Learning for Dynamic Pricing

Online Posted-Price Mechanisms

Assume the mechanism designer (MD) has infinite copies of the same item.

At round $t = 1, \dots, T$:

1. MD selects price p_t.
2. An agent with (hidden) valuation v_t arrives.
3. Agent buys iff $v_t \geq p_t$.
4. MD observes $1\{v_t \geq p_t\}$.
5. MD gets revenue $p_t \cdot 1\{v_t \geq p_t\}$.

If agents come from a fixed valuation distribution D (i.e., $v_t \sim D$) then the price maximizing the mechanism designer's expected revenue is:

$$p^\star = \max_{p \in [0,1]} p \cdot \underbrace{\Pr_D[v \geq p]}_{\text{"demand curve"}}$$

? How should the MD/learner choose prices $\{p_t\}_{t \in [T]}$ if demand curve is unknown?

[Kleinberg and Leighton, FOCS03]

[Kleinberg and Leighton, FOCS03]

Types of valuation sequences:

1. **Stochastic** from unknown distribution $v_t \sim D, \forall t$.
2. **Adversarial**: $\{v_t\}_{t \in [T]}$ is chosen arbitrarily.
3. **Identical** (but unknown).

Goal: (Tight) Regret Minimization

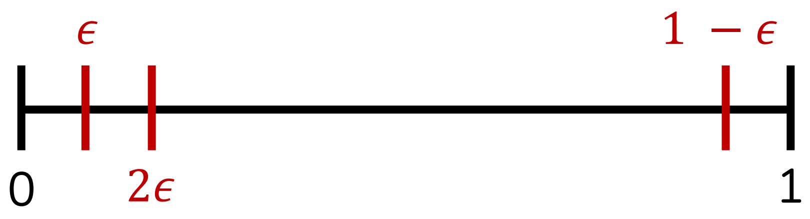
Learning in Dynamic Pricing

Sequence of v_t chosen stochastically/adversarially.

At round $t = 1, \dots, T$:

1. Learner selects price p_t.
2. An agent with (hidden) valuation v_t arrives.
3. Agent buys iff $v_t \geq p_t$.
4. Learner observes $1\{v_t \geq p_t\}$.
5. Learner gets revenue $p_t \cdot 1\{v_t \geq p_t\}$.

Regret upper bound: uniform discretization



Nearly tight in worst case!

For adversarial $\{v_t\}_{t \in [T]}$: [**Podimata** and Slivkins, COLT21] proposed *adaptive discretization algorithm* that works despite the learner's revenue function not being (fully) Lipschitz.

[Kleinberg and Leighton, FOCS03]

Types of valuation sequences:

1. **Stochastic** from unknown distribution $v_t \sim D, \forall t$.
2. **Adversarial**: $\{v_t\}_{t \in [T]}$ is chosen arbitrarily.
3. **Identical** (but unknown).

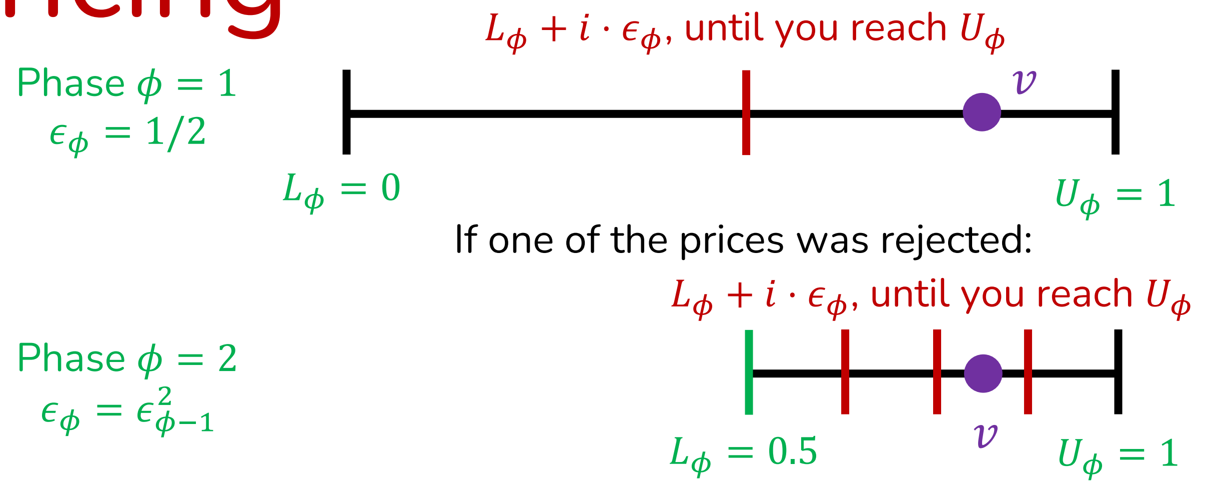
Goal: (Tight) Regret Minimization

Learning in Dynamic Pricing

Common $v_t = v, \forall t \in [T]$.

