Algorithms for Incentive-Compatible and Incentive-Aware Learning

Chara Podimata, Harvard University

Who am I

Volos, Greece

NTUA, Greece

Ugrad in EE + CS

→Thesis advisor
→AGT study groups
→ Thesis: combinatorial auctions w. budgets

Harvard

PhD in CS

→ PhD advisor
→ Dissertation: Algorithms for Incentive-Compatible and Incentive-Aware ML

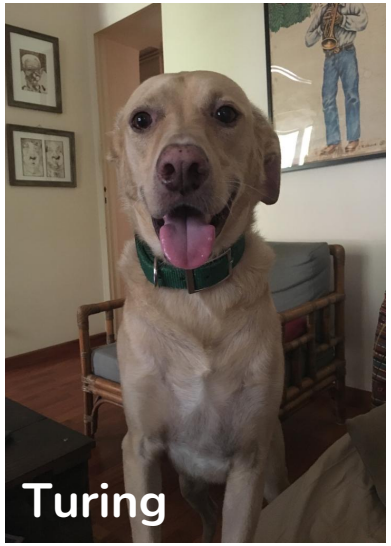
Internships

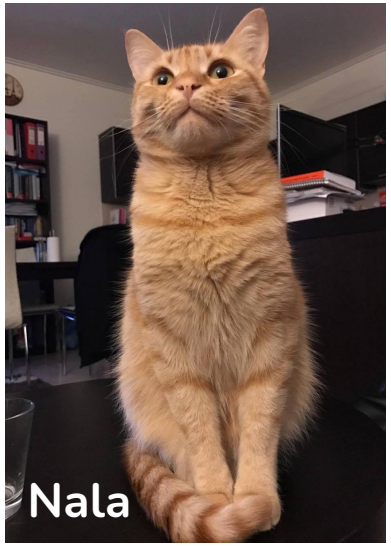
Microsoft Research

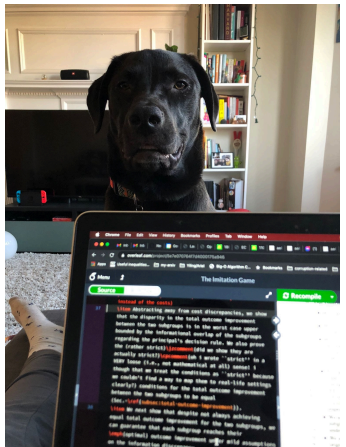
Google Research

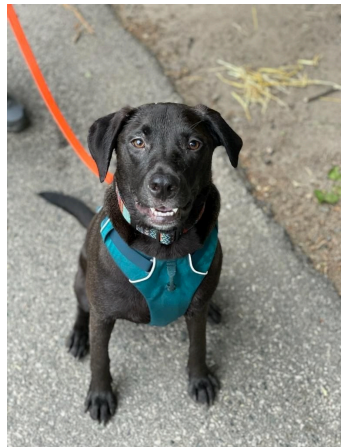
Who am I (Outside of Research)

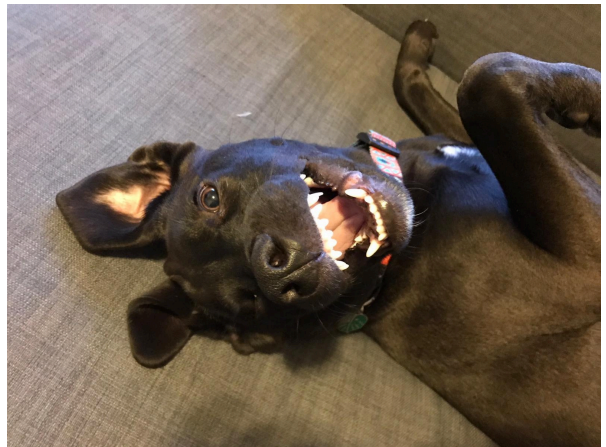
Extended family in Greece

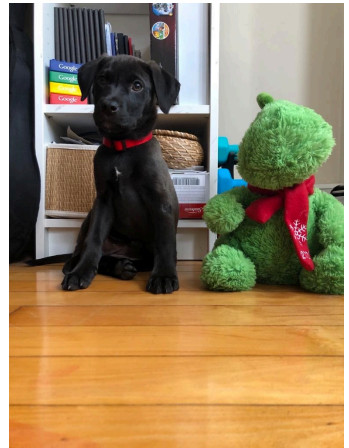












Agenda

Time	Activity
06:45 am – 07:00 am ET	Checking in
07:00 am – 08:30 am ET	Introduction + Primers on AGT and Online Learning
08:30 am – 08:45 am ET	Coffee break
08:45 am – 09:45 am ET	Learning for Stackelberg Security Games
09:45 am – 10:00 am ET	Coffee break
10:00 am – 11:30 am ET	Strategic Classification
11:30 am – 11:45 am ET	Coffee break
11:45 pm – 01:00 pm ET	Learning for Dynamic Pricing

Strategic Prediction


The Washington Post
Democracy Dies in Darkness
Get one yea

Business

Student tracking, secret scores: How college admissions offices rank prospects before they apply

Before many schools even look at an application, they comb through prospective students' personal data, such a financial history

