Adaptive Discretization for Adversarial Lipschitz Bandits

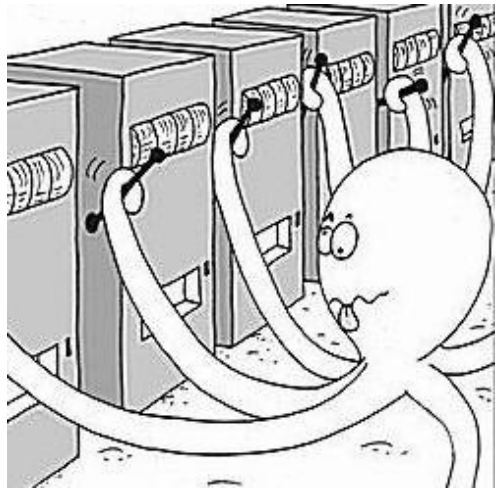
Chara Podimata (Harvard)

Joint work with Aleksandrs Slivkins (Microsoft Research NYC)

COLT2021

Multi-Armed Bandits (MAB)

Bandit feedback



- Fixed set of actions A that you can play for T rounds.
- At each round t ALG picks some arm $x_t \in A$.
- Receives reward $g_t(x_t) \in [0,1]$.
- explore-exploit tradeoff

- IID rewards
- Adversarial rewards

Regret: $R(T) = \max_{x^\star \in A} \sum_{t\in[T]} g_t(x^\star) - \sum_{t\in[T]} g_t(x_t) \le o(T)$

General Setting $R(T) = \tilde{O}\left(\sqrt{TK}\right)$, K=#arms

Well - studied problem with lots of applications

Dynamic Pricing
- arms = prices
- reward = revenue condition on purchase
- seller wants to pick prices to max revenue

Web Ad Placement
- arms = ads
- reward = click through rate
- seller wants to pick ads to max clicks

Bandits with Similarity Information

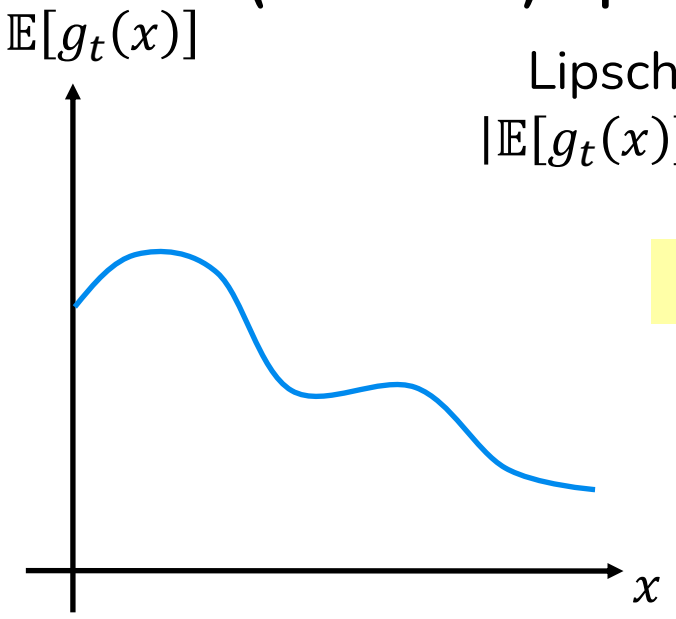
What if set of actions is **super large/infinite**?

$o(T)$ regret impossible without **side info**!

Side Info in Dynamic Pricing
- Numerical similarity between arms.
- Smoothness of revenue function.

Side Info in Web Ad Placement
- Topical taxonomy, feature vectors etc.
- Context: user profile, page content etc.

