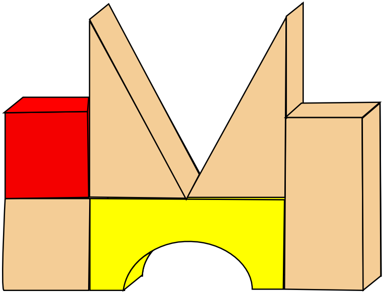
Matching Regret Lower Bounds in Structured Stochastic Bandits



Wouter M. Koolen

CWI

Centrum Wiskunde & Informatica

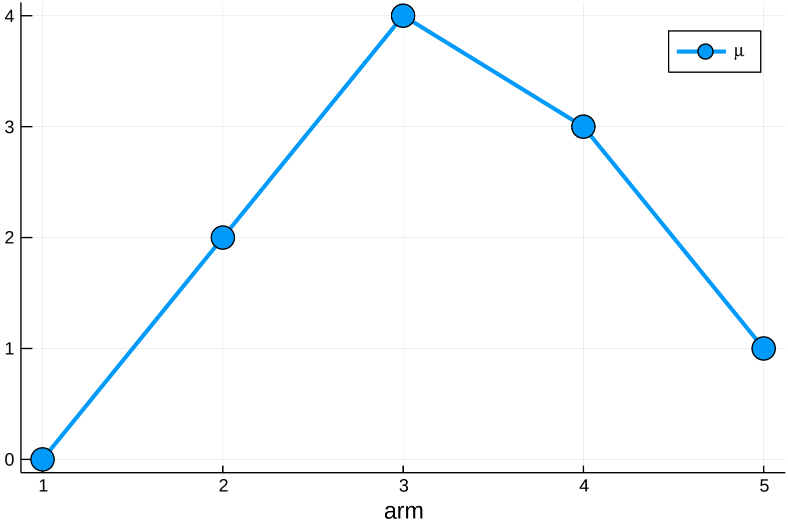
Team

Rémy Degenne

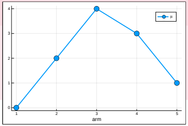
Han Shao (邵涵)

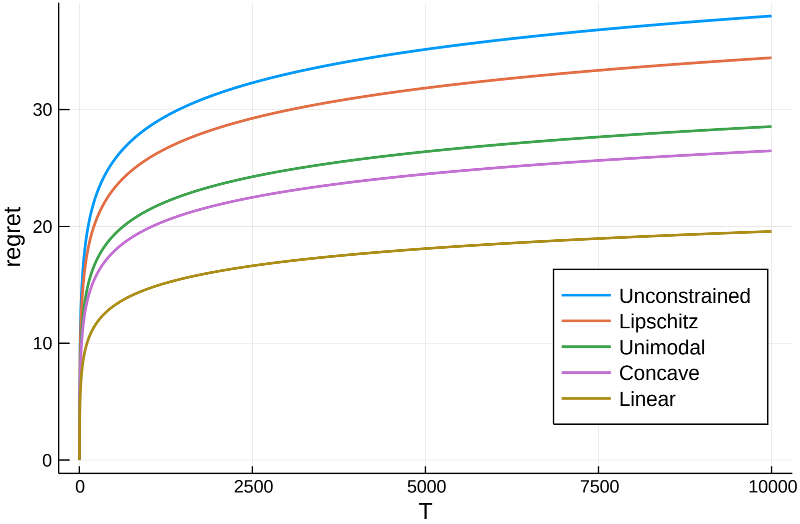
Wouter Koolen

Stochastic Bandit Instance (Running Example)