→ improve GPA
→ retake GRE / pay for classes
→ change schools

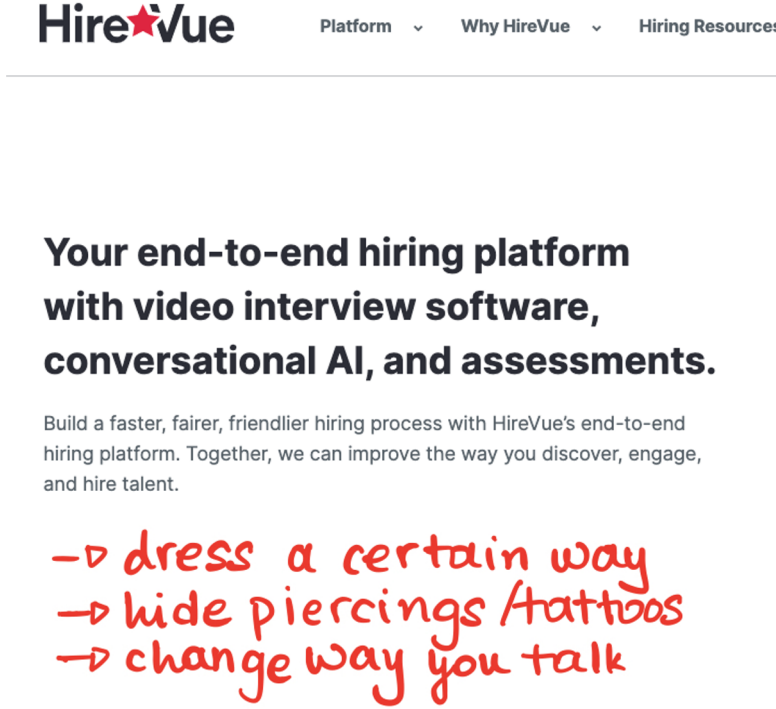
ML algorithms making consequential decisions are almost everywhere nowadays

The New York Times

Is an Algorithm Less Racist Than a Loan Officer?

Digital mortgage platforms have the potential to reduce discrimination. But automated systems provide rich opportunities to perpetuate bias, too.

→ increase # credit cards
→ increase # bank accounts
→ improve credit history



→ dress a certain way
→ hide piercings / tattoos
→ change way you talk

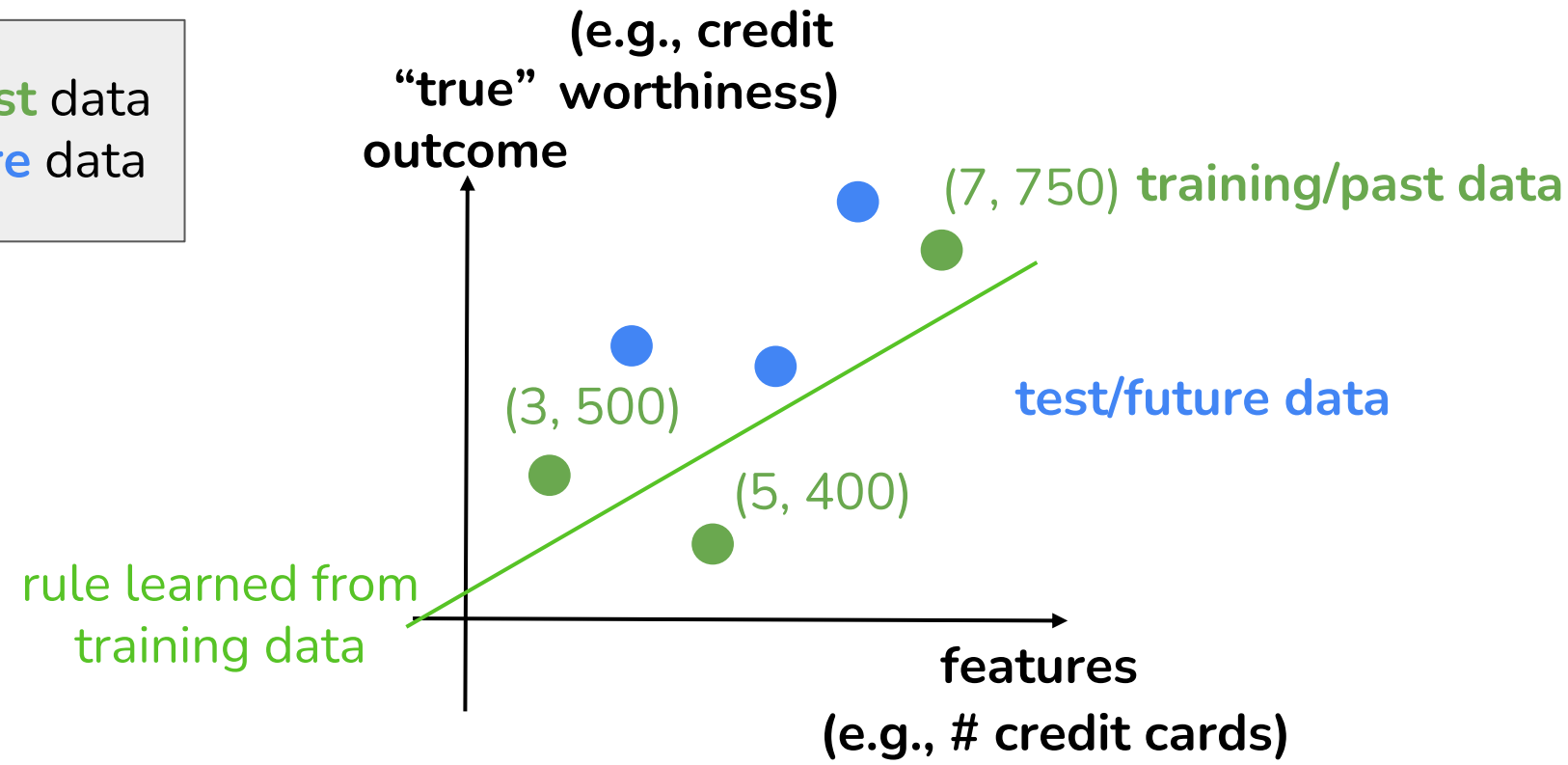
But **why** use automated decision making/ML?