(Stochastic) Lipschitz Bandits



Lipschitz reward functions

$|\mathbb{E}[g_t(x)] - \mathbb{E}[g_t(y)]| \leq D(x,y)$

similarity metric

* This talk: arms space $A = [0,1]^d$; results generalize to arbitrary metrics

Very well-studied in the literature

[Agrawal, 95], [Kleinberg NIPS14], [Kleinberg Slivkins, Upfal STOC08/JACM19], [Bubeck, Munos, Stoltz, Szepesvari NIPS08/JMLR11], [Slivkins, Radlinski, Gollapudi, ICML10/JMLR13], [Slivkins COLT11/JMLR14], [Munos NIPS11], [Slivkins NIPS11], [Valko, Carpentier, Munos, ICML13], [Minsker COLT13], [Bull, BJS15], [Ho, Slivkins, Vaughan, EC14/JAIR16], [Grill, Valko, Munos NIPS15], [Krishnamurthy, Langford, Slivkins, Zhang COLT19/JMLR20]

Background: Uniform vs. Adaptive Discretization

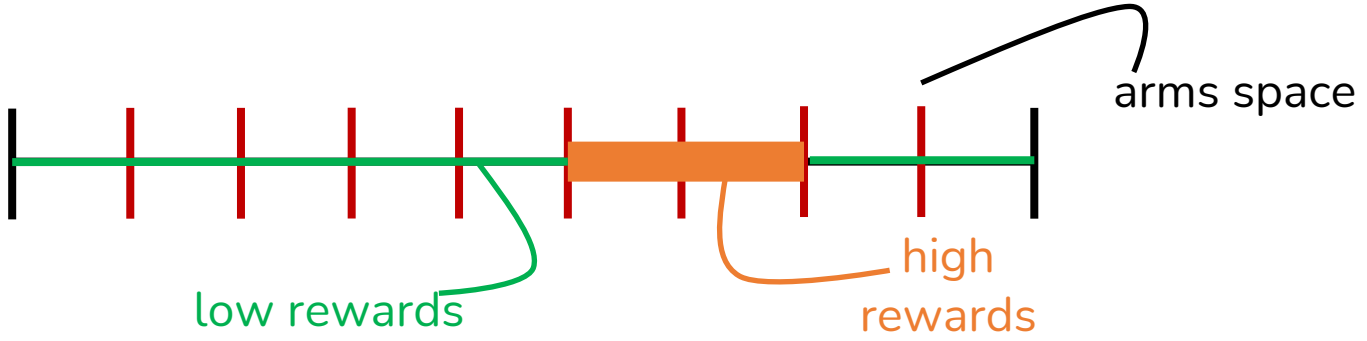
Uniform Discretization

[Kleinberg & Leighton FOCS03], [Kleinberg NIPS14]

- Create ϵ-net for arms: $K = \epsilon^{-d}$ arms
- Apply any K-MAB algo — Discretization
- Lipschitz → info for all arms — Error

$$R(T) \leq R_{MAB}(T) + DE(X)$$
$$\leq O\left(\sqrt{T\epsilon^{-d}\log(\epsilon^{-d})}\right) + L \cdot \epsilon \cdot T \leq \tilde{O}\left(T^{\frac{d+1}{d+2}}\right)$$

Worst-case regret for Lipschitz MAB: $R(T) = \tilde{O}\left(T^{\frac{d+1}{d+2}}\right)$