<u>At round $t = 1, \dots, T$:</u>

1. Learner selects price p_t.
2. An agent with (hidden) valuation v_t arrives.
3. Agent buys iff $v_t \geq p_t$.
4. Learner observes $1\{v_t \geq p_t\}$.
5. Learner gets revenue $p_t \cdot 1\{v_t \geq p_t\}$.



- Continue with phases until interval with length $1/T$.
- Then, play lowest price in interval forever.

$$\boxed{R(T) = \Theta(\log \log T)}$$

[Kleinberg and Leighton, FOCS03]

Types of valuation sequences:

1. **Stochastic** from unknown distribution $v_t \sim D, \forall t$.
2. **Adversarial**: $\{v_t\}_{t \in [T]}$ is chosen arbitrarily.
3. **Identical** (but unknown).

Goal: (Tight) Regret Minimization

Contextual/Feature-Based Pricing

<u>Repeat:</u>

Nature chooses d-dimensional context x_t

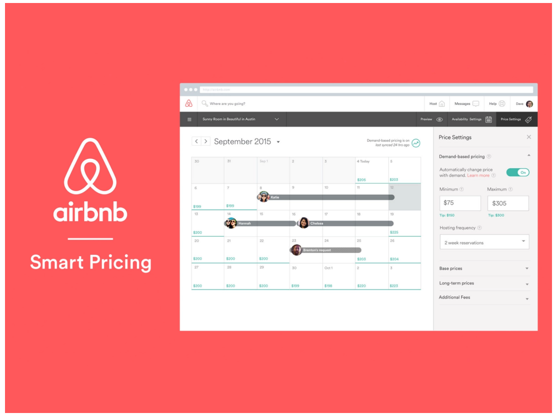
1. Product arrives with *features (location, #rooms, etc)*
2. Airbnb suggests *a price*
3. User *purchases or not*
4. Airbnb *observes feedback (if purchase occurred)*
5. Airbnb *collects revenue*

Learner chooses price $p_t \in \mathbb{R}$

- Exists common feature value $\theta^\star \in \mathbb{R}^d$.
- Agent buys **only iff** context valuation above price: $\langle x_t, \theta^\star \rangle \geq p_t$

Learner's feedback: $1\{\langle x_t, \theta^\star \rangle \geq p_t\}$

Learner's reward: $p_t \cdot 1\{\langle x_t, \theta^\star \rangle \geq p_t\}$



Set prices to collect as much revenue as possible.

$Regret$ = Total revenue for benchmark pricing policy – Total revenue collected

(Mostly) Related Work

Dynamic Pricing is a very important problem in OR; there are a ton of very interesting papers that I'm not listing here

- [Kleinberg and Leighton, FOCS03]: single dim, no context
- [Paes Leme, Sivan, Teng, Worah, ICML21]: single dim, v_t's change by a little from round to round
- [Amin, Rostamizadeh, Syed NeurIPS14]: multidimensional, iid contexts
- [Cohen, Lobel, Paes Leme EC16/MS19], [Lobel, Paes Leme, Vladu EC17/OR18], [Paes Leme and Schneider, FOCS18], [Liu, Paes Leme, Schneider SODA21]: adversarial contexts, refined bounds
- [Mao, Paes Leme, Schneider NeurIPS18]: adversarial contexts, Lipschitz buyers
- [Krishnamurthy, Lykouris, **Podimata**, Schapire, STOC21]: adversarial contexts, some adversarially irrational buyers

Behavioral Model Core Assumption

Full Rationality

- Exists common feature value $\theta^\star \in \mathbb{R}^d$.
- Agent buys **only iff** their context valuation above price: $\langle x_t, \theta^\star \rangle \geq p_t$

!

