.g.

- OGD, Hedge, FTRL, FTPL, ... $\sqrt{T}$ **regret** $\Rightarrow \epsilon = 1/\sqrt{T}$

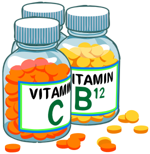
while the **second** player can ensure **negative regret** with

- Be-The-Leader
- Best Response **no memory**

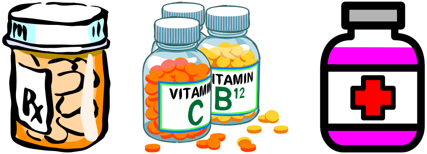
Many more options. Optimism [Rakhlin and Sridharan, 2013], principled path to Nesterov acceleration [Wang and Abernethy, 2018]

Outline

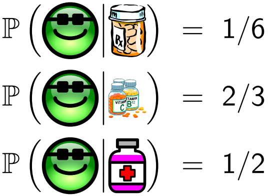
Stochastic Bandit



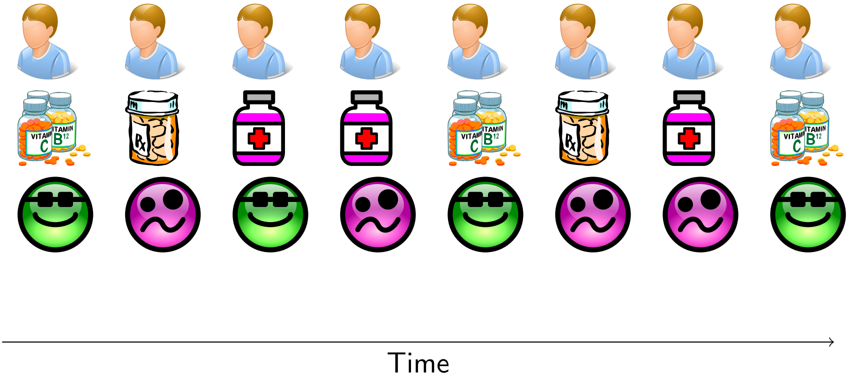
Stochastic Bandit



Model (Unknown)



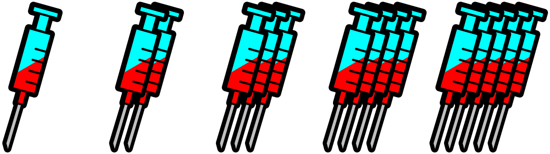
Stochastic Bandit Interaction



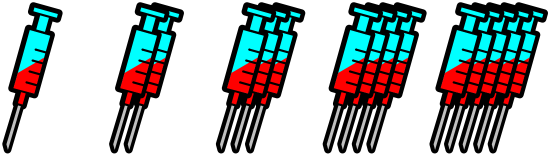
Tasks

1. Pure Exploration: use trial to cure **population**
2. Reward Maximisation: cure **patients in trial**

Structured Stochastic Bandit

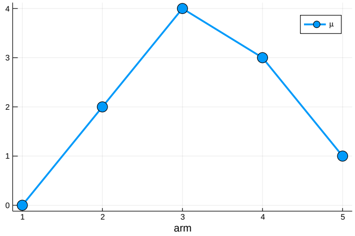


Structured Stochastic Bandit

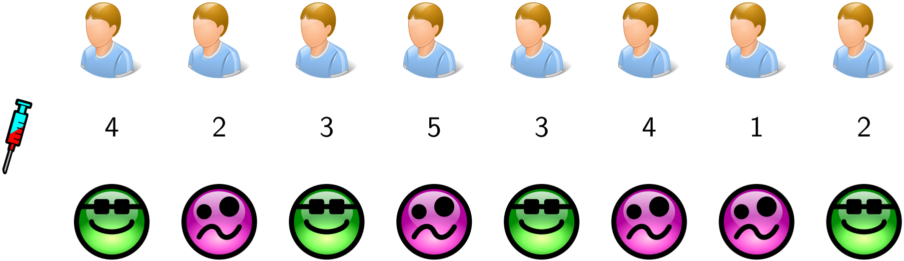


Model (Unknown)

$$\mathbb{P}\left(\;\middle|\; \times 1\right) = 1/6$$
$$\mathbb{P}\left(\;\middle|\; \times 2\right) = 3/6$$
$$\mathbb{P}\left(\;\middle|\; \times 3\right) = 5/6$$
$$\mathbb{P}\left(\;\middle|\; \times 4\right) = 4/6$$
$$\mathbb{P}\left(\;\middle|\; \times 5\right) = 2/6$$



Structured Stochastic Bandit Interaction



What This Tutorial is About

We will develop **efficient learning algorithms** for **Pure Exploration** and **Reward Maximisation**.

Information-theoretic lower bounds will tell us that the complexity of each task is characterised by a certain **two-player zero-sum game**.

We will base our learning algorithms on iterative **saddle point solvers** for this game.

Part II

Pure Exploration / Active Testing

Outline

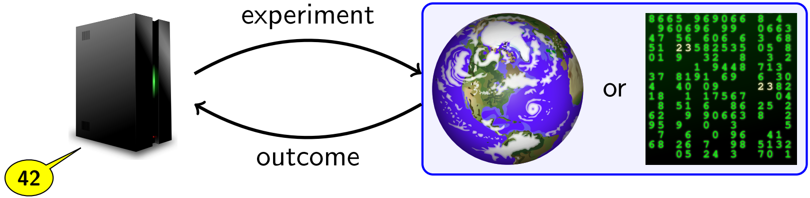
Topic: Pure Exploration

Query:

most effective drug dose?

most appealing website layout?

safest next robot action?



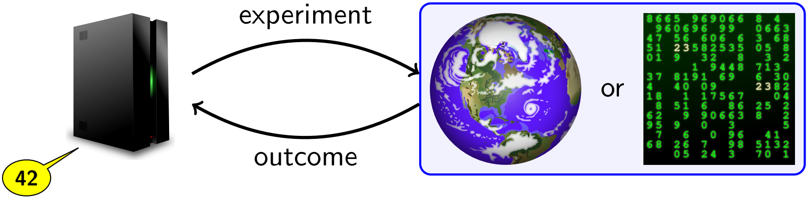
Topic: Pure Exploration

Query:

most effective drug dose?

most appealing website layout?

safest next robot action?



Main scientific questions

- **Efficient** systems
- **Sample complexity** as function of query and environment

Outline

Environments

We fix an 1-d exponential family (Bernoulli, Gaussian, ...) parameterised by the **mean**. KL divergence denoted by $d(\mu, \lambda)$.

Multi-armed bandit model

A K-**armed bandit model** is a tuple $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)$.

Environments

We fix an 1-d exponential family (Bernoulli, Gaussian, ...) parameterised by the **mean**. KL divergence denoted by $d(\mu, \lambda)$.

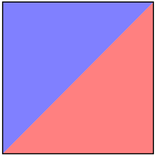
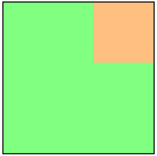
Multi-armed bandit model

A K-**armed bandit model** is a tuple $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)$.

Query

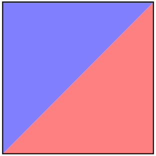
Set of possible environments $\mathcal{M} \subseteq \mathbb{R}^K$. Set of possible answers $\mathcal{I}$.
Correct answer function $i^* : \mathcal{M} \to \mathcal{I}$.