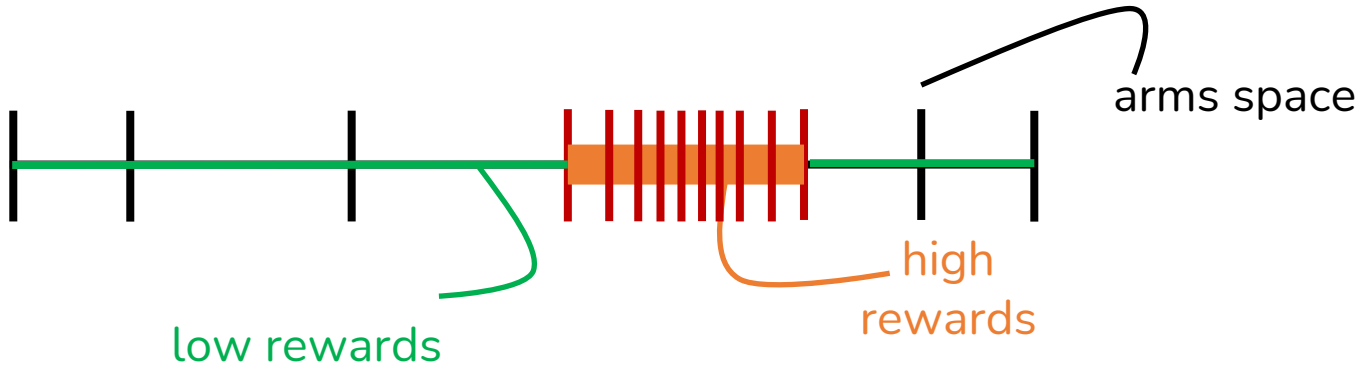


Easy to implement + optimal for worst-case instances, wasteful for benign ones.

Adaptive Discretization

[Kleinberg, Slivkins, Upfal STOC08] [Bubeck, Munos, Stoltz, Szepesvari: NIPS08]

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



Our Focus: Adversarial Lipschitz Bandits

Model

- Known problem instance (A, D), arms $A = [0,1]^d$.
- Adversary picks sequence of $\{ g_t(\cdot) \}_{t \in [T]}$

  adversarially, Lipschitz in expectation.

$$|\mathbb{E}[g_t(x)] - \mathbb{E}[g_t(y)]| \leq D(x, y)$$

- At round t, learner picks x_t & observes $g_t(x_t)$.

- ✓ Uniform discretization: still worst-case optimal

$$R(T) \leq \widetilde{O}\left(T^{\frac{d+1}{d+2}}\right)$$

- ✓ Adaptive discretization: prior work only for IID

How do we take advantage of "nicer" instances in adversarial Lipschitz bandits, while performing optimally in the worst case?

- New algorithm
- Similar results to IID, with (much) more work & new ideas!

Main Result: Adversarial Zooming Algorithm

Regret $\widetilde{O}\left(T^{\frac{z+1}{z+2}}\right)$, where z = "Adversarial Zooming Dimension" $\leq d$

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

Construction: set of examples for IID rewards & small ZoomDim → examples with adv. rewards & small AdvZoomDim

---

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

Covering dim: $A_\epsilon = A$

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

Adversarial rewards:

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

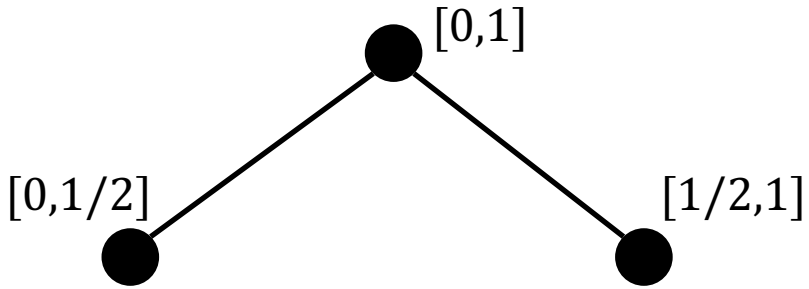
A_ϵ: arms x such that $AdvGap_t(x) < \widetilde{O}(\epsilon)$ for some stopping time $t > \Omega(\epsilon^{-2})$.

What It Means to “Zoom-In”

1. Maintain a hierarchical partition of the space.
2. Start with 1 node = whole space.
3. Choose node ~ probability distribution.
4. Zoom-in on node once you have enough confidence about its reward.