But *sometimes* agents deviate from the prescribed behavioral model (irrationality)

e.g., agents may not buy, even if price is below their valuation, due to their psychological state

Problem

Model misspecifications can cause arbitrary failures to standard algorithms

Main Question

How do we design **contextual search** algorithms that are **robust** to the presence of some **irrational** agents?

Contextual Pricing for Fully Rational Agents

For round $t = 1 \ldots T$:

1. Nature chooses observable context $x_t \in \mathbb{R}^d$ with $||x_t||_2 \leq 1$

[Cohen, Lobel, Paes Leme, EC16/MS19]
[Lobel, Paes Leme, Vladu, EC17/OR18]
[Paes Leme, Schneider, FOCS18]
[Liu, Paes Leme, Schneider, SODA21]

2. Learner selects *a price* p_t
3. Agent makes purchase if $\langle x_t, \theta^\star \rangle \geq p_t$
4. Learner observes if purchase occurred *(one-sided feedback)*
5. Learner collects revenue if purchase occurred

$$Regret = O(d \log T)$$

Benchmark pricing policy: $p_t^\star = \langle x_t, \theta^\star \rangle$

- For **revenue** objective, [Liu, Paes Leme, Schneider, SODA21] improved to: $O(d \log \log T)$
- For $d = 1$, optimal solution due to [Kleinberg and Leighton, FOCS03]

Contextual Pricing for **Adversarially Irrational** Agents

Total corruption: $C = \sum_t c_t$

For round $t = 1 \dots T$:

1. Nature chooses observable context $x_t \in \mathbb{R}^d$ with $||x_t||_2 \leq 1$

   and corruption $c_t \in \{0, 1\}$
2. Learner selects *a price* p_t
3. Agent makes purchase if $\langle x_t, \theta^\star \rangle \geq p_t$ **if $c_t = 0$**

   or arbitrarily if $c_t = 1$
4. Learner observes if purchase occurred *(one-sided feedback)*
5. Learner collects revenue if purchase occurred

Extends the corruptions model studied in:
[Lykouris, Mirrokni, Paes Leme STOC18]
[Gupta, Koren, Talwar COLT19]
[Zimmert, Seldin JMLR21]

Desiderata for our Algorithms

✓ Graceful degradation of regret with C
✓ No knowledge of C assumed (*agnostic*)

Total corruption: $C = \sum_t c_t$
Number of features: d

Robust contextual search algorithm agnostic to C achieving

$Regret \leq C \cdot d^3 \cdot (\log T)^3$ (with high probability)

[Ulam 76], [Rivest, Meyer, Kleitman, Winklmann, Spencer, JCSS80], [Karp and Kleinberg, SODA07]
For Ulam's game ($d = 1$) tight bound of $\log T + C \log C + C \log\log T$ [RMKWS, JCSS80]

Extends classical Ulam's game (aka twenty questions with a liar) in 2 ways:

1. Multidimensional
2. Agnostic to C

The Non-Robust Algorithm

[Lobel, Paes Leme, Vladu, EC17/OR18]

Multidimensional Binary Search

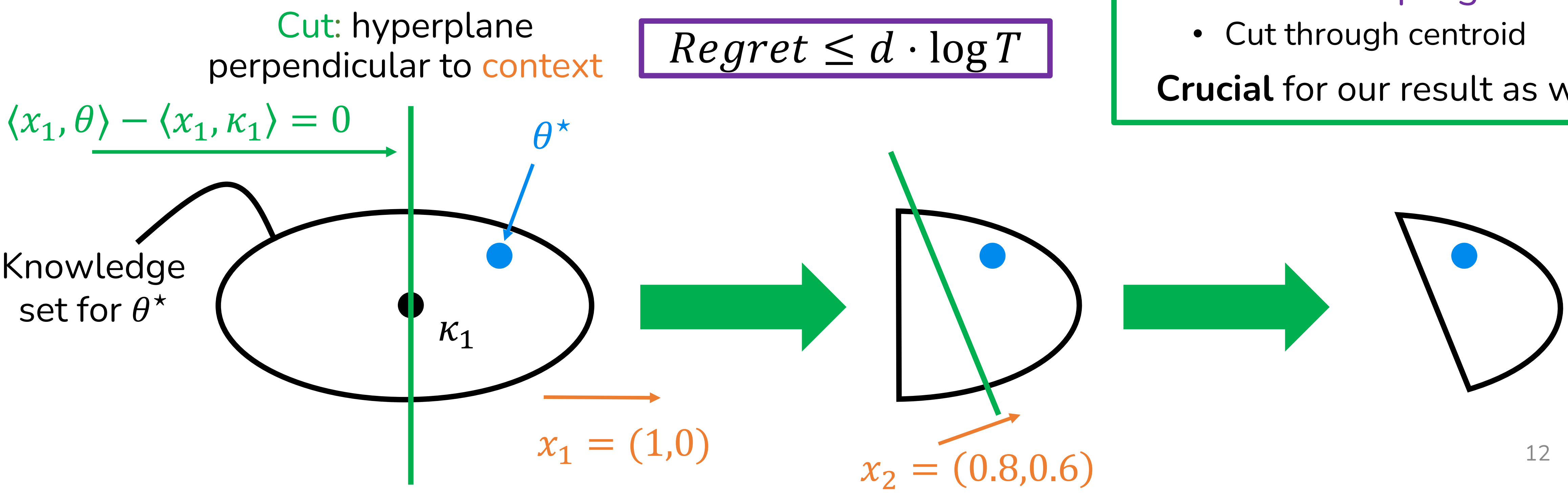
- Maintain active knowledge set with feasible values for $\theta^\star$.
- Set price $\langle \kappa_1, x_1 \rangle$ where κ_1 centroid of knowledge set.
- Eliminate inconsistent side of knowledge set.