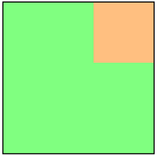
Examples

Problem name	Best Arm	Minimum Threshold
Possible answers $\mathcal{I}$	$[K]$	$\{\text{lo}, \text{hi}\}$
Correct answer $i^*(\boldsymbol{\mu})$	$\text{argmax}_k \, \mu_k$	lo if $\min_k \mu_k < \gamma$ hi if $\min_k \mu_k > \gamma$
		

Examples

Problem name	Best Arm	Minimum Threshold
Possible answers $\mathcal{I}$	$[K]$	$\{\text{lo}, \text{hi}\}$
Correct answer $i^*(\boldsymbol{\mu})$	$\text{argmax}_k \, \mu_k$	lo if $\min_k \mu_k < \gamma$ hi if $\min_k \mu_k > \gamma$





- Top-M
- Combinatorial Best Arm
- Maximum Profit
- Unit Ball
- Thresholding Bandit
- Pure Nash equilibrium
- Game Tree Search
- ...

Strategy for Learner

Strategy

- **Stopping rule** $\tau \in \mathbb{N}$
- In round $t \leq \tau$ **sampling rule** picks $A_t \in [K]$. **See $X_t \sim \mu_{A_t}$.**
- **Recommendation rule** $\hat{I} \in \mathcal{I}$.

Strategy for Learner

Strategy

- **Stopping rule** $\tau \in \mathbb{N}$
- In round $t \leq \tau$ **sampling rule** picks $A_t \in [K]$. **See** $X_t \sim \mu_{A_t}$**.**
- **Recommendation rule** $\hat{I} \in \mathcal{I}$.

Realisation of interaction: $\mathcal{H} \coloneqq \left(A_1, X_1, \ldots, A_\tau, X_\tau, \hat{I}\right)$.

Strategy for Learner

Strategy

- **Stopping rule** $\tau \in \mathbb{N}$
- In round $t \leq \tau$ **sampling rule** picks $A_t \in [K]$. **See $X_t \sim \mu_{A_t}$.**
- **Recommendation rule** $\hat{I} \in \mathcal{I}$.

Realisation of interaction: $\mathcal{H} := \left(A_1, X_1, \ldots, A_\tau, X_\tau, \hat{I}\right)$.

Two objectives: **sample efficiency** τ and **correctness** $\hat{I} = i^*(\boldsymbol{\mu})$.

Goal: PAC learning

Definition

Fix small confidence $\delta \in (0, 1)$. A strategy is δ-**correct** if

$$\mathbb{P}_{\boldsymbol{\mu}}\big(\hat{I} \neq i^*(\boldsymbol{\mu})\big) \ \leq \ \delta \qquad \text{for every bandit model } \boldsymbol{\mu} \in \mathcal{M}.$$

Goal: PAC learning

Definition

Fix small confidence $\delta \in (0,1)$. A strategy is δ-**correct** if

$$\mathbb{P}_{\boldsymbol{\mu}}\big(\hat{I} \neq i^*(\boldsymbol{\mu})\big) \;\leq\; \delta \qquad \text{for every bandit model } \boldsymbol{\mu} \in \mathcal{M}.$$

Goal: minimise sample complexity $\mathbb{E}_{\boldsymbol{\mu}}[\tau]$ over **all δ-correct strategies**.

Goal: PAC learning

Definition

Fix small confidence $\delta \in (0,1)$. A strategy is δ-**correct** if

$$\mathbb{P}_{\boldsymbol{\mu}}\big(\hat{I} \neq i^*(\boldsymbol{\mu})\big) \;\leq\; \delta \qquad \text{for every bandit model } \boldsymbol{\mu} \in \mathcal{M}.$$

Goal: minimise sample complexity $\mathbb{E}_{\boldsymbol{\mu}}[\tau]$ over **all δ-correct strategies**.

Not in this talk: Fixed Budget.

Outline

Instance-Dependent Sample Complexity Lower Bound

Intuition (going back at least to Lai and Robbins [1985]): if observations are likely under both $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$, yet $i^*(\boldsymbol{\mu}) \neq i^*(\boldsymbol{\lambda})$, then learner cannot stop and be correct in both.

Instance-Dependent Sample Complexity Lower Bound

Intuition (going back at least to Lai and Robbins [1985]): if observations are likely under both $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$, yet $i^*(\boldsymbol{\mu}) \neq i^*(\boldsymbol{\lambda})$, then learner cannot stop and be correct in both.

Define the **alternative** to answer $i \in \mathcal{I}$ by $\neg i := \{\boldsymbol{\lambda} \in \mathcal{M} | i^*(\boldsymbol{\lambda}) \neq i\}$.

Instance-Dependent Sample Complexity Lower Bound

Intuition (going back at least to Lai and Robbins [1985]): if observations are likely under both $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$, yet $i^*(\boldsymbol{\mu}) \neq i^*(\boldsymbol{\lambda})$, then learner cannot stop and be correct in both.

Define the **alternative** to answer $i \in \mathcal{I}$ by $\neg i := \{\boldsymbol{\lambda} \in \mathcal{M} | i^*(\boldsymbol{\lambda}) \neq i\}$.

Theorem (Castro 2014, Garivier and Kaufmann 2016)

Fix a δ-correct strategy. Then for every bandit model $\boldsymbol{\mu} \in \mathcal{M}$

$$\mathbb{E}_{\boldsymbol{\mu}}[\tau] \;\geq\; T^*(\boldsymbol{\mu}) \ln \frac{1}{\delta}$$

where the **characteristic time** $T^*(\boldsymbol{\mu})$ *is given by*

$$\frac{1}{T^*(\boldsymbol{\mu})} \;=\; \max_{\boldsymbol{w} \in \triangle_K} \min_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \sum_{i=1}^K w_i d(\mu_i, \lambda_i).$$

Example

Best Arm identification: $i^*(\boldsymbol{\mu}) = \operatorname{argmax}_i \mu_i$.
$K = 5$ Bernoulli arms, $\boldsymbol{\mu} = (0.4, 0.3, 0.2, 0.1, 0.0)$.

$$T^*(\boldsymbol{\mu}) \;=\; 200.4 \qquad \boldsymbol{w}^*(\boldsymbol{\mu}) \;=\; (0.45, 0.46, 0.06, 0.02, 0.01)$$

At $\delta = 0.05$, the time gets multiplied by $\ln \frac{1}{\delta} = 3.0$.

Outline

Pure Exploration Algorithms

Recall, a strategy is defined by

- Stopping rule
- Recommendation rule
- Sampling rule

Stopping and Recommendation Rules

Quantify evidence for $\{i^*(\boldsymbol{\mu}) = i^*(\hat{\boldsymbol{\mu}}_t)\}$ vs alternative $\{i^*(\boldsymbol{\mu}) \neq i^*(\hat{\boldsymbol{\mu}}_t)\}$.

Stopping and Recommendation Rules

Quantify evidence for $\{i^*(\boldsymbol{\mu}) = i^*(\hat{\boldsymbol{\mu}}_t)\}$ vs alternative $\{i^*(\boldsymbol{\mu}) \neq i^*(\hat{\boldsymbol{\mu}}_t)\}$.