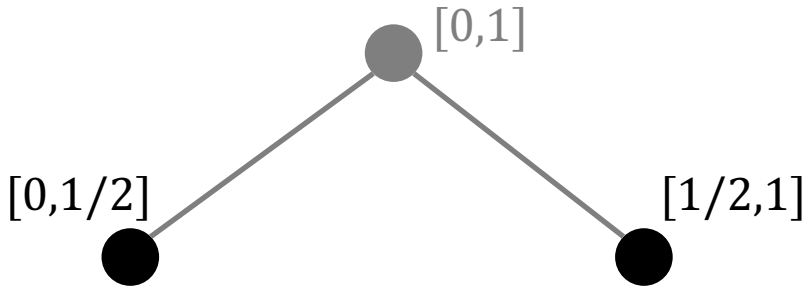
What It Means to “Zoom-In”

1. Maintain a hierarchical partition of the space.
2. Start with 1 node = whole space.
3. Choose node ~ probability distribution.
4. Zoom-in on node once you have enough confidence about its reward.



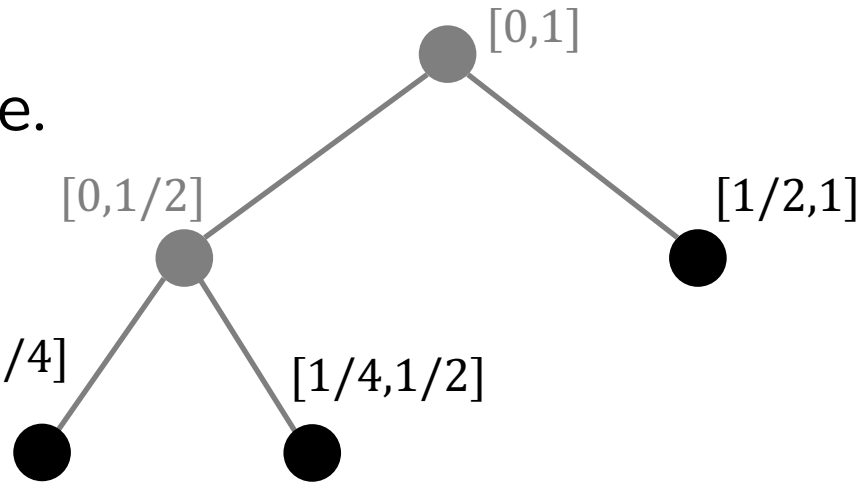
What It Means to “Zoom-In”

1. Maintain a hierarchical partition of the space.
2. Start with 1 node = whole space.
3. Choose node ~ probability distribution.
4. Zoom-in on node once you have enough confidence about its reward.
5. Parent de-activated, children get inherited information, weights.



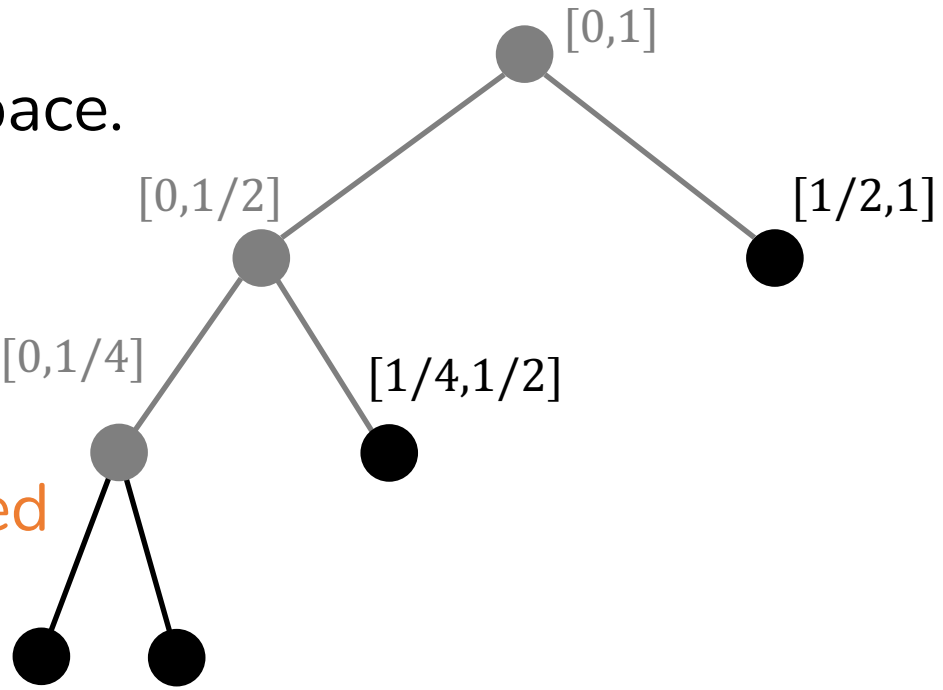
What It Means to “Zoom-In”