Important properties of cut

1. Never eliminate $\theta^\star$
   - Retain all parameters consistent with feedback
2. Volumetric progress
   - Cut through centroid

Crucial for our result as well

$$\boxed{Regret \leq d \cdot \log T}$$



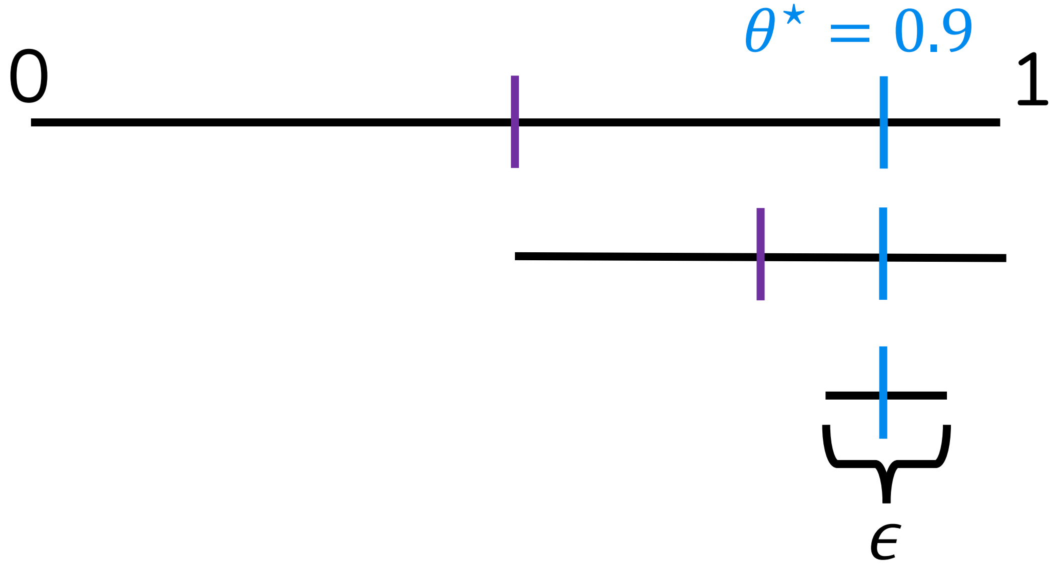
Issue with Adversarial Irrationality: Ulam's Game