Desired behaviour





Outline

1 Introduction

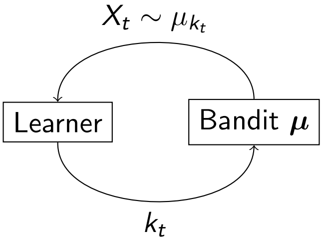
2 Lower bound

3 Noise Free Case

4 The Real Deal

5 Experiments

Setting



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

$$\sum_{t=1}^{T} \mathbb{E}[\Delta^{k_t}]$$

Goals

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

Banditual 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] **yesterday**

Outline

1. Introduction
2. Lower bound
3. Noise Free Case
4. The Real Deal
5. Experiments

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

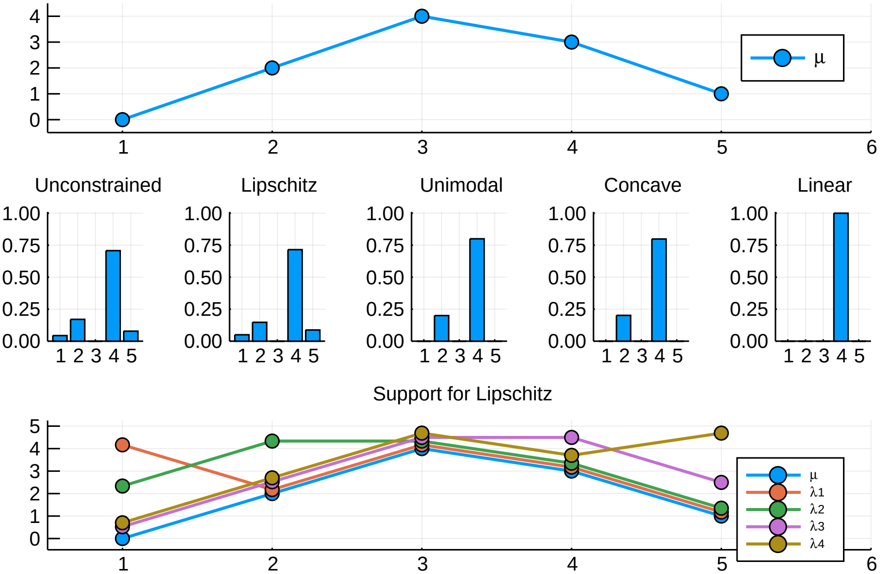
We have $V_T = \frac{\ln T}{D^*}$ where

$$
\begin{aligned}
D^* &= \boxed{\max_{\boldsymbol{w} \in \triangle} \inf_{\boldsymbol{\lambda} \in \Lambda} \frac{\sum_k w^k d(\mu^k, \lambda^k)}{\sum_k w^k \Delta^k}} \\
&= \boxed{\max_{\tilde{\boldsymbol{w}} \in \triangle} \inf_{\boldsymbol{\lambda} \in \Lambda} \sum_k \tilde{w}^k \frac{d(\mu^k, \lambda^k)}{\Delta^k}} \\
&= \boxed{\inf_{\boldsymbol{q} \in \triangle(\Lambda)} \max_k \frac{\mathbb{E}_{\boldsymbol{\lambda} \sim \boldsymbol{q}} \left[d(\mu^k, \lambda^k) \right]}{\Delta^k}}
\end{aligned}
$$

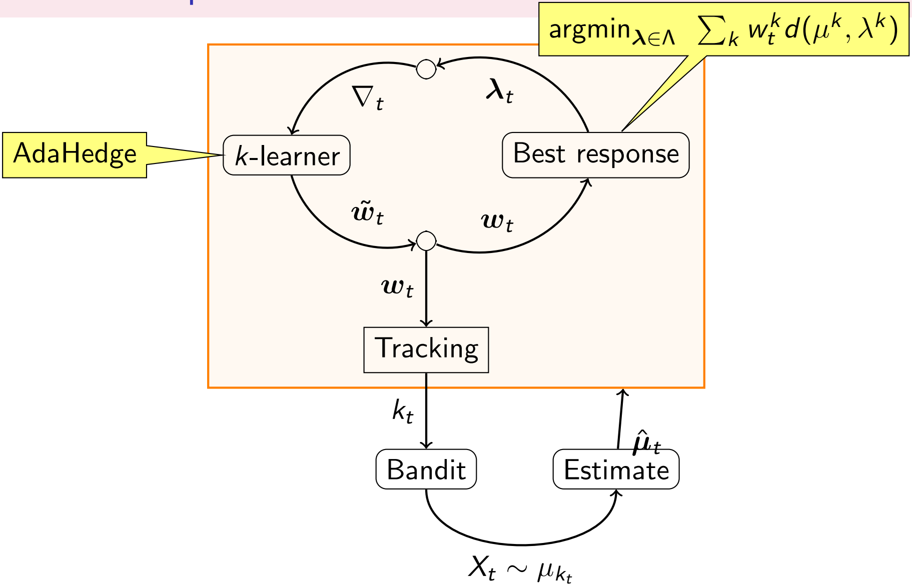
$w^k \propto N^k$
pulls

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

Illustration



Overall Setup



Outline

1 Introduction

2 Lower bound

3 Noise Free Case

4 The Real Deal

5 Experiments

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

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

Regret analysis (ctd)