1. Maintain a hierarchical partition of the space.
2. Start with 1 node = whole space.
3. Choose node ~ probability distribution.
4. Zoom-in on node once you have enough confidence about its reward.
5. Parent de-activated, children get inherited information, weights.



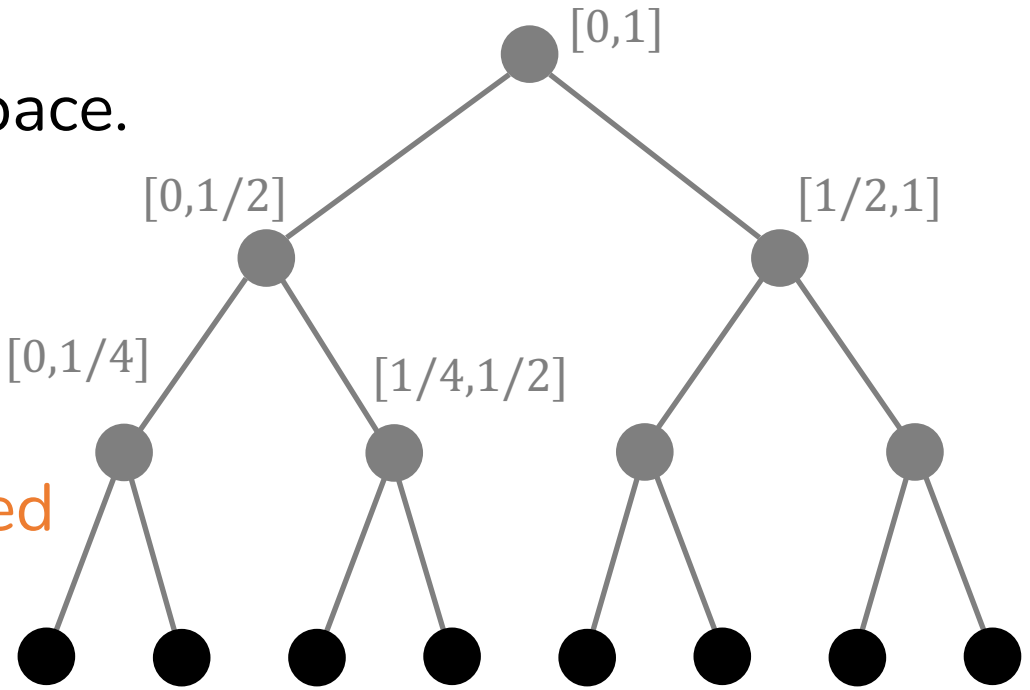
Tree of a Benign Instance

1. Maintain a hierarchical partition of the space.
2. Start with 1 node = whole space.
3. Choose node ~ probability distribution.
4. Zoom-in on node once you have enough confidence about its reward.
5. Parent de-activated, children get inherited information, weights.



Tree of a Worst-Case Instance

1. Maintain a hierarchical partition of the space.
2. Start with 1 node = whole space.
3. Choose node ~ probability distribution.
4. Zoom-in on node once you have enough confidence about its reward.
5. Parent de-activated, children get inherited information, weights.



What It Means to "Zoom-In"

1. Maintain a hierarchical partition of the space.
2. Start with 1 node = whole space.
3. Choose node ~ probability distribution.
4. Zoom-in on node once you have enough confidence about its reward.
5. Parent de-activated, children get inherited information, weights.