Challenge

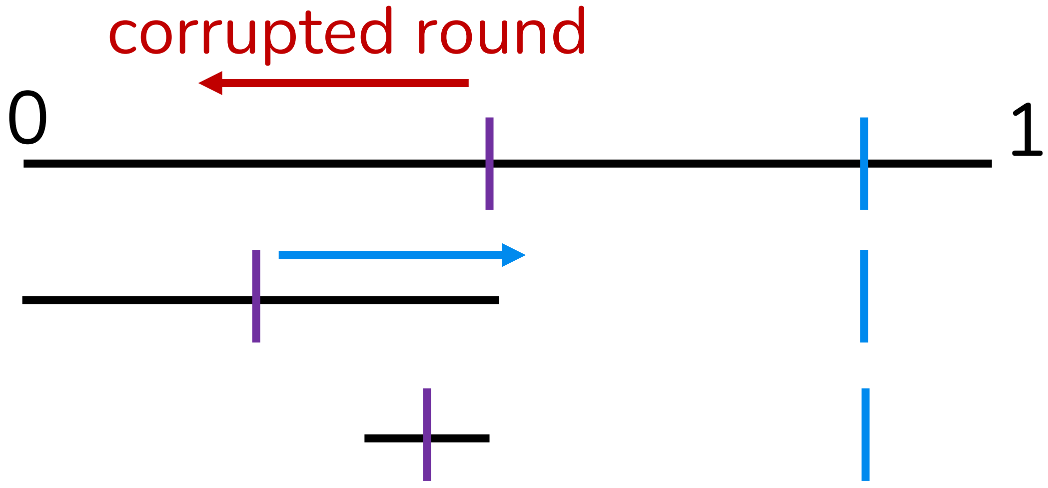
How can you fix it (in high dimensions) when you know that $\bar{c}$ responses are corrupted?

<u>Binary search ($d = 1$)</u>

- I fix a number $\theta^\star \in [0,1]$.
- How many queries do you need to find it with accuracy ϵ?



… when *one response* is corrupted?

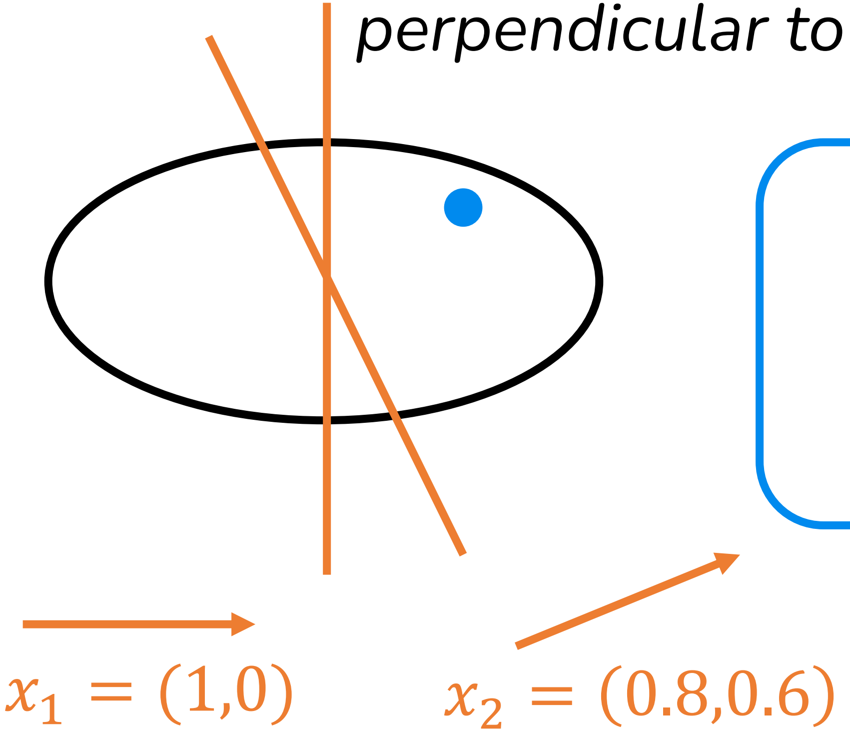


Robust Volumetric Progress

Challenge 1

We cannot repeat the same query (contexts are different at different rounds).

Context cut: hyperplane perpendicular to context



Idea 1

Keep "penalty" for each parameter & make cut once a context cut fully retains protected region on one side

- Protected region: all parameters with "penalty" $\leq \bar{c}$

Robust Volumetric Progress

Challenge 2

We may never have a context cut with the protected region fully on one side.

Counterexample: Even with infinite contexts and $\bar{c} = 1$, no such context cut.

Robust Volumetric Progress

Challenge 2

We may never have a context cut with the protected region fully on one side.

Counterexample: Even with infinite contexts and $\bar{\boldsymbol{c}} = \mathbf{1}$, no such context cut.