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

$$\begin{aligned}
\ln T + \mathcal{B}_n^k \;&\geq\; \max_k \sum_{t=1}^n \langle \boldsymbol{w}_t, \Delta \rangle \frac{d(\mu^k, \lambda_t^k)}{\Delta^k} \;=\; R_n \max_k \sum_{t=1}^n \frac{\langle \boldsymbol{w}_t, \Delta \rangle}{R_n} \frac{d(\mu^k, \lambda_t^k)}{\Delta^k} \\
&\geq\; R_n \inf_{\boldsymbol{q} \in \triangle(\Lambda)} \max_k \frac{\mathbb{E}_{\boldsymbol{\lambda} \sim \boldsymbol{q}} \left[d(\mu^k, \lambda^k) \right]}{\Delta^k} \;=\; R_n D^*
\end{aligned}$$

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

1 Introduction

2 Lower bound

3 Noise Free Case

4 The Real Deal

5 Experiments

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 $\Delta_\epsilon^k = \max\left\{\Delta^k, \epsilon\right\}$.

Theorem

$$\lim_{\epsilon \to 0} V_T^\epsilon \ = \ V_T$$

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

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.

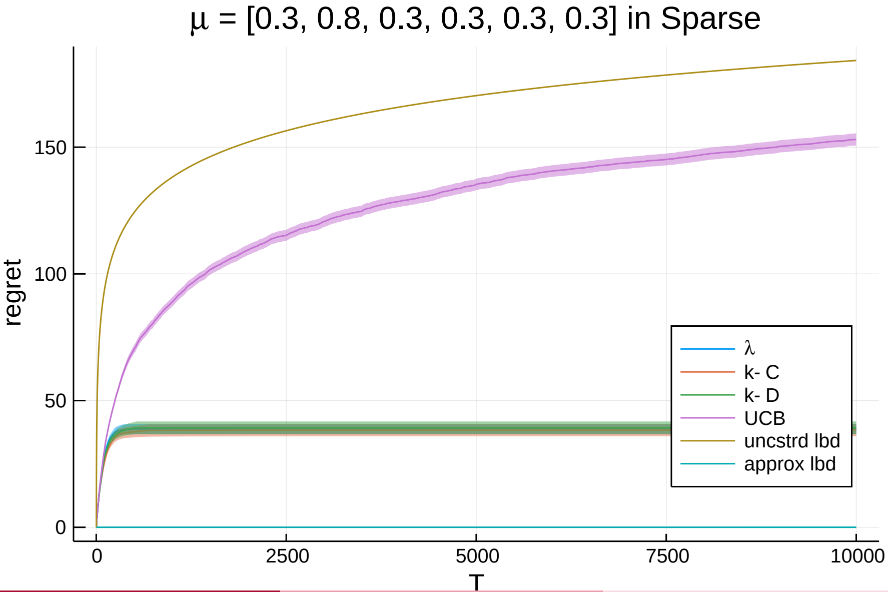
Algorithm

```
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
```

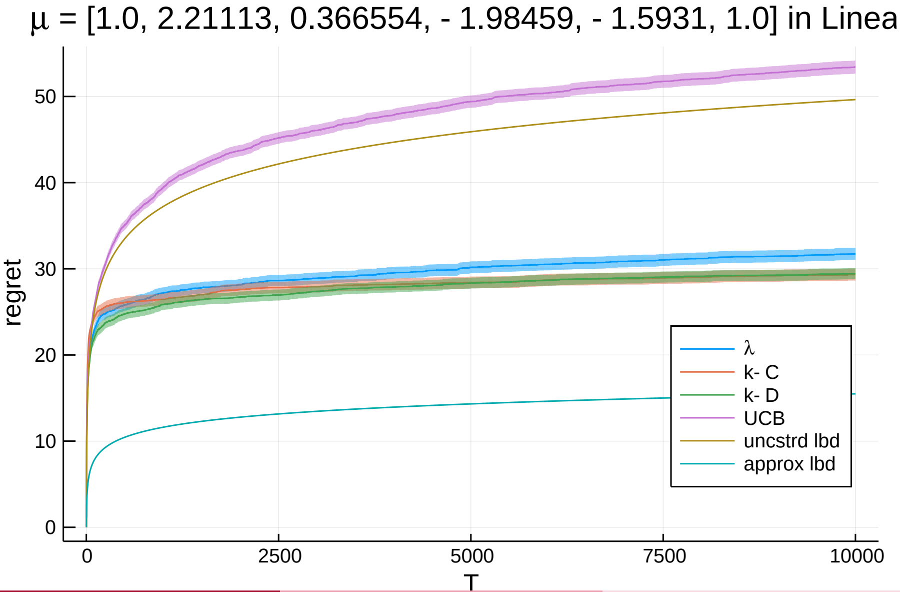
Outline

1. Introduction
2. Lower bound
3. Noise Free Case
4. The Real Deal
5. Experiments

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!