Patterns in training/past data = patterns in test/future data

→ **abundance of people's data** and the **heart of ML paradigm**

Standard ML

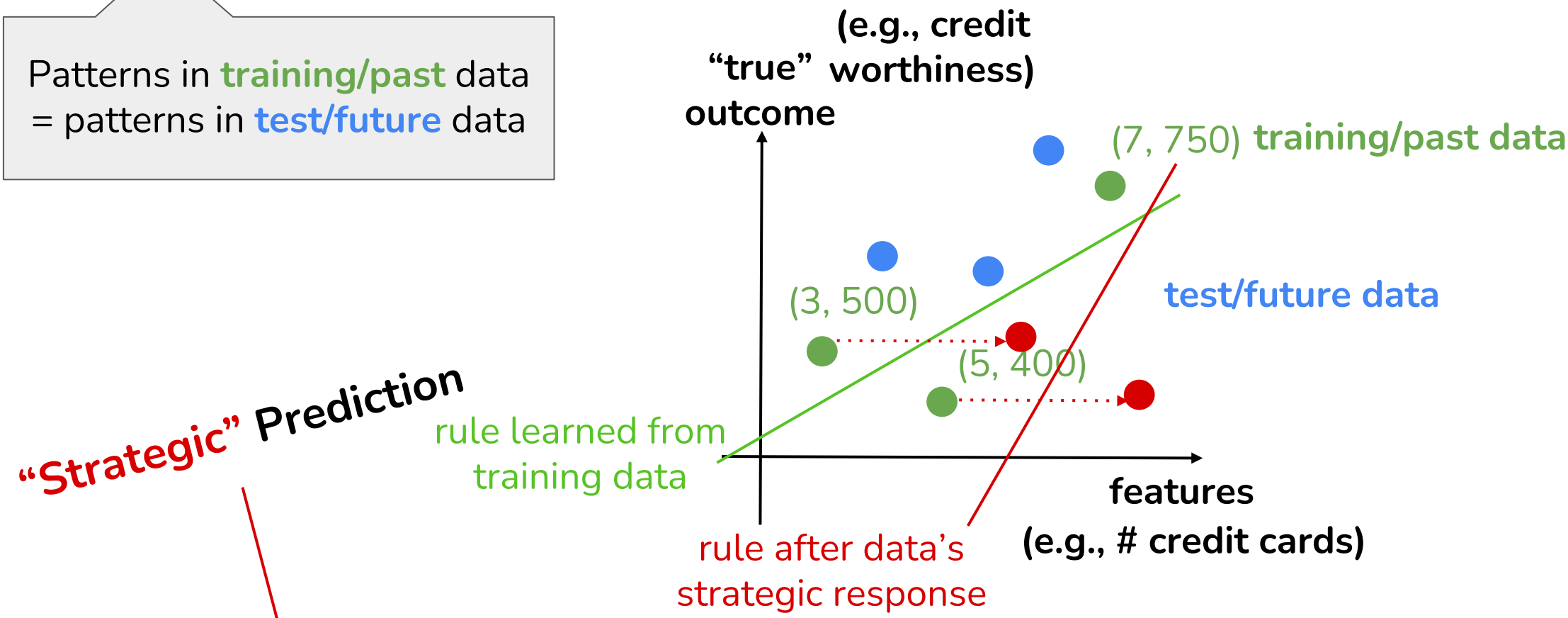
Patterns in **training/past** data
= patterns in **test/future** data



- Features: e.g., age, education level, ZIP code, # credit cards, # bank accounts, past credit history, etc.
- "True" outcome: e.g., current credit score, loan/mortgage qualification, etc.

