Adaptive Discretization for Adversarial Lipschitz Bandits

Chara Podimata (Harvard), and Aleksandrs Slivkins (MSR-NYC)

COLT 2021

Full version: SCAN ME

Adversarial Lipschitz Bandits

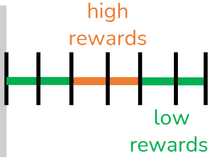
- Known problem instance (A, D), arms $A = [0,1]^d$. (extends to arbitrary metric spaces)
- Rewards $\{ g_t(\cdot) \}_{t\in[T]}$ chosen adversarially, Lipschitz in expectation: $|\mathbb{E}[g_t(x)] - \mathbb{E}[g_t(y)]| \le D(x,y)$
- At round t, learner picks x_t & observes $g_t(x_t)$.
- **Regret** $R(T) = \max_{x^\star \in A} \sum_{t\in[T]} g_t(x^\star) - \sum_{t\in[T]} g_t(x_t)$
- Worst-case optimal regret: $R(T) \le \tilde{O}\left(T^{(d+1)/(d+2)}\right)$

? Improve for "nice" instances, perform optimally in the worst case

Background: IID rewards

Uniform Discretization [Kleinberg NIPS04]

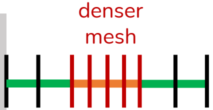
- Create ϵ-net for arms: $K = \epsilon^{-d}$ arms
- Apply any K-MAB algorithm
- Lipschitz → info for all arms
- $R(T) \le \tilde{O}\left(T^{\frac{d+1}{d+2}}\right)$



→ works for adversarial rewards!

Adaptive Discretization (a.k.a. Zooming) [Kleinberg, Slivkins, Upfal STOC08] [Bubeck, Munos, Stoltz, Szepesvari NIPS08]

- "Zoom in" on regions with better payoffs
- $R(T) \le \tilde{O}\left(T^{(z+1)/(z+2)}\right)$, $z = ZoomDim \le d$ captures "nice" instances



breaks for adversarial rewards!

Main Result: Adversarial Zooming

$$R(T) = \tilde{O}\left(T^{(z+1)/(z+2)}\right),$$
z = "Adversarial Zooming Dimension" $\le d$

1) Worst-case optimal, improves for "nice" instances
2) Matches prior work for IID rewards: $z \approx ZoomDim$
3) 1-sided Lipschitzness suffices ⇒ dynamic pricing

$$\text{“}dimension\text{”} = \inf\{d' \ge 0 : A_\epsilon \text{ can be covered with } \gamma \cdot \epsilon^{-d'} \text{ sets of diameter} \le \epsilon, \forall \epsilon > 0\}$$

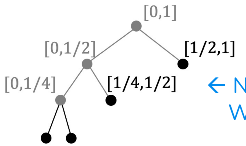
- Covering dimension: $A_\epsilon = A$
- $ZoomDim$ for **IID rewards**: $A_\epsilon = \{\text{arms with gap} \le \epsilon\}$ where $\mathrm{Gap}(x) \coloneqq \max_{y\in A} \mathbb{E}[g_t(y)] - \mathbb{E}[g_t(x)]$

Adversarial: $AdvGap_t(x) \coloneqq \frac{1}{t} \max_{y\in A} \sum_{\tau\in[t]} g_\tau(y) - g_\tau(x)$

A_ϵ: arms x: $AdvGap_t(x) < \tilde{O}(\epsilon)$ for some time $t > \Omega(\epsilon^{-2})$

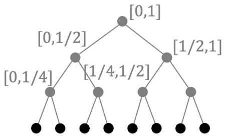
What it Means to Zoom-In

1. Maintain a tree: a hierarchical partition of the arms' space
2. Start with one node: the whole space.
3. In each round, choose a node ~ probability distribution.
4. Zoom in on a node iff enough confidence about its reward
5. Then: parent node de-activated, children "activated", parent's "weight" split equally among children



← Nice instance

Worst-case →



Roadblocks

Issue	IID	Adversarial
small "confidence interval" implies small "gap"	Easy	Breaks
Bound the total regret from arms with very small "gap"	Easy	Breaks
Key steps for K-arm bandit analysis hold for variable K	Easy	Hard
Bound parent's influence	No need	Must
Node's "exploration amount" is expressed as	#samples	total prob. mass
Confidence interval directly uses "exploration amount"	Yes	No

Algorithm: "zooming" & EXP3.P

Zoom in on node u iff its both "confidence terms" are small

- Total conf. term: $\approx \sum_{\tau\in[t]} \beta_\tau / \pi_\tau(act_\tau(u))$,
- instantaneous conf. term: $\approx \beta_t / \pi_t(u)$,

where $\pi_t(u)$ is sampling prob and $\beta_t \sim 1/\sqrt{t}$.

$act_\tau(u)$ = active ancestor node of u at time τ

Complex, multi-stage analysis, see the paper for the outline

Future Directions

1) Mitigate Lipschitz assumptions via "smoothed regret"
2) Extend to more general pricing problems (ongoing work)
3) Extend to Contextual Bandits