Definition

The **extended GLR statistic** is defined as

$$\hat{\Lambda}_t \;=\; \inf_{\boldsymbol{\lambda} \in \neg i^*(\hat{\boldsymbol{\mu}}(t))} \sum_{a=1}^{K} N_a(t) d\left(\hat{\mu}_a(t), \lambda_a\right).$$

Stopping and Recommendation Rules

Quantify evidence for $\{i^*(\boldsymbol{\mu}) = i^*(\hat{\boldsymbol{\mu}}_t)\}$ vs alternative $\{i^*(\boldsymbol{\mu}) \neq i^*(\hat{\boldsymbol{\mu}}_t)\}$.

Definition

The **extended GLR statistic** is defined as

$$\hat{\Lambda}_t \;=\; \inf_{\boldsymbol{\lambda} \in \neg i^*(\hat{\boldsymbol{\mu}}(t))} \sum_{a=1}^{K} N_a(t) d\left(\hat{\mu}_a(t), \lambda_a\right).$$

Proposal: **stop** when

$$\hat{\Lambda}_t \;\geq\; \beta(t, \delta) \;:=\; \ln \frac{1}{\delta} + K \ln \ln t + K \ln \ln \frac{1}{\delta}$$

and **recommend** $\hat{I} = i^*(\hat{\boldsymbol{\mu}}_t)$.

Theorem (Kaufmann and Koolen 2018)

The above stopping and recommendation rules, combined with any sampling rule give a δ-correct algorithm.

Operationalisation of the Oracle Weights

Recall sample complexity lower bound governed by

$$\max_{\boldsymbol{w} \in \triangle_K} \min_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \sum_{i=1}^{K} w_i d(\mu_i, \lambda_i)$$

Operationalisation of the Oracle Weights

Recall sample complexity lower bound governed by

$$\max_{\boldsymbol{w} \in \triangle_K} \min_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \sum_{i=1}^{K} w_i d(\mu_i, \lambda_i)$$

Any matching algorithm **must** sample with optimal (**oracle**) proportions

$$\boldsymbol{w}^*(\boldsymbol{\mu}) = \operatorname*{argmax}_{\boldsymbol{w} \in \triangle_K} \min_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \sum_{i=1}^{K} w_i d(\mu_i, \lambda_i)$$

Putting it all together

Idea: draw $A_t \sim \boldsymbol{w}^*(\hat{\boldsymbol{\mu}}(t))$.

Putting it all together

Idea: draw $A_t \sim \boldsymbol{w}^*(\hat{\boldsymbol{\mu}}(t))$.

Track-and-Stop [Garivier and Kaufmann, 2016]

- Ensure $\hat{\boldsymbol{\mu}}(t) \to \boldsymbol{\mu}$ by **forced exploration**
- **assuming $\boldsymbol{w}^*$ is continuous**, this ensures $\boldsymbol{w}^*(\hat{\boldsymbol{\mu}}_t) \to \boldsymbol{w}^*(\boldsymbol{\mu})$.
- Draw arm with $N_i(t)$ below $\sum_{s=1}^t w_i^*(\hat{\boldsymbol{\mu}}_s)$ (C-tracking)
- hence $N_i(t)/t \to w_i^*(\boldsymbol{\mu})$

Inherit δ-correctness from GLR stopping/recommendation rule.

Putting it all together

Idea: draw $A_t \sim \boldsymbol{w}^*(\hat{\boldsymbol{\mu}}(t))$.

Track-and-Stop [Garivier and Kaufmann, 2016]

- Ensure $\hat{\boldsymbol{\mu}}(t) \to \boldsymbol{\mu}$ by **forced exploration**
- **assuming $\boldsymbol{w}^*$ is continuous**, this ensures $\boldsymbol{w}^*(\hat{\boldsymbol{\mu}}_t) \to \boldsymbol{w}^*(\boldsymbol{\mu})$.
- Draw arm with $N_i(t)$ below $\sum_{s=1}^t w_i^*(\hat{\boldsymbol{\mu}}_s)$ (C-tracking)
- hence $N_i(t)/t \to w_i^*(\boldsymbol{\mu})$

Inherit δ-correctness from GLR stopping/recommendation rule.

Theorem (Degenne et al. 2019)

Track-and-Stop with C-tracking has asymptotically optimal sample complexity.

Putting it all together

Idea: draw $A_t \sim \boldsymbol{w}^*(\hat{\boldsymbol{\mu}}(t))$.

Track-and-Stop [Garivier and Kaufmann, 2016]

- Ensure $\hat{\boldsymbol{\mu}}(t) \to \boldsymbol{\mu}$ by **forced exploration**
- **assuming $\boldsymbol{w}^*$ is continuous**, this ensures $\boldsymbol{w}^*(\hat{\boldsymbol{\mu}}_t) \to \boldsymbol{w}^*(\boldsymbol{\mu})$.
- Draw arm with $N_i(t)$ below $\sum_{s=1}^t w_i^*(\hat{\boldsymbol{\mu}}_s)$ (C-tracking)
- hence $N_i(t)/t \to w_i^*(\boldsymbol{\mu})$

Inherit δ-correctness from GLR stopping/recommendation rule.

Theorem (Degenne et al. 2019)

Track-and-Stop with C-tracking has asymptotically optimal sample complexity.

Theorem (Degenne et al. 2019)

Track-and-Stop with D-tracking may fail to converge.

Outline

Games perspective

Recall TaS based on **plug-in estimate** $\boldsymbol{w}^*(\hat{\boldsymbol{\mu}}_t)$ of **oracle weights**

$$\boldsymbol{w}^*(\boldsymbol{\mu}) = \operatorname*{argmax}_{\boldsymbol{w} \in \triangle_K} \min_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \sum_{i=1}^K w_i d(\mu_i, \lambda_i)$$

Games perspective

Recall TaS based on **plug-in estimate** $\boldsymbol{w}^*(\hat{\boldsymbol{\mu}}_t)$ of **oracle weights**

$$\boldsymbol{w}^*(\boldsymbol{\mu}) = \underset{\boldsymbol{w} \in \triangle_K}{\operatorname{argmax}} \min_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \sum_{i=1}^K w_i d(\mu_i, \lambda_i)$$

We can implement the Track-and-Stop sampling rule by running an online-learning based saddle point solver to (approximate) convergence every round.

Choice of learners: AdaHedge vs Best Response.

Games perspective

Recall TaS based on **plug-in estimate** $\boldsymbol{w}^*(\hat{\boldsymbol{\mu}}_t)$ of **oracle weights**

$$\boldsymbol{w}^*(\boldsymbol{\mu}) \;=\; \operatorname*{argmax}_{\boldsymbol{w} \in \triangle_K} \; \min_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \; \sum_{i=1}^K w_i d(\mu_i, \lambda_i)$$

We can implement the Track-and-Stop sampling rule by running an online-learning based saddle point solver to (approximate) convergence every round.

Choice of learners: AdaHedge vs Best Response.