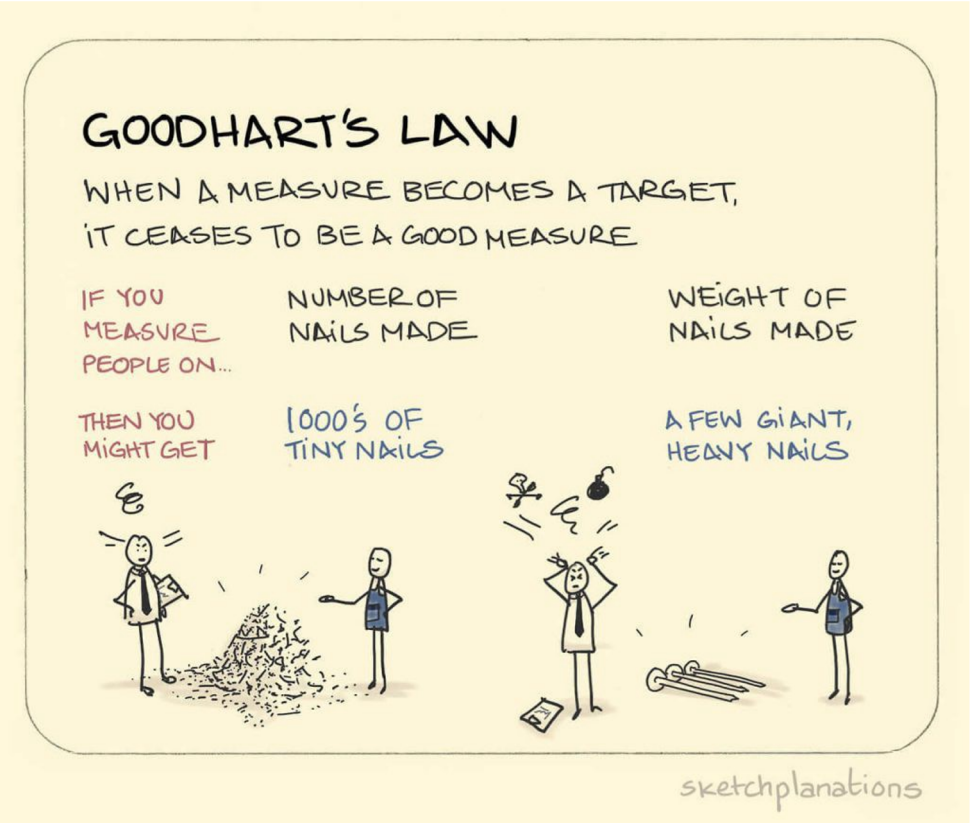
Why standard ML is not enough

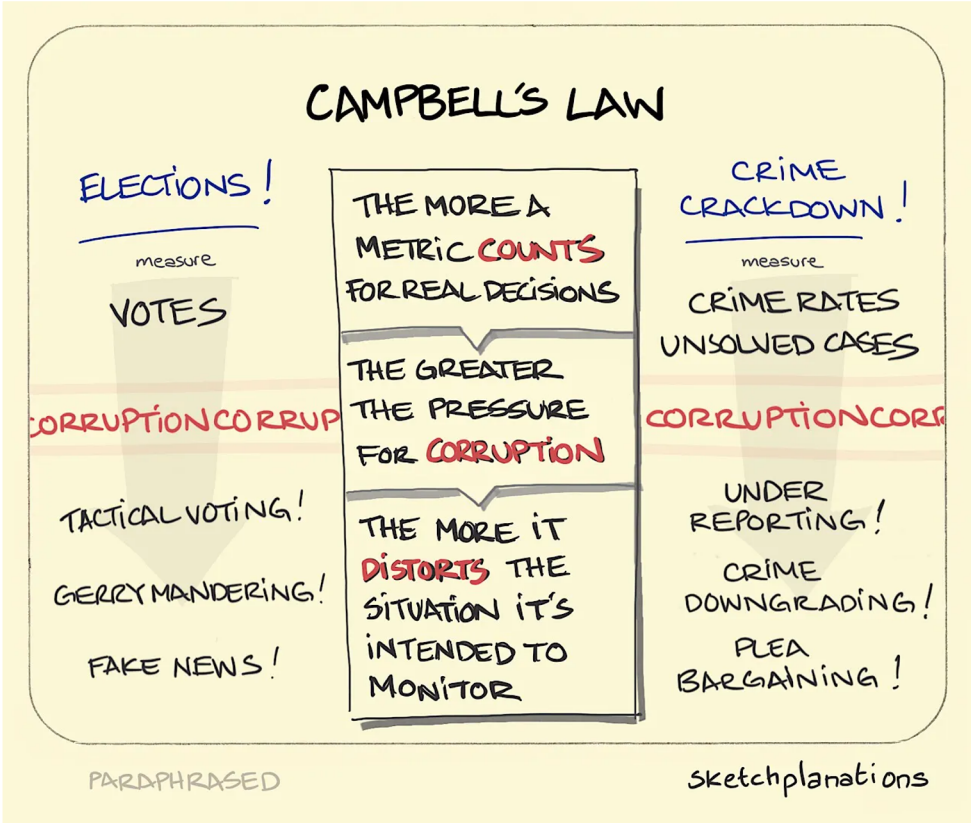
Patterns in **training/past** data
= patterns in **test/future** data



Data **corresponds to individuals who have agency** and want to affect the decisions made on them by the ML algorithms.

Similar problem, different fields





- School's admission rule: admit anyone who has more than 100 books in their house.
- Students with (say) 90 and more books can "easily" buy (**but need not read!**) 10 more and get admitted.

→ defeats the purpose of having the # books as a measure of qualifications

Strategic data sources for ML algorithms present both a challenge and an opportunity

Data received for training ML algos is not accurate

ML algorithms can adapt to the behavior of strategic agents.

Solution

Incentive-Compatibility

Incentive-Awareness

Incentive-Compatible and Incentive-Aware Learning

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

➡ Brief introductions on Game Theory, Mechanism Design and ML

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

1. What is the goal of the learner, and what the goals of the agents?
   - goal of the learner: Decide who to admit in college
   - goals of the agents: "Pass" the cutoff and get admitted
2. What is observed by the learner regarding the agents?
   - What is observed: Candidates' submitted SAT scores
3. What is the agents' ability to respond to decisions that are made?
   - ability to respond: Times that they can take SAT, amount of money spent for tutoring

Incentive-Compatible ML

Linear Regression:

- IC: [Perote, Perote-Pena, MSS04], [Dekel, Fischer, Procaccia, JCSS10], [Cummings, Ioannidis, Ligett, COLT15], [Chen, Procaccia, **Podimata**, Shah, EC18],[Farhadkhani, Guerraoui, Hoang arXiv21]
- Other considerations: [Ben-Porat, Tennenholtz, NeurIPS17], [Ben-Porat, Tennenholtz, EC19], [Hossain, Shah, AAMAS20], [Gast, Ioannidis, Loiseau, Roussillon, TEAC20], [Hossain, Shah, preprint]

Classification: [Meir, Procaccia and Rosenschein, AAMAS10], [Meir, Almagor, Michaely, Rosenschein, AAMAS11], [Meir, Procaccia, Rosenschein, AI12]

Clustering: [Perote, Perote-Pena, EB03]

Active Learning: [Enchenique, Prasad, ITCS20]

Prediction with Expert Advice: [Roughgarden, Schrijvers, NeurIPS17], [Freeman, Pennock, **Podimata**, Vaughan ICML20], [Frongillo, Gomez, Thilagar, Waggoner EC21]

Univariate Estimation: [Caragiannis, Procaccia, Shah, ICML16]

Information Elicitation for Statistical Inference: [Cai, Daskalakis, Papadimitriou, COLT15], [Liu and Chen, NeurIPS18], [Chen, Immorlica, Lucier, Syrgkanis, Ziani, EC18], [Kong, Schoenebeck, Tao, Yu, AAAI20]