Idea 2

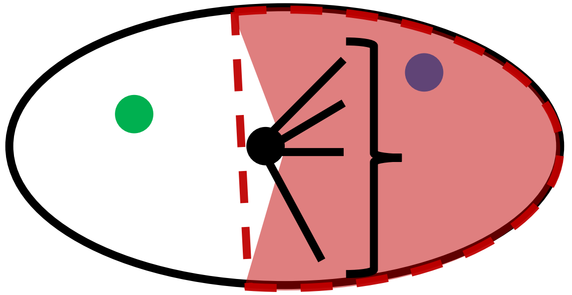
Combine context cuts to compute a “valid cut”.

<u>**Important properties of valid cut**</u>

1. Never eliminate $\theta^\star$
   - Fully retains protected region on one side.
2. Volumetric progress
   - Cross **close** to centroid

Idea 3

Show that $2d \cdot (d+1) \cdot \bar{c} + 1$ context cuts have enough information to compute such a valid cut **(Caratheodory’s theorem)**



Robust Volumetric Progress

Challenge 2

We may never have a context cut with the protected region fully on one side.

Counterexample: Even with infinite contexts and $\bar{c} = 1$, no such context cut.

Idea 2

Combine context cuts to compute a "valid cut".

Important properties of valid cut

1. Never eliminate $\theta^\star$
   - Fully retains protected region on one side.
2. Volumetric progress
   - Cross **close** to centroid

Idea 3

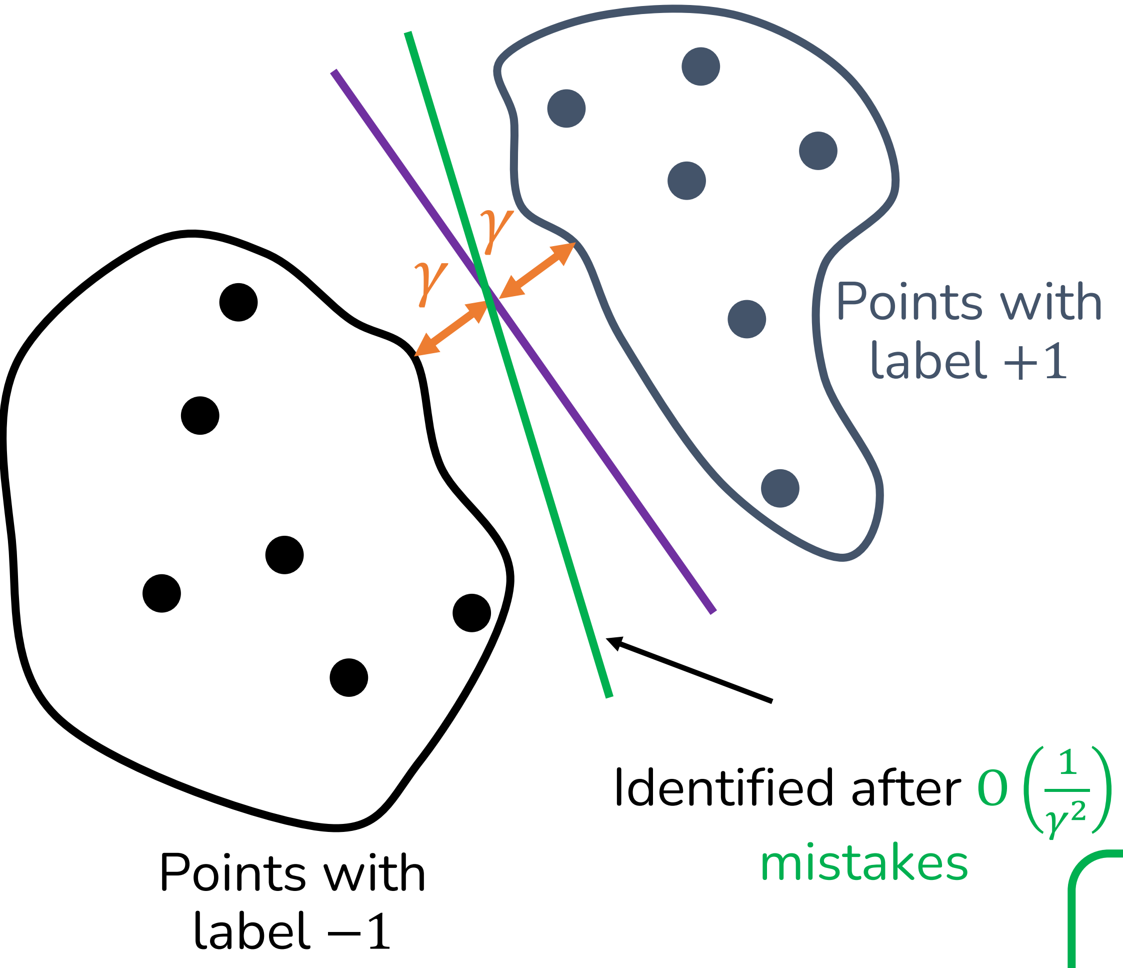
Show that $2d \cdot (d+1) \cdot \bar{c} + 1$ context cuts have enough information to compute such a valid cut **(Caratheodory's theorem)**