Prior work on iid setting
Zoom-in idea + optimism in the face of uncertainty

Adversarial Lipschitz MAB: Zoom-in idea + EXP3

Roadblocks

Issue	IID	Adversarial
When zooming: small confidence term → small gap	Easy	Breaks

Roadblocks

Issue	IID	Adversarial
When zooming: small confidence term → small gap	Easy	Breaks
Bound the total regret from arms with very small gap	Easy	Breaks

Roadblocks

Issue	IID	Adversarial
When zooming: small confidence term → small gap	Easy	Breaks
Bound the total regret from arms with very small gap	Easy	Breaks
Key steps for K-arm analysis hold for variable K =#arms	Easy	Hard

Roadblocks

Issue	IID	Adversarial
When zooming: small confidence term → small gap	Easy	Breaks
Bound the total regret from arms with very small gap	Easy	Breaks
Key steps for K-arm analysis hold for variable K =#arms	Easy	Hard
Bounding the parent's influence on the child	No need	Must

Roadblocks

Issue	IID	Adversarial
When zooming: small confidence term → small gap	Easy	Breaks
Bound the total regret from arms with very small gap	Easy	Breaks
Key steps for K-arm analysis hold for variable K =#arms	Easy	Hard
Bounding the parent's influence on the child	No need	Must
"exploration statistic" for a given arm	#samples	total prob. mass

Roadblocks

Issue	IID	Adversarial
When zooming: small confidence term → small gap	Easy	Breaks
Bound the total regret from arms with very small gap	Easy	Breaks
Key steps for K-arm analysis hold for variable K =#arms	Easy	Hard
Bounding the parent's influence on the child	No need	Must
"exploration statistic" for a given arm	#samples	total prob. mass
Confidence radius directly uses "exploration statistic"	Yes	No

Adversarial Zooming

Parameters: $\beta_t, \gamma_t, \eta_t \in (0, ½]$ $\forall t$

Variables: active nodes A_t, weights $w_{t,\eta}$

For all rounds $t = 1, \ldots, T$:

1. Sample tree node $U_t \sim \pi_t(\cdot)$, where
   $$\pi_t(\cdot) \leftarrow (1-\gamma_t)p_t(\cdot) + \frac{\gamma_t}{|A_t|} \text{ and } p_t \propto w_{t,\eta_t}.$$
2. Play default arm x_t for U_t, observe reward $g_t(x_t)$.

For all active nodes $u \in A_t$:

3. Estimator (= IPS + "conf term"):
   $$\hat{g}_t(u) = \frac{g_t(x_t)\cdot 1\{u=U_t\}}{\pi_t(u)} + \frac{(1+4\log T)\cdot\beta_t}{\pi_t(u)} .$$
4. MW update:
   $$w_{t+1,\eta}(u) = w_{t,\eta}(u) \cdot \exp(\eta \cdot \hat{g}_t(u)).$$

If both confidence terms are small:

5. Activate children, deactivate parent:
   $$A_{t+1} \leftarrow A_t \cup Children(u) \setminus \{u\}$$
6. Split parent's weight among children v:
   $$w_{t+1}(v) = w_{t+1}(u)/|Children(u)|$$

Total conf term: $\approx \sum_{\tau\in[t]} \beta_\tau/\pi_\tau(act_\tau(u))$,
$act_\tau(u)$ = active ancestor of u at round τ
Instantaneous conf term: $\approx \beta_t/\pi_t(u)$

Adversarial Zooming

Parameters: $\beta_t, \gamma_t, \eta_t \in (0, ½]$ $\forall t$

Variables: active nodes A_t, weights $w_{t,\eta}$

For all rounds $t = 1, \ldots, T$:

1. Sample tree node $U_t \sim \pi_t(\cdot)$, where
$$\pi_t(\cdot) \leftarrow (1-\gamma_t)p_t(\cdot) + \frac{\gamma_t}{|A_t|} \text{ and } p_t \propto w_{t,\eta_t}.$$
2. Play default arm x_t for U_t, observe reward $g_t(x_t)$.