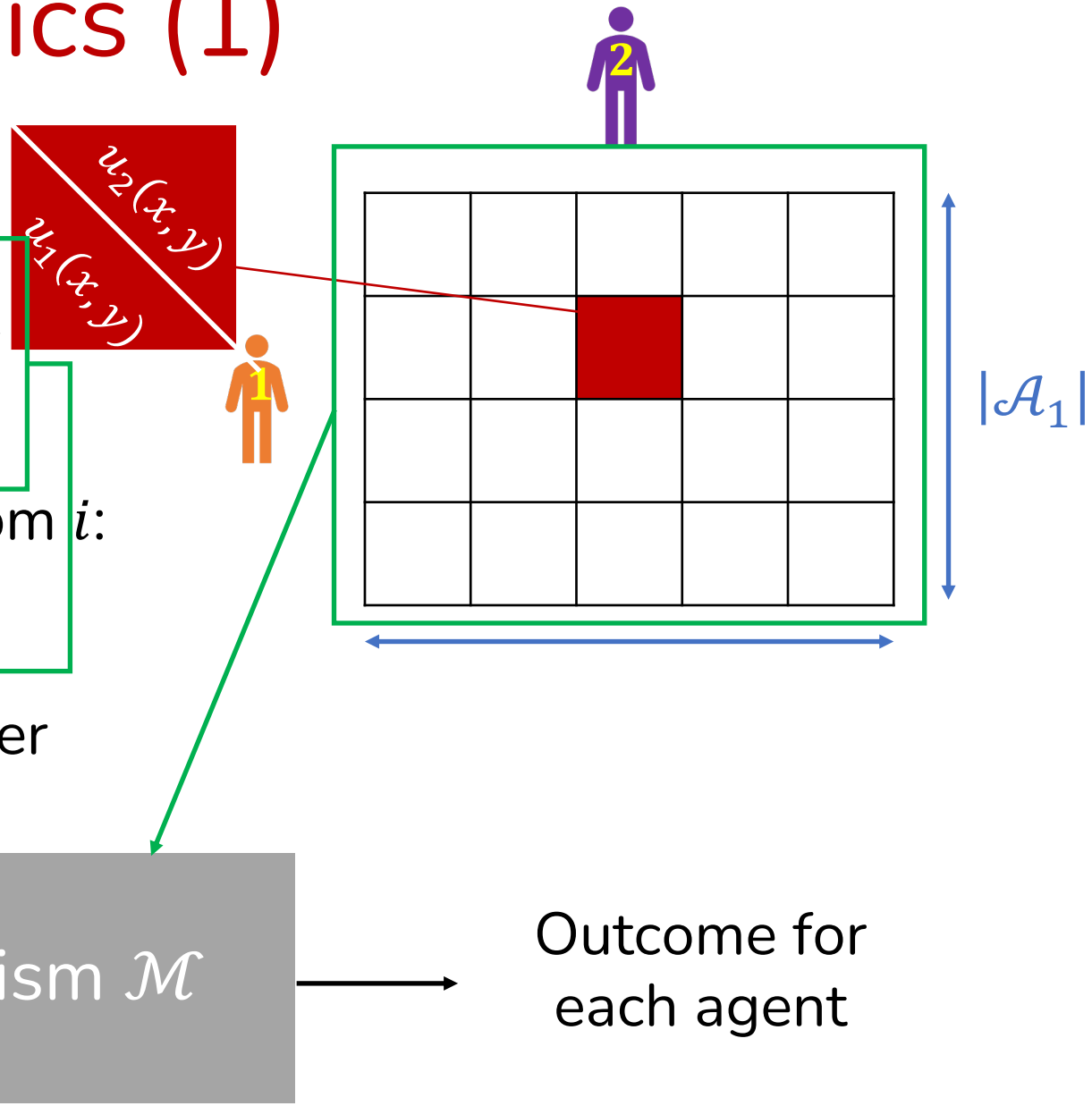
Game Theory and Mechanism Design Basics

Mechanism Design Basics (1)

- N agents, each with **actual** type v_i
- Agents **choose** their **declared** types b_i
- Set of alternatives for agent i: $\mathcal{A}_i$
- Set of alternatives for agents apart from i:

$$\mathcal{A}_{-i} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_{i-1} \times \mathcal{A}_{i+1} \times \cdots \times \mathcal{A}_N$$

- i's utility for playing $x_i \in \mathcal{A}_i$ when other agents play $x_{-i} \in \mathcal{A}_{-i}$



Auctions

N bidders  Item being auctioned 

(**Actual Types**) Valuations $v_1, v_2, \cdots, v_N$ (Valuation vector $\boldsymbol{v} = (v_1, \ldots, v_N)$)

(**Declared Types**) Bids $b_1, b_2, \cdots, b_N$ (Bid vector $\boldsymbol{b} = (b_1, \ldots, b_N)$)

Auction determines the allocation and payment rule

who wins what they pay

Bidder's quasilinear utility: $u_i(b_i, b_{-i}) = (v_i - \text{payment}) \cdot 1\{i \text{ wins item}\}$

! Affected by both true and declared types!

Mechanism Design Basics (2)

Individually-Rational (IR) if by reporting truthfully you get non-negative utility no matter what others do $u_i\left(\mathcal{M}(v_i, b_{-i})\right) \geq 0, \forall i \in [N]$

utility of not participating

Example: First Price Auction

- Highest bidder wins
- Pays her bid

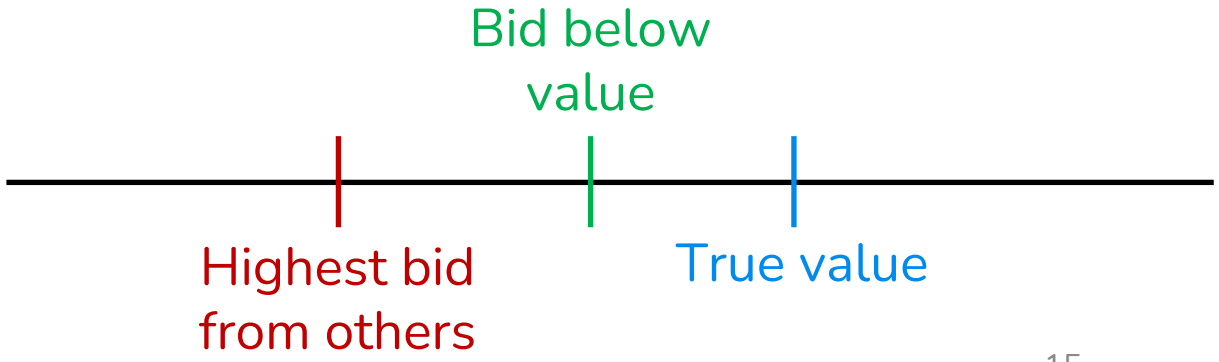
If $v_i = 2$ for  then for $b_i = 2 \rightarrow u_i(v_i, b_{-i}) = 0$

$u_i(b_i, b_{-i}) = (v_i - \text{payment}) \cdot 1\{i \text{ wins item}\}$

Incentive-Compatible (IC) if reporting true value gives highest utility

$u_i\left(\mathcal{M}(v_i, b_{-i})\right) \geq u_i\left(\mathcal{M}(b_i', b_{-i})\right), \forall i \in [N]$

First Price Auction is **not** IC.