The user needs to provide **best response oracle** (often tractable)

$$\boldsymbol{w}, \boldsymbol{\mu} \;\mapsto\; \operatorname*{argmin}_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \; \sum_{i=1}^K w_i d(\mu_i, \lambda_i).$$

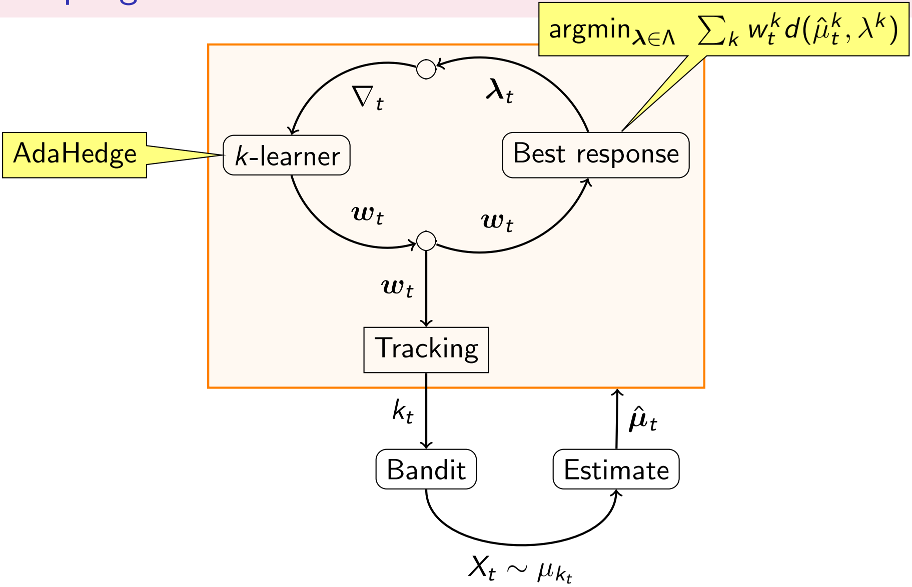
Ironing out Inefficiencies

- We are computing $\boldsymbol{w}^*(\hat{\boldsymbol{\mu}}_t)$ on noisy $\hat{\boldsymbol{\mu}}_t \approx \boldsymbol{\mu}$. So how precise does our saddle point need to be?
- $\hat{\boldsymbol{\mu}}_t \approx \hat{\boldsymbol{\mu}}_{t+1}$. Can we reuse (most) computation?
- Can we get finite confidence guarantees?

Interleaved Iterative Solution

Main idea [Degenne, Koolen, and Ménard, 2019]: advance the saddle point solver by **one** iteration for every bandit interaction.

Sampling Rule



Compositionality

The "overheads" of the ingredients **compose**: Tracking $O(1)$, concentration $\sqrt{T}$, regret $\sqrt{T}$, optimism $\sqrt{T}$.

Theorem (Degenne, Koolen, and Ménard 2019)

The sample complexity is at most

$$\mathbb{E}_{\boldsymbol{\mu}}[\tau] \;\leq\; T^*(\boldsymbol{\mu}) \ln \frac{1}{\delta} + \textit{small}$$

Proof ideas (cheating with optimism, $i_t = i^*$)

As long as we do not stop, $t < \tau$,

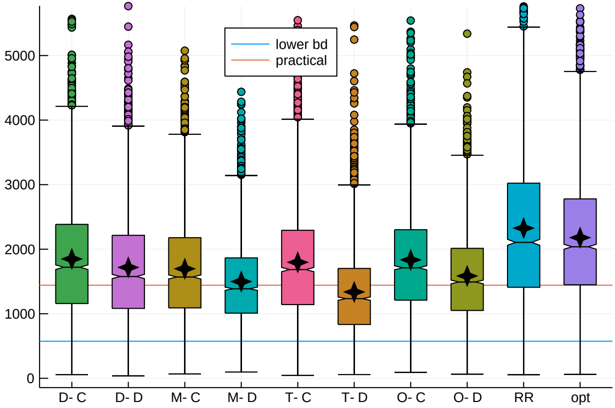
$$
\begin{aligned}
\beta(t,\delta) &\geq \inf_{\boldsymbol{\lambda} \in \neg i_t} \sum_{k=1}^K N_t^k d(\mu^k, \lambda^k) && \text{(stop rule)} \\
&\approx \inf_{\boldsymbol{\lambda} \in \neg i^*} \sum_{s=1}^t \sum_{k=1}^K w_s^k d(\mu^k, \lambda^k) && \text{(tracking)} \\
&\geq \sum_{s=1}^t \sum_{k=1}^K w_s^k \, \mathbb{E}_{\boldsymbol{\lambda} \sim \boldsymbol{q}_s} d(\mu^k, \lambda^k) - R_t^\lambda && \text{(regret } \boldsymbol{\lambda}\text{)} \\
&\geq \max_k \sum_{s=1}^t \mathbb{E}_{\boldsymbol{\lambda} \sim \boldsymbol{q}_s} d(\mu^k, \lambda^k) - R_t^\lambda - R_t^k && \text{(regret } k\text{)} \\
&\geq t \inf_{\boldsymbol{q} \in \mathcal{P}(\neg i^*)} \max_k \mathbb{E}_{\boldsymbol{\lambda} \sim \boldsymbol{q}} d(\mu^k, \lambda^k) - O(\sqrt{t})
\end{aligned}
$$

Find maximal t to get bound on τ.

Outline

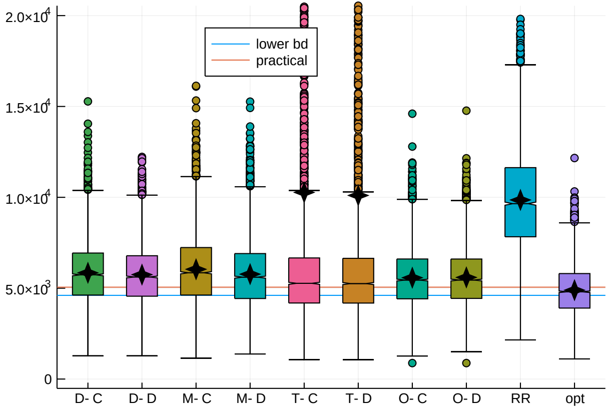
Best Arm Identification Experiment

Best Arm for Bernoulli bandit model $\boldsymbol{\mu} = (0.3, 0.21, 0.2, 0.19, 0.18)$. The oracle weights are $\boldsymbol{w}^* = (0.34, 0.25, 0.18, 0.13, 0.10)$. $\delta = 0.1$.



Minimum Threshold Experiment

Minimum Threshold for Gaussian bandit model $\boldsymbol{\mu} = (0.5, 0.6)$ with threshold $\gamma = 0.6$, $\boldsymbol{w}^* = (1, 0)$. Note the excessive sample complexity of T-C/T-D. $\delta = 10^{-10}$.



Outline

Conclusion

- Pure Exploration currently going through a renaissance
- **New** and **different** instance-optimal identification algorithms
  - Best Arm
  - Combinatorial best action
  - Game Tree Search
  - ...
- Moving toward more complex queries. RL on the horizon ...
- Useful submodules

Topics Skipped

- Multiple correct answers (e.g. ϵ Best Arm) [Degenne and Koolen, 2019] $\Leftarrow$ **surprisingly subtle**.
- Optimal algorithms based on variations of Thompson Sampling
  - Top-Two for Best Arm [Russo, 2016]
  - Murphy Sampling for Minimum Threshold [Kaufmann et al., 2018].