For all active nodes $u \in A_t$:

3. Estimator (= IPS + "conf term"):
$$\hat{g}_t(u) = \frac{g_t(x_t)\cdot 1\{u=U_t\}}{\pi_t(u)} + \frac{(1+4\log T)\cdot\beta_t}{\pi_t(u)} .$$
4. MW update:
$$w_{t+1,\eta}(u) = w_{t,\eta}(u) \cdot \exp(\eta \cdot \hat{g}_t(u)).$$

If both confidence terms are small:

5. Activate children, deactivate parent:
$$A_{t+1} \leftarrow A_t \cup Children(u) \setminus \{u\}$$
6. Split parent's weight among children v:
$$w_{t+1}(v) = w_{t+1}(u)/|Children(u)|$$

Analysis I: Zooming Rule

1) Zooming Invariant for all active arms
2) Lifespan of node u bounded: $\propto 1/L(u)$
3) Node's own data drown out inherited ones
4) When zoomed in, total prob. mass of node u is large: $\geq L^{-2}(u)$.
5) When zoomed in, current prob of node u is large.

Total conf term: $\approx \sum_{\tau\in[t]} \beta_\tau/\pi_\tau(act_\tau(u))$,
$act_\tau(u)$ = active ancestor of u at round τ
Instantaneous conf term: $\approx \beta_t/\pi_t(u)$

Adversarial Zooming

Parameters: $\beta_t, \gamma_t, \eta_t \in (0, ½]\ \forall t$

Variables: active nodes A_t, weights $w_{t,\eta}$

For all rounds $t = 1, \ldots, T$:

1. Sample tree node $U_t \sim \pi_t(\cdot)$, where
$$\pi_t(\cdot) \leftarrow (1-\gamma_t)p_t(\cdot) + \frac{\gamma_t}{|A_t|} \text{ and } p_t \propto w_{t,\eta_t}.$$
2. Play default arm x_t for U_t, observe reward $g_t(x_t)$.

For all active nodes $u \in A_t$:

3. Estimator (= IPS + "conf term"):
$$\hat{g}_t(u) = \frac{g_t(x_t)\cdot 1\{u=U_t\}}{\pi_t(u)} + \frac{(1+4\log T)\cdot\beta_t}{\pi_t(u)}\ .$$
4. MW update:
$$w_{t+1,\eta}(u) = w_{t,\eta}(u)\cdot \exp(\eta\cdot\hat{g}_t(u)).$$

If both confidence terms are small:

5. Activate children, deactivate parent:
$$A_{t+1} \leftarrow A_t \cup Children(u) \setminus \{u\}$$
6. Split parent's weight among children v:
$$w_{t+1}(v) = w_{t+1}(u)/|Children(u)|$$

Analysis II: Multiplicative Weights

1) Analyzing the weight and bias inheritance:

$$w_{t+1,\eta}(u) = C_{prod}(u)\cdot \exp(\eta \textstyle\sum_{\tau\in[t]} \hat{g}_\tau(act_\tau(u))$$

$\Pi_{v\in Ancestors(u)}\ Children(v)$

active ancestor of u at round τ

2) Changing sets of active arms + changing η_t's → potential function definition:

$$\Phi_t(\eta) = \left(\frac{1}{|A_t|}\sum_{u\in A_t} w_{t+1,\eta}(u)\right)^{1/\eta}$$

3) MW-style analysis

$$Q := \ln\left(\frac{\Phi_T(\eta_T)}{\Phi_0(\eta_0)}\right)$$

$$\leq \sum_{t\in[T]} g_t(x_t) + O(\ln T)\left(\gamma_t + \beta_t + \sum_{u\in A_t}\hat{g}_t(u)\right)$$

Adversarial Zooming

Parameters: $\beta_t, \gamma_t, \eta_t \in (0, ½]\ \forall t$

Variables: active nodes A_t, weights $w_{t,\eta}$

For all rounds $t = 1, \ldots, T$:

1. Sample tree node $U_t \sim \pi_t(\cdot)$, where
$$\pi_t(\cdot) \leftarrow (1-\gamma_t)p_t(\cdot) + \frac{\gamma_t}{|A_t|} \text{ and } p_t \propto w_{t,\eta_t}.$$
2. Play default arm x_t for U_t, observe reward $g_t(x_t)$.