Mechanism Design Basics (2)

Individually-Rational (IR) if by reporting truthfully you get non-negative utility no matter what others do $u_i\left(\mathcal{M}(v_i, b_{-i})\right) \geq 0, \forall i \in [N]$

utility of not participating

Example: First Price Auction

- Highest bidder wins
- Pays her bid

If $v_i = 2$ for  then for $b_i = 2 \rightarrow u_i(v_i, b_{-i}) = 0$

$u_i(b_i, b_{-i}) = (v_i - \text{payment}) \cdot 1\{i \text{ wins item}\}$

Incentive-Compatible (IC) if reporting true value gives highest utility

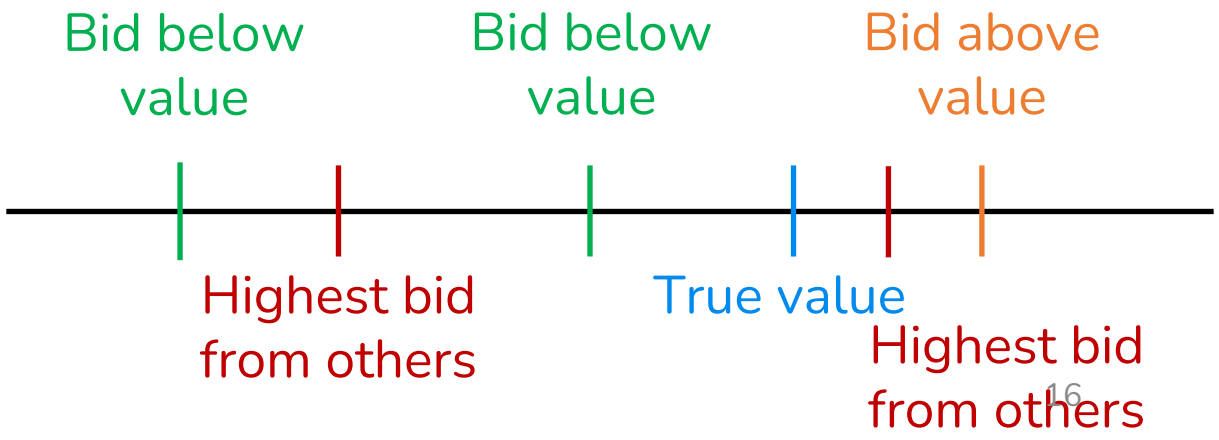
$u_i\left(\mathcal{M}(v_i, b_{-i})\right) \geq u_i\left(\mathcal{M}(b_i', b_{-i})\right), \forall i \in [N]$

Example: Second Price Auction

- Highest bidder wins
- Pays bid of second highest

Bidding below can be worse.

Bidding above can be worse.



Posted-Price Mechanisms

Mechanism designer wants to sell a series of identical items  (assume she has infinite copies)

She posts a fixed price for the item: $p \in (0,1)$.

by definition IC

Agent with private valuation v arrives and buys iff $v \geq p$, else leaves.

Seller collects reward: $p \cdot 1\{v \geq p\}$.

Nash Equilibrium (NE)

Mixed Strategy $x^\star = (x_1^\star, \dots, x_N^\star)$ where $x_i^\star \in \Delta_{|\mathcal{A}_i|}$ is **Nash Equilibrium** if:
$u_i(x_i^\star, x_{-i}^\star) \geq u_i(\tilde{x}_i, x_{-i}^\star)$ for **any** $\tilde{x}_i$

Zero-Sum Game Example

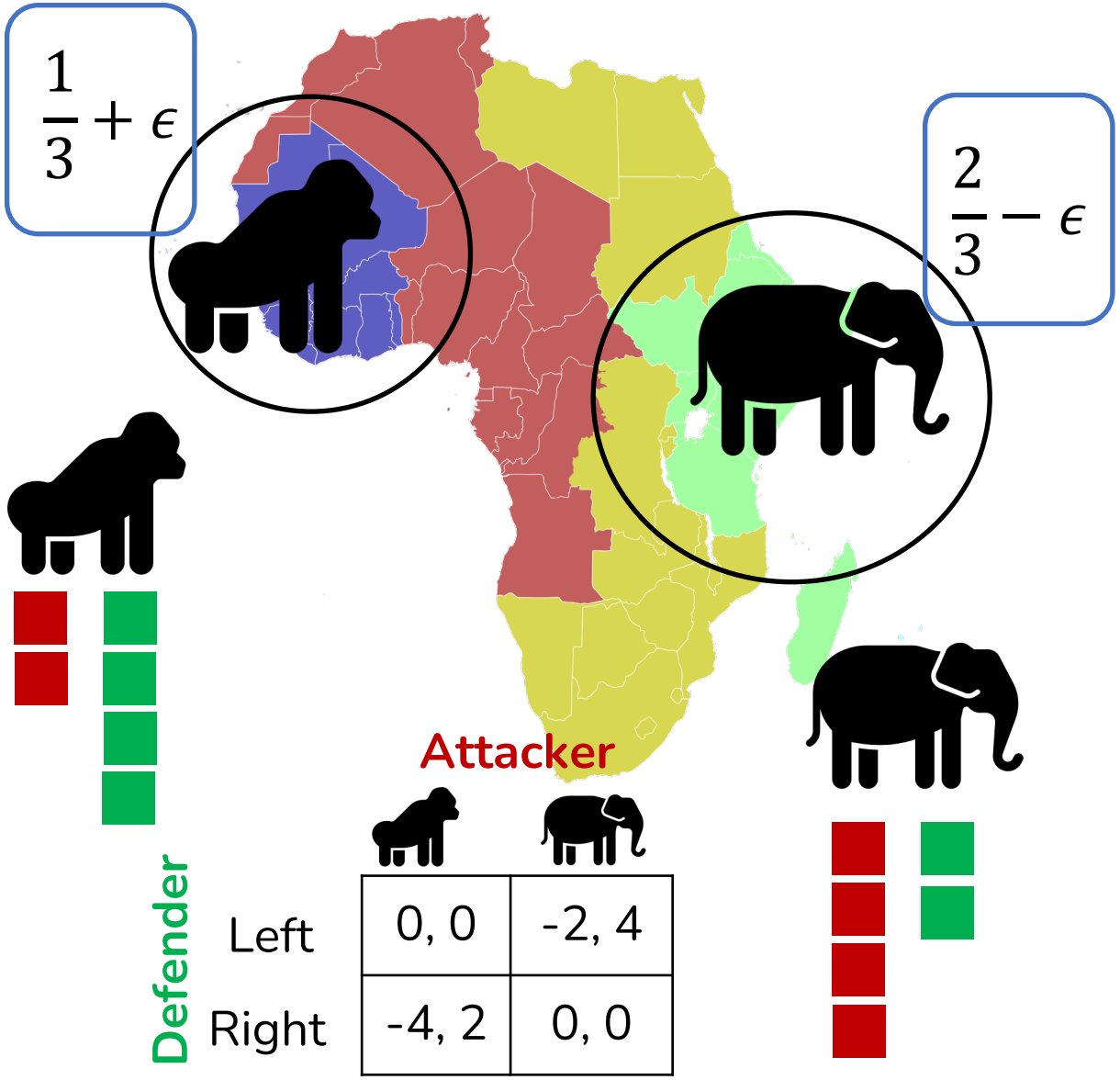
Example from: Sanjoy Dasgupta, Christos Papadimitriou and Umesh Vazirani. Algorithms. McGraw-Hill, 2006.