Many questions remain open

- Practically efficient algorithms
- Remove forced exploration
- Moderate confidence $\delta \not\to 0$ regime [Simchowitz et al., 2017].
- Understand sparsity patterns
- Dynamically expanding horizon

Part III

Reward Maximisation / Regret Minimisation

Change of perspective

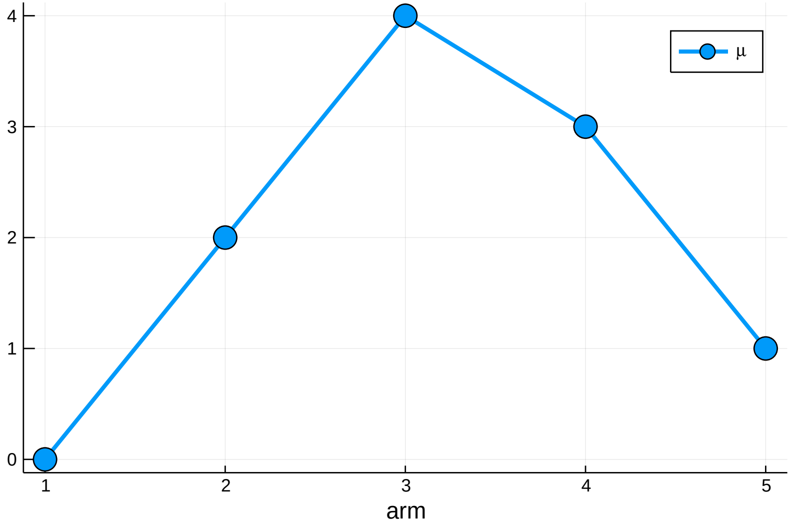
Now samples represent **reward**, and the algorithm aims to collect **as much reward as possible**.

Regret metric: maximum reward minus reward collected.

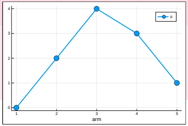
Famous UCB algorithm (family) [Auer et al., 2002].

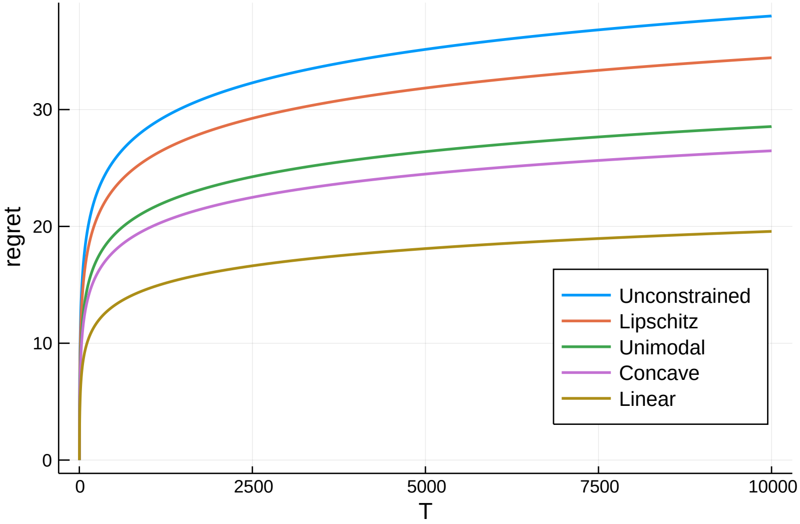
$O(\ln T)$ regret possible.

Stochastic Bandit Instance (Running Example)



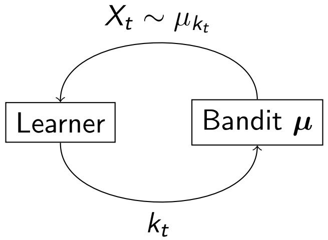
Desired behaviour





Outline

Setting



Structure $\mathcal{M} \subseteq R^K$.
MAB instance $\boldsymbol{\mu} \in \mathcal{M}$
Time horizon T
Expfam $d(\mu, \lambda)$
Gaps $\Delta^k = \mu^* - \mu^k$

$$\text{Regret} := \sum_{k=1}^{K} \mathbb{E}[N_T^k] \Delta^k$$

Goals

- Asymptotic Optimality
- Finite-time Regret Guarantees
- General Structure-Aware Methodology
- Computational Efficiency

Bandital Context

Regret

- Unimodal [Combes and Proutiere, 2014]
- Lipschitz [Magureanu, Combes, and Proutière, 2014]
- Rank-1 [Katariya, Kveton, Szepesvári, Vernade, and Wen, 2017]
- Linear [Lattimore and Szepesvári, 2017]
- OSSB [Combes, Magureanu, and Proutiere, 2017]

Pure Exploration

- Track-and-Stop (MAB) [Garivier and Kaufmann, 2016]
- Structure, Gaussian [Chen, Gupta, Li, Qiao, and Wang, 2017]
- Structure, ExpFam [Kaufmann and Koolen, 2018]
- Game core [Degenne, Koolen, and Ménard, 2019] **part 1**

Outline

Argument [Graves and Lai, 1997]

Fix an **asymptotically consistent** algorithm for structure $\mathcal{M}$. Consider its behaviour on $\boldsymbol{\mu} \in \mathcal{M}$, and on any alternative bandit model $\boldsymbol{\lambda} \in \mathcal{M}$ with $i^*(\boldsymbol{\mu}) \neq i^*(\boldsymbol{\lambda})$:

$$\mathbb{E}_{\boldsymbol{\mu}}\left[N_T^{i^*(\boldsymbol{\mu})}\right]/T \to 1 \qquad \text{but} \qquad \mathbb{E}_{\boldsymbol{\lambda}}\left[N_T^{i^*(\boldsymbol{\mu})}\right]/T \to 0.$$

This stark **difference in behaviour** requires **discriminating information**! Specifically,

$$\mathrm{KL}\left(\mathbb{P}_{\boldsymbol{\mu}}^T \,\middle\|\, \mathbb{P}_{\boldsymbol{\lambda}}^T\right) \;=\; \sum_k \mathbb{E}_{\boldsymbol{\mu}}[N_T^k] d(\mu^k, \lambda^k) \;\geq\; \ln T.$$

Instance-Dependent Regret Lower Bound

Any asymptotically consistent algorithm for structure $\mathcal{M}$ must incur on each $\boldsymbol{\mu} \in \mathcal{M}$ regret at least

$$V_T \;=\; \boxed{\min_{\boldsymbol{N} \geq \boldsymbol{0}} \sum_k N^k \Delta^k \quad \text{subject to} \quad \inf_{\boldsymbol{\lambda} \in \Lambda} \sum_k N^k d(\mu^k, \lambda^k) \geq \ln T}$$

where

$$\Lambda \;=\; \left\{ \boldsymbol{\lambda} \in \mathcal{M} \;\middle|\; i^*(\boldsymbol{\lambda}) \neq i^*(\boldsymbol{\mu}) \right\}$$

This is a (semi-infinite) **covering linear program**.

Operationalising the Lower Bound

Earlier work

At each time step

- compute **oracle sample counts** $\boldsymbol{N}^*(\hat{\boldsymbol{\mu}}_t)$ and advance $\boldsymbol{N}_t \to \boldsymbol{N}^*$, or
- **force exploration** to ensure $\hat{\boldsymbol{\mu}}_t \to \boldsymbol{\mu}$.