Idea 4

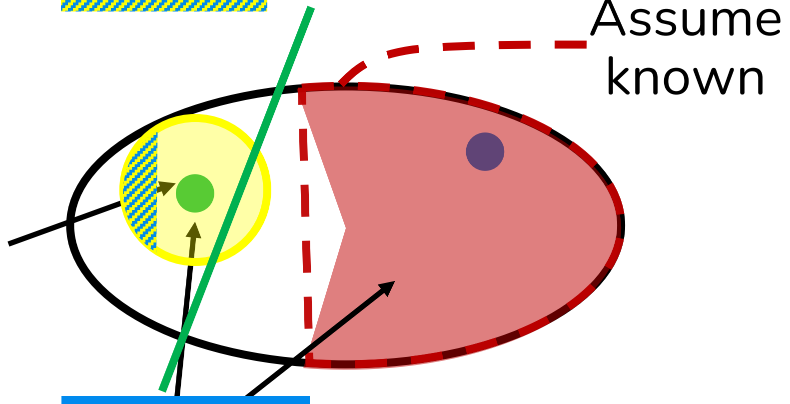
Use Perceptron as a **witness** for finding a valid cut.

Perceptron as a Witness

Standard Perceptron



- Sample randomly from ball
- With constant probability, you will sample point from pattern.



Idea

If we knew them exactly + they had margin → Perceptron → done

- If dataset correct → Perceptron mistakes $O\left(\frac{1}{\gamma^2}\right)$
- Else, resample

Unknown C: Multi-Layering Race to the Rescue

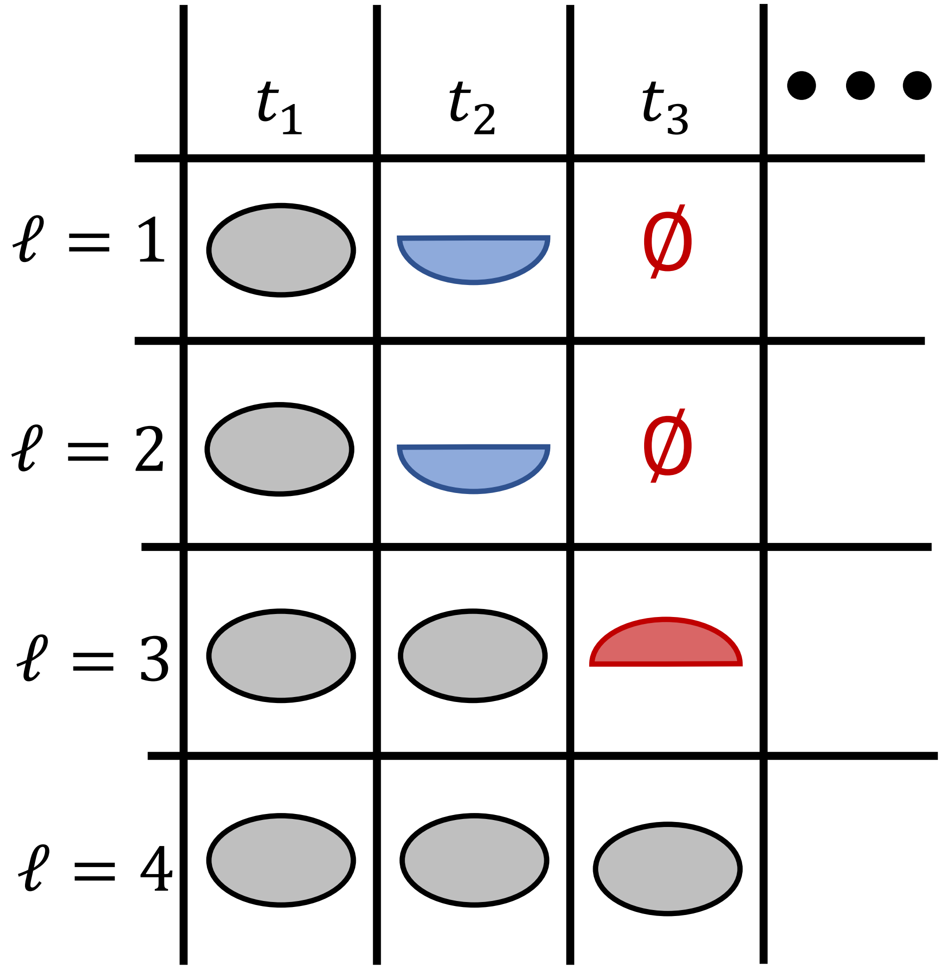
Original: [Lykouris, Mirrokni, Paes Leme STOC18]
Adaptation to continuous action spaces: [Krishnamurthy, Lykouris, **Podimata**, Schapire, STOC21