	2/7	5/7	
	Morality	**Tax-Cuts**	
Economy	3, -3	-1, 1	3/7
Society	-2, 2	1, -1	4/7

Remark

In zero-sum games, any of the players can move first, and the other best-responds.

Stackelberg Equilibria (mixed)



Optimal Policy

- Attacker's payoff:  : $2 \cdot \frac{2}{3}$ and  : $4 \cdot \frac{1}{3}$
- Defender's payoff: $-2 \cdot \frac{1}{3}$

Mixed Strategy $x^\star \in \Delta_{|\mathcal{A}_{\text{leader}}|}$ is **Stackelberg Equilibrium** if:

$$u_l\left(x^\star, r_f(x^\star)\right) \geq u_l\left(\tilde{x}, r_f(\tilde{x})\right) \text{ for } \textbf{any } \tilde{x}$$

Randomization is crucial!

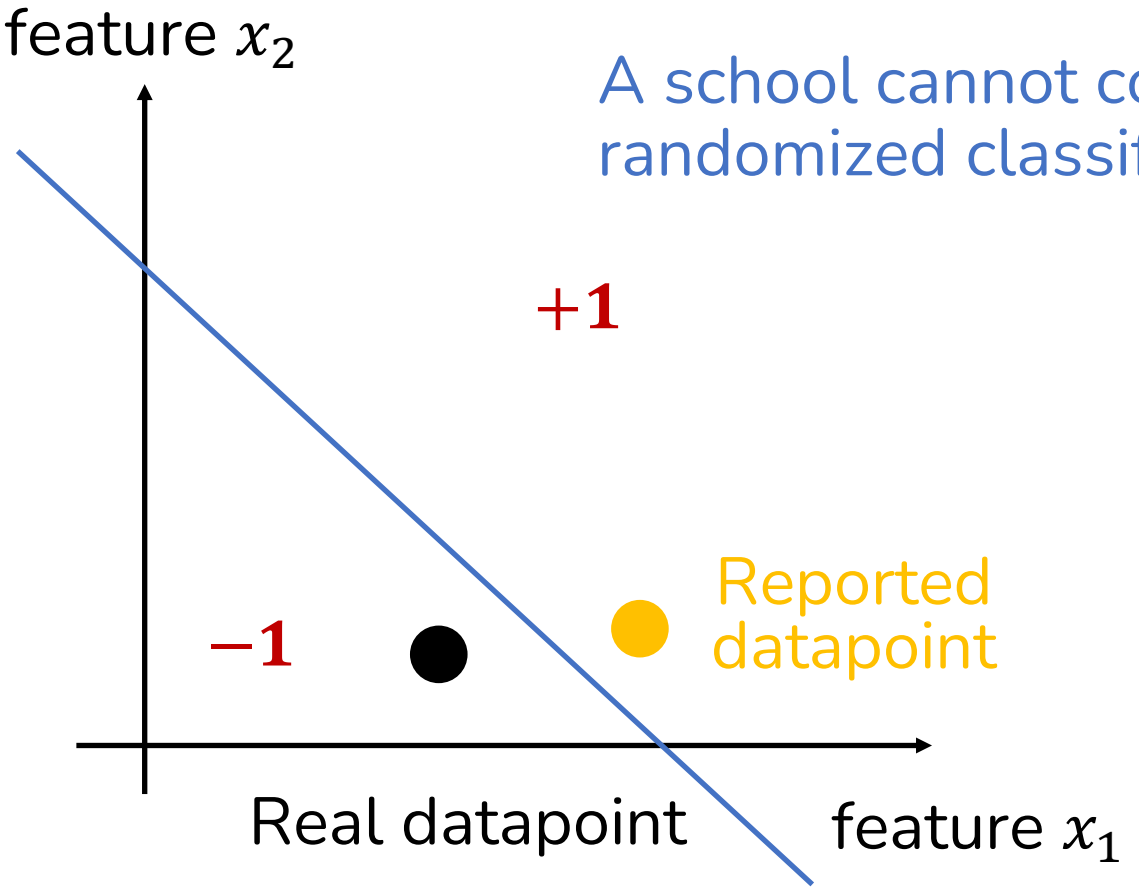
Slide adapted from Nika Haghtalab

Stackelberg Games with Pure Strategies

Randomization is not desired in some settings (e.g., classification for schools' admissions).