Operationalising the Lower Bound

Earlier work

At each time step

- compute **oracle sample counts** $\boldsymbol{N}^*(\hat{\boldsymbol{\mu}}_t)$ and advance $\boldsymbol{N}_t \to \boldsymbol{N}^*$, or
- **force exploration** to ensure $\hat{\boldsymbol{\mu}}_t \to \boldsymbol{\mu}$.

This talk

- Reformat lower bound as zero-sum "minigame".
- **Iteratively** solve minigame by full information online learning.
- Use iterates to advance $\boldsymbol{N}_t$.
- Add optimism to induce exploration.
- **Compose** regret bound from minigame regret + estimation regret

Minigame

We have $V_T = \frac{\ln T}{D^*}$ where

$$D^* = \boxed{\max_{\boldsymbol{w} \in \triangle} \inf_{\boldsymbol{\lambda} \in \Lambda} \frac{\sum_k w^k d(\mu^k, \lambda^k)}{\sum_k w^k \Delta^k}}$$

$w^k \propto N^k$
pulls

Minigame

We have $V_T = \frac{\ln T}{D^*}$ where

$$D^* = \boxed{\max_{\boldsymbol{w} \in \triangle} \inf_{\boldsymbol{\lambda} \in \Lambda} \frac{\sum_k w^k d(\mu^k, \lambda^k)}{\sum_k w^k \Delta^k}}$$

$w^k \propto N^k$
pulls

$$= \boxed{\max_{\tilde{\boldsymbol{w}} \in \triangle} \inf_{\boldsymbol{\lambda} \in \Lambda} \sum_k \tilde{w}^k \frac{d(\mu^k, \lambda^k)}{\Delta^k}}$$

$\tilde{w}^k \propto N^k \Delta^k$
regret

Minigame

We have $V_T = \frac{\ln T}{D^*}$ where

$$D^* = \boxed{\max_{\boldsymbol{w} \in \triangle} \inf_{\boldsymbol{\lambda} \in \Lambda} \frac{\sum_k w^k d(\mu^k, \lambda^k)}{\sum_k w^k \Delta^k}}$$

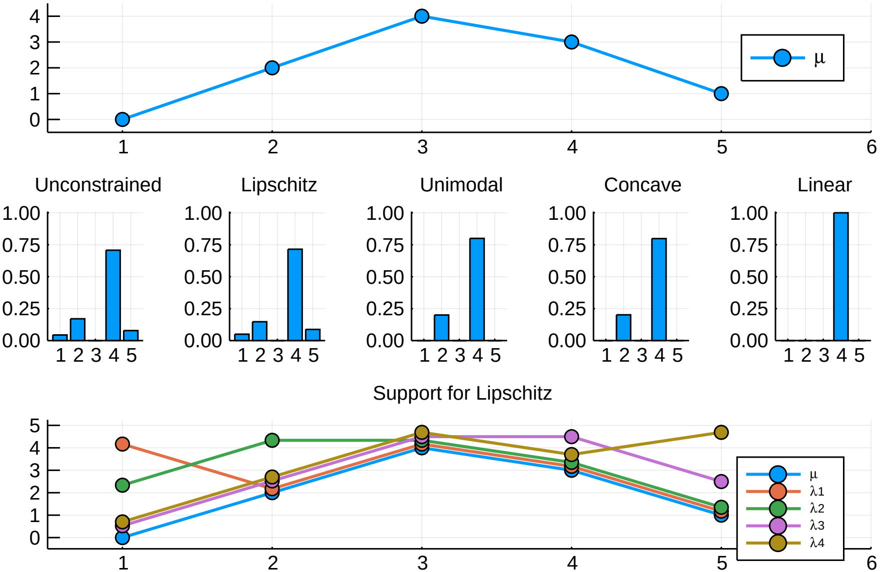
$w^k \propto N^k$
pulls

$$= \boxed{\max_{\tilde{\boldsymbol{w}} \in \triangle} \inf_{\boldsymbol{\lambda} \in \Lambda} \sum_k \tilde{w}^k \frac{d(\mu^k, \lambda^k)}{\Delta^k}}$$

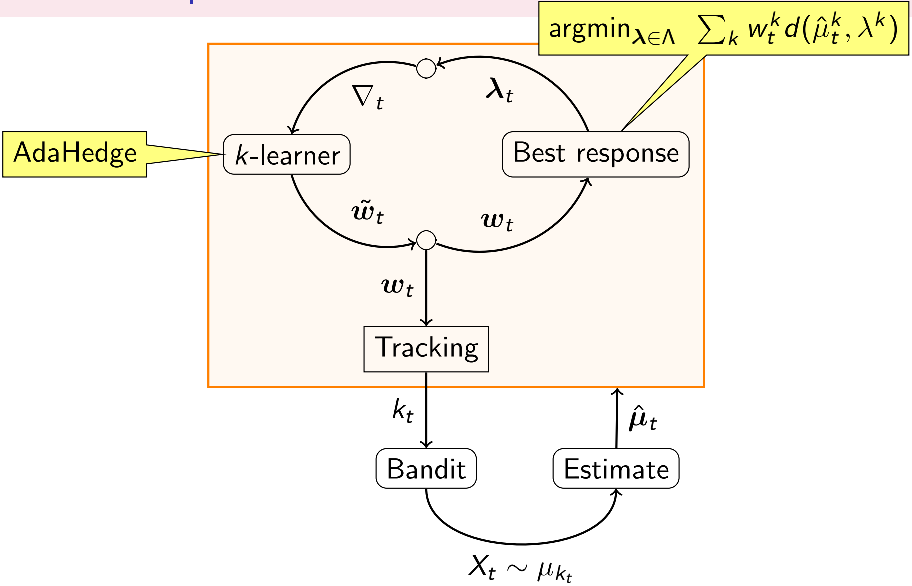
$\tilde{w}^k \propto N^k \Delta^k$
regret

$$= \boxed{\inf_{\boldsymbol{q} \in \triangle(\Lambda)} \max_k \frac{\mathbb{E}_{\boldsymbol{\lambda} \sim \boldsymbol{q}} \left[d(\mu^k, \lambda^k) \right]}{\Delta^k}}$$

Illustration



Overall Setup



Outline

Noise-free result

Let $\mathcal{B}_n^k$ be regret of full information online learning (AdaHedge) w. linear losses on the simplex.

Theorem

Consider running our algorithm until $\inf_{\boldsymbol{\lambda} \in \Lambda} \sum_{t=1}^n \sum_k w_t^k d(\mu^k, \lambda^k) \geq \ln T$. *The iterates* $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n$ *satisfy*

$$R_n \;=\; \sum_{t=1}^n \langle \boldsymbol{w}_t, \Delta \rangle \;\leq\; V_T + \frac{\mathcal{B}_n^k}{D^*}$$

Note

- Can get $k_1, \ldots, k_n$ using tracking (at cost $\Delta^{\max} \ln K$)
- Standard choice gives $n = O(\ln T)$ and $\mathcal{B}_n^k = O(\sqrt{n}) = O(\sqrt{\ln T}) = o(\ln T)$.