For all active nodes $u \in A_t$:

3. Estimator (= IPS + "conf term"):
$$\hat{g}_t(u) = \frac{g_t(x_t)\cdot 1\{u=U_t\}}{\pi_t(u)} + \frac{(1+4\log T)\cdot\beta_t}{\pi_t(u)}\ .$$
4. MW update:
$$w_{t+1,\eta}(u) = w_{t,\eta}(u)\cdot \exp(\eta\cdot \hat{g}_t(u)).$$

If both confidence terms are small:

5. Activate children, deactivate parent:
$$A_{t+1} \leftarrow A_t \cup Children(u) \setminus \{u\}$$
6. Split parent's weight among children v:
$$w_{t+1}(v) = w_{t+1}(u)/|Children(u)|$$

Analysis III: Estimated Rewards

1) Lipschitzness in expectation
2) IPS concentration
3) Zooming invariant
4) Effect of inherited rewards is drowned out

$$R(T) \le \widetilde{O}\left(\sqrt{dT} + O\left(\frac{1}{\beta_T}\right) + O\left(\frac{\ln|A_T|}{\eta_T}\right) + \sum_{t\in[T]} \beta_t + \gamma_t \ln T\right)$$

Final Computation

1) Worst-case number of active nodes does **not** grow arbitrarily large.
2) Plug in def of $AdvGap_t(x)$.

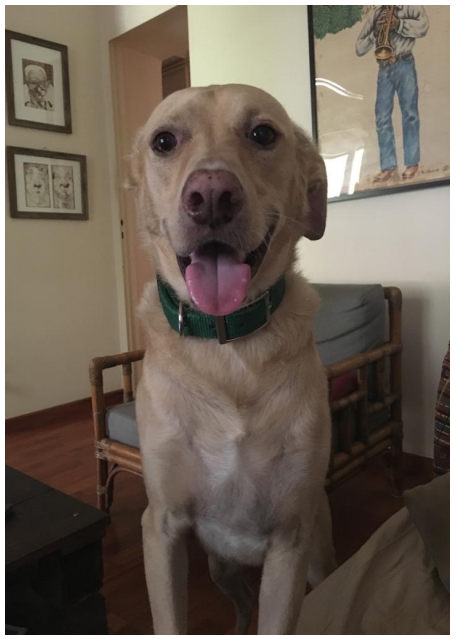
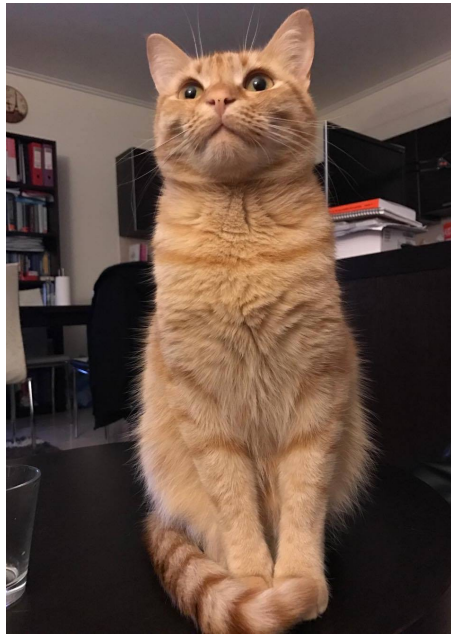
$$R(T) = \widetilde{O}\left(T^{\frac{z+1}{z+2}}\right)$$

Conclusions

Adaptive discretization works for Adversarial Lipschitz bandits, achieves similar results as in the IID case, takes new ideas

Future Directions

1) Mitigate Lipschitz assumptions [further], e.g., via “smoothed regret”
2) Extend to more general pricing problems (ongoing work)
3) Extend to Contextual Bandits

Thank you!