Subsampling

- $\log T$ layers
- Layer ℓ corresponds to corruption level: $C = 2^{\ell}$
- Sample a layer ℓ to play, with probability $2^{-\ell}$

Global Eliminations

- Maintain consistent knowledge sets
- Below layers: more robust → they should inform above layers.



Incentive-Compatible and Incentive-Aware Learning

! Need to draw both from literature in ML and Mechanism Design.

Questions to keep in mind for modeling incentive-compatible and incentive-aware learning settings

Decide who to admit in college

"Pass" the cutoff and get admitted

1. What is the goal of the learner, and what the goals of the agents?

Candidates' submitted SAT scores

2. What is observed by the learner regarding the agents?

Times that they can take SAT, amount of money spent for tutoring

3. What is the agents' ability to respond to decisions that are made?

Stackelberg Security Games

Strategic Classification

Dynamic Pricing

The Broader Picture

We have merely scratched the surface!

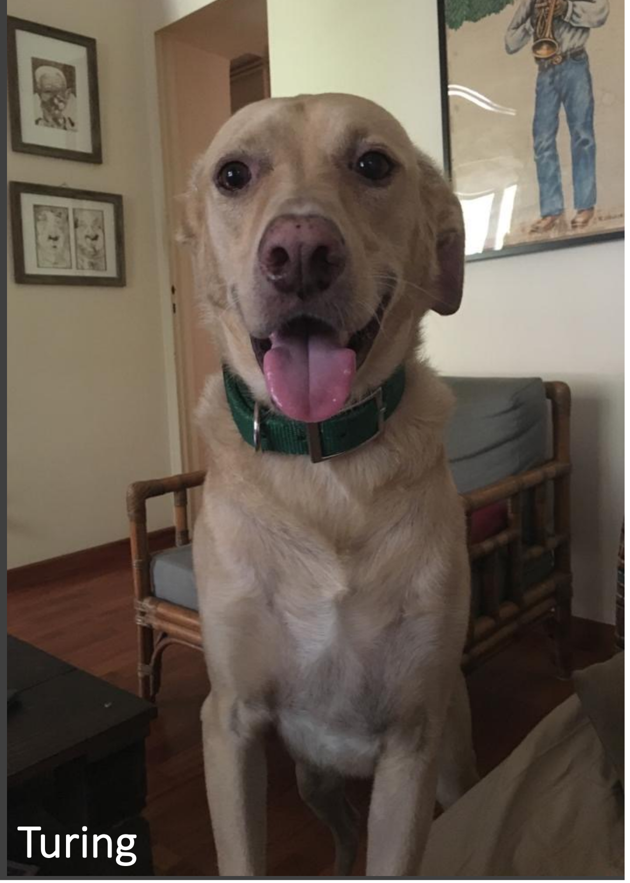
This Course

Incentive-Compatible and Incentive-Aware Learning

Examples of topics not covered here: Learning in auctions, learning in games, learning from revealed preferences, learning for matching markets, recommendation systems and many more…

If interested in learning more about Incentive-Compatible Learning:

[Chen, **Podimata**, Procaccia, Shah, EC18]: algorithms for strategyproof high dimensional linear regression
[Feng, **Podimata**, Syrgkanis, EC18]: bandit algorithms for learning bidders in online auctions
[Freeman, Pennock, **Podimata**, Vaughan, ICML20]: strategyproof online prediction algorithms with optimal regret guarantees



Thank You!