Regret analysis

Given moves $\boldsymbol{w}_t \in \triangle_K$ and $\boldsymbol{\lambda}_t \in \Lambda$, we instantiate a k-learner for the gain function

$$g_t(\tilde{\boldsymbol{w}}) \;=\; \langle \boldsymbol{w}_t, \Delta \rangle \sum_k \tilde{w}^k \frac{d(\mu^k, \lambda_t^k)}{\Delta^k}$$

to provide regret bound

$$\sum_{t=1}^n g_t(\tilde{\boldsymbol{w}}_t) \;\geq\; \max_k \sum_{t=1}^n \langle \boldsymbol{w}_t, \Delta \rangle \frac{d(\mu^k, \lambda_t^k)}{\Delta^k} - \mathcal{B}_n^k. \tag{1}$$

Regret analysis (ctd)

Given $\tilde{\boldsymbol{w}}_t$ from the k-learner, we define player and opponent by

$$w_t^k \propto \tilde{w}_t^k / \Delta^k \tag{2}$$

$$\boldsymbol{\lambda}_t \in \operatorname*{argmin}_{\boldsymbol{\lambda} \in \Lambda} \sum_k w_t^k d(\mu^k, \lambda^k) \tag{3}$$

to obtain

$$\begin{aligned} \sum_{t=1}^n g_t(\tilde{\boldsymbol{w}}_t) &= \sum_{t=1}^n \langle \boldsymbol{w}_t, \Delta \rangle \sum_k \tilde{w}_t^k \frac{d(\mu^k, \lambda_t^k)}{\Delta^k} \overset{(2)}{=} \sum_{t=1}^n \sum_k w_t^k d(\mu^k, \lambda_t^k) \\ &\overset{(3)}{=} \sum_{t=1}^n \inf_{\boldsymbol{\lambda} \in \Lambda} \sum_k w_t^k d(\mu^k, \lambda^k) \le \inf_{\boldsymbol{\lambda} \in \Lambda} \sum_{t=1}^n \sum_k w_t^k d(\mu^k, \lambda^k) \end{aligned} \tag{4}$$

Regret analysis (ctd)

The stopping condition plus regret bounds (1) and (4) result in

$$\ln T + \mathcal{B}_n^k \;\geq\; \max_k \sum_{t=1}^n \langle \boldsymbol{w}_t, \Delta \rangle \frac{d(\mu^k, \lambda_t^k)}{\Delta^k} \;=\; R_n \max_k \sum_{t=1}^n \frac{\langle \boldsymbol{w}_t, \Delta \rangle}{R_n} \frac{d(\mu^k, \lambda_t^k)}{\Delta^k}$$

$$\geq\; R_n \inf_{\boldsymbol{q} \in \triangle(\Lambda)} \max_k \frac{\mathbb{E}_{\boldsymbol{\lambda} \sim \boldsymbol{q}} \left[d(\mu^k, \lambda^k) \right]}{\Delta^k} \;=\; R_n D^*$$

where we abbreviated $R_n = \sum_{t=1}^n \langle \boldsymbol{w}_t, \Delta \rangle$. All in all we showed

$$R_n \;\leq\; V_T + \frac{\mathcal{B}_n^k}{D^*}$$

On Symmetry

Game-theoretic equilibrium is **symmetric** concept.

Can also focus on $\boldsymbol{\lambda}$-learner instead of k-learner. Interesting trade-offs

- More complex domain $\boldsymbol{\lambda} \in \Lambda$.
- No need for tracking, best response in k is "pure" arm.

Will show both in experiments.

Outline

Scaling up

Can use what we developed so far to compute oracle weights every round (OSSB). Efficient for **every** bandit structure for which best response is tractable.

Scaling up

Can use what we developed so far to compute oracle weights every round (OSSB). Efficient for **every** bandit structure for which best response is tractable.

But we can do much better!

Scaling up

Can use what we developed so far to compute oracle weights every round (OSSB). Efficient for **every** bandit structure for which best response is tractable.

But we can do much better!
Idea:

- Run only one iteration every round.
- Deal with unknown $\boldsymbol{\mu}$.
- Exploitation.

some issues ...

First Issue

Actually, $\Delta^* = 0$. And we were dividing by it all over the place.

First Issue

Actually, $\Delta^* = 0$. And we were dividing by it all over the place.

Idea: run on $\Delta^k_\epsilon = \max\left\{\Delta^k, \epsilon\right\}$.

Theorem

$$\lim_{\epsilon \to 0} V^\epsilon_T \;=\; V_T$$

In several cases we can show perturbed value is $V^\epsilon_T \leq V_T + \sqrt{2\epsilon V_T}$.

One iteration every round

- Replace $\boldsymbol{\mu}$ by **estimate** $\hat{\boldsymbol{\mu}}_t$.
- Add **optimism** to force exploration.
  We introduce upper confidence bounds on the ratio KL/gap.

$$\mathrm{UCB}_s^k = \sup_{\xi \in \mathcal{C}_{s-1}^k} \frac{d(\xi, \lambda_t^k)}{\max\left\{\epsilon_s, \mathbf{1}\{k \neq j_s\}\left[\mu_{s-1}^+ - \xi\right]\right\}}$$

$$\text{where } \mathcal{C}_{s-1}^k = \left[\hat{\mu}_{s-1}^k \pm \sqrt{\frac{\overline{\ln}(n_{s-1}^{j_s}, N_{s-1}^k)}{N_{s-1}^k}}\right].$$

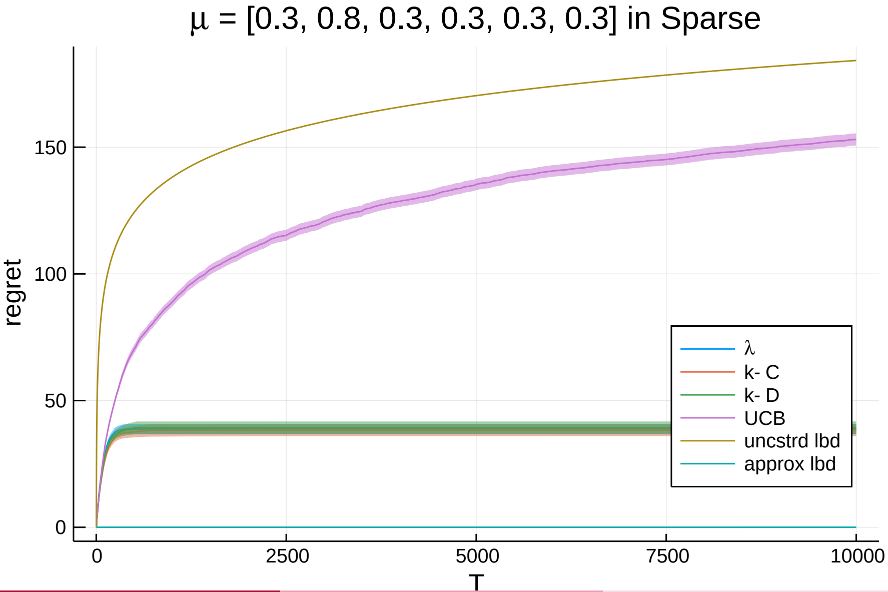
- We do not know **identity of the best arm**, and hence Λ (domain of $\boldsymbol{\lambda}$) Estimate best arm, and run K independent interactions.

Algorithm

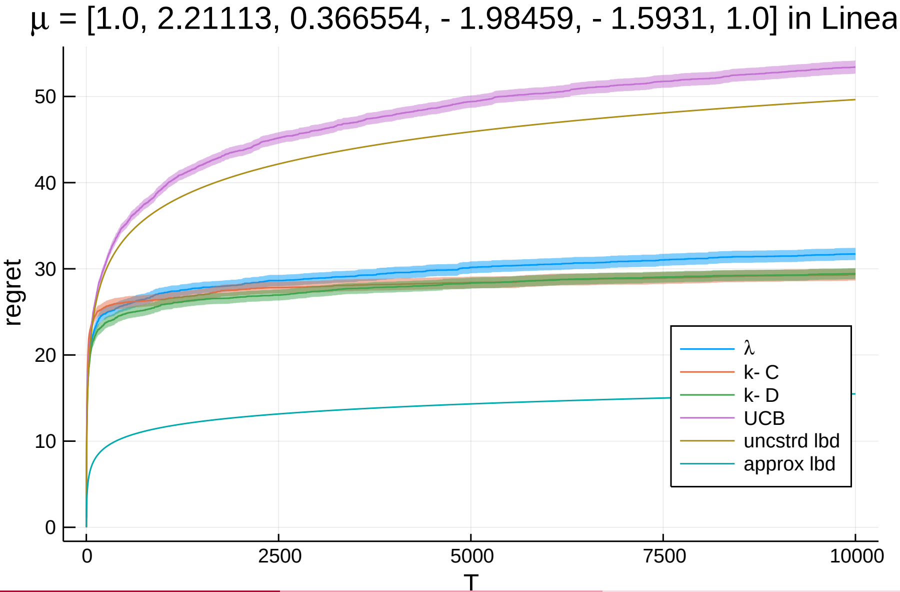
Pull each arm once and get $\hat{\boldsymbol{\mu}}_K$.
for $t = K+1, \cdots, T$ **do**
    if $\exists i \in [K]$, $\min_{\boldsymbol{\lambda} \in \neg i} \sum_k N^k_{t-1} d(\hat{\mu}^k_{t-1}, \lambda^k) > f(t-1)$ **then**
        $k_t = i$ (if there are several suitable i, pull any one of them)
    else
        $\mu^+_{t-1}, j_t = (\arg)\max_{j \in [K]} \hat{\mu}^j_{t-1} + \sqrt{\frac{\overline{\ln}(n^j_{t-1}, N^j_{t-1})}{N^j_{t-1}}}$.
        get $\tilde{\boldsymbol{w}}_t$ from learner $\mathcal{A}^k_{j_t}$, compute $w^k_t \propto \tilde{w}^k_t / \tilde{\Delta}^k$.
        compute best response $\boldsymbol{\lambda}_t$.
        Compute $\mathrm{UCB}^k_t = \max_{\xi \in [\hat{\mu}^k_{t-1} - \ldots, \hat{\mu}^k_{t-1} + \ldots]} \left[\frac{d(\xi, \lambda^k_t)}{\max\{\varepsilon_t, \mathbf{1}\{k \neq j_t\}[\mu^+_{t-1} - \xi]\}} \right]$
        $k_t = \operatorname{argmin}_{k \in [K]} N^k_{t-1} - \sum_{s=1}^t w^k_s$. ▷ Tracking
    end if
    Access $X^{k_t}_t$, update $\hat{\boldsymbol{\mu}}_t$ and $\boldsymbol{N}_t$
end for

Outline

Experiment: Sparse



Experiment: Linear



Conclusion

Game equilibrium based technique for matching **instance dependent lower bounds** for structured stochastic bandits.
All you need is **Best Response oracle**.

- Fine tuning
- What about “lower-order” terms not scaling with $\ln T$?
- Is minigame interaction “easy data”? MetaGrad [van Erven and Koolen, 2016]
- Minigames for other problems?

Conclusion

Game equilibrium based technique for matching **instance dependent lower bounds** for structured stochastic bandits.
All you need is **Best Response oracle**.

- Fine tuning
- What about "lower-order" terms not scaling with $\ln T$?
- Is minigame interaction "easy data"? MetaGrad [van Erven and Koolen, 2016]
- Minigames for other problems?

Thank you!

Outline

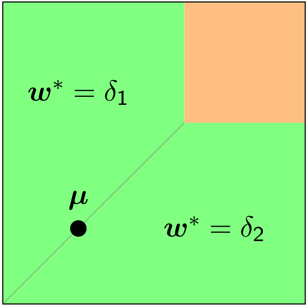
About that continuity assumption?

Can $\boldsymbol{w}^*$ be discontinuous?

About that continuity assumption?

Can $\boldsymbol{w}^*$ be discontinuous?

Example: Minimum Threshold



Continuity restored

Recall oracle weights are given by

$$\boldsymbol{w}^*(\boldsymbol{\mu}) \;=\; \operatorname*{argmax}_{\boldsymbol{w}\in\triangle} \; \inf_{\boldsymbol{\lambda}\in\neg i^*(\boldsymbol{\mu})} \sum_a w_a d(\mu_a, \lambda_a)$$

Continuity restored

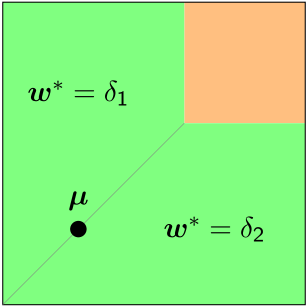
Recall oracle weights are given by

$$\boldsymbol{w}^*(\boldsymbol{\mu}) \; = \; \operatorname*{argmax}_{\boldsymbol{w} \in \triangle} \; \inf_{\boldsymbol{\lambda} \in \neg i^*(\boldsymbol{\mu})} \sum_a w_a d(\mu_a, \lambda_a)$$

Theorem

$\boldsymbol{w}^*$, *when viewed as a* **set-valued** *function, is upper hemicontinuous. Moreover, its output is always a convex set.*

Intuition



On bandit model μ, our empirical distribution will be a convex combination of δ_1 and δ_2.

Outline

KL Tensorises

Algorithm is common and observations are IID. Hence

$$\frac{P_{\boldsymbol{\mu}}\left(I^T, X^T\right)}{P_{\boldsymbol{\lambda}}\left(I^T, X^T\right)} = \prod_{t=1}^{T} \frac{P_{Alg}\left(I_t \middle| I^{t-1}, X^{t-1}\right) \mu_{I_t}(X_t)}{P_{Alg}\left(I_t \middle| I^{t-1}, X^{t-1}\right) \lambda_{I_t}(X_t)} = \prod_{t=1}^{T} \frac{\mu_{I_t}(X_t)}{\lambda_{I_t}(X_t)}$$

It follows that

$$\begin{aligned}
\mathrm{KL}\big(P_{\boldsymbol{\mu}}(I^T, X^T) \big\| P_{\boldsymbol{\lambda}}(I^T, X^T)\big) &= \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{\mu}}\left[\ln \frac{\mu_{I_t}(X_t)}{\lambda_{I_t}(X_t)}\right] \\
&= \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{\mu}}\left[d(\mu_{I_t}, \lambda_{I_t})\right] \\
&= \sum_{i=1}^{K} \mathbb{E}_{\boldsymbol{\mu}}\left[N_{i,T}\right] d(\mu_i, \lambda_i)
\end{aligned}$$