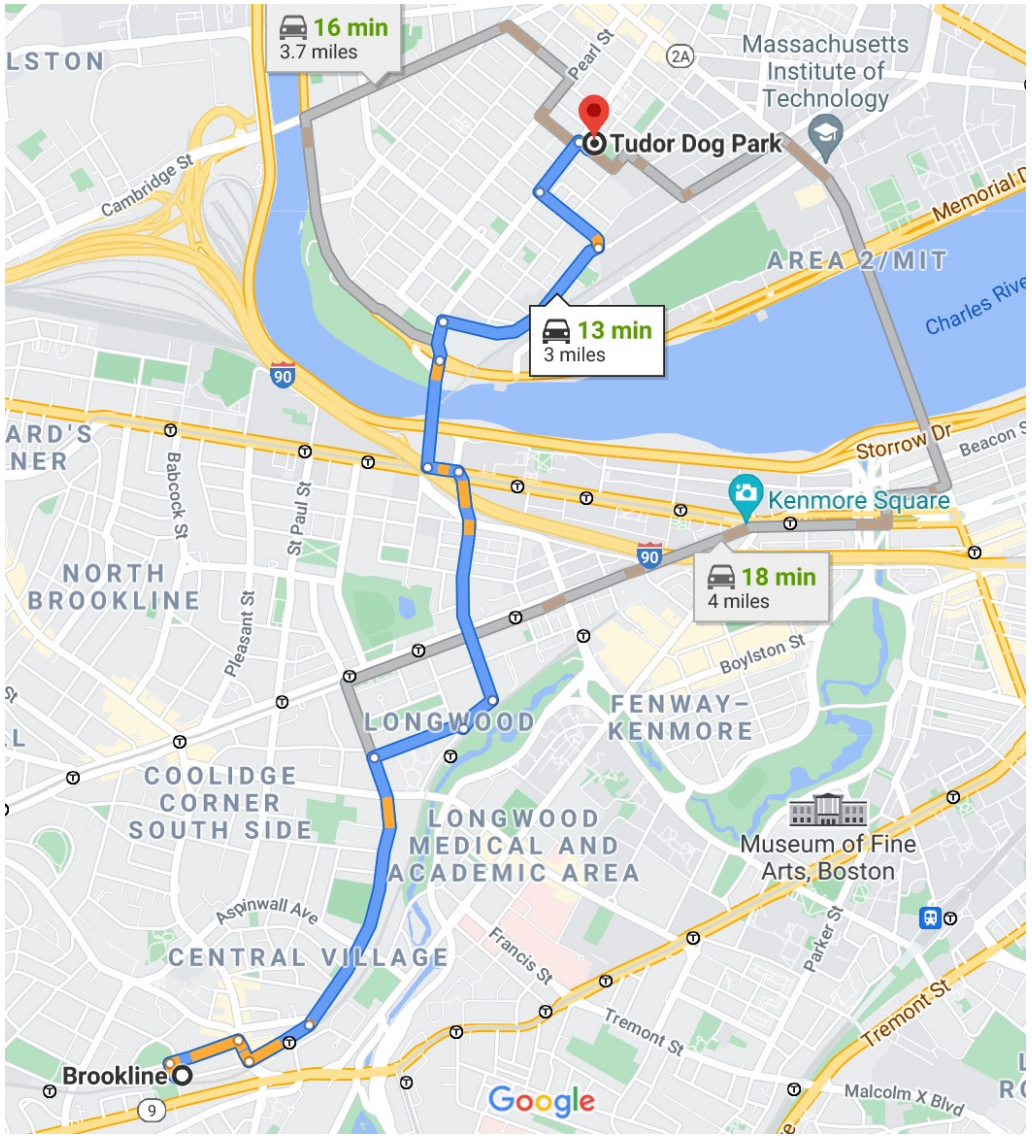
Leader commits to one classifier

A school cannot commit to a randomized classifier for students.



Online Learning and Multi-Armed Bandits Basics

Online Learning -- Intuition



- Every day I need to go from home → dog park.
- I know there are 3 possible routes but I don't have access to Google Maps.
- I have to decide which route to take at each day, based on what information (e.g., traffic, roadblocks etc) I have collected from previous days.

→ Can't **hope** to compete with the daily optimal route.

→ Compete with **best-fixed route in hindsight**.

Useful for when environment changes rapidly, data evolves unpredictably, and we have to make decisions on the fly.

Online Learning -- Mathematically

Nature chooses (possibly adversarially) the loss functions: $\ell_t(\cdot)\ \forall t \in [T]$.

For round $t = 1 \dots T$:

1. Choose action $a_t \in A$.
2. Observe either $\ell_t(\cdot)$ **[full feedback]** or only $\ell_t(a_t)$ **[bandit feedback]**.

Learner's Goal

Minimize (External) Regret

$$R(T) = \sum_{t\in[T]} \ell_t(a_t) - \min_{a^\star \in A} \sum_{t\in[T]} \ell_t(a^\star)$$

"Greedy" Approaches Fail

	$t=1$	$t=2$	• • •	$t=T$
Loss(route1)	0	1		0
Loss(route2)	1	0		1

- Every day I need to go from home → dog park.
- I know there are 3 possible routes but I don't have access to Google Maps.
- I have to decide which route to take at each day, based on what information (e.g., traffic, roadblocks etc) I have collected from previous days.

→ Can't **hope** to compete with the daily optimal route.

→ Compete with **best-fixed route in hindsight**.

Useful for when environment changes rapidly, data evolves unpredictably, and we have to make decisions on the fly.

Multiplicative Weights Update

Celebrated algorithm for **full information** settings. [Freund and Schapire, JCSS97]

Idea:

- Find correct tradeoff between randomizing strategies and past performance.
- Maintain probability distribution $\pi_t(\cdot)$ over actions in A. (Originally: $\pi_0(i) = \frac{1}{|A|}, \forall i \in A$.)
- At round t decrease $\pi_t(\cdot)$ based on loss function observed $\ell_t(\cdot)$.

Algorithm

Initialize weights: $w_0(i) = 1, \forall i \in A$.

Fix learning step size $\eta \in (0, ½)$.

For round $t = 1 \ldots T$:

1. Choose action $i_t \in A$ from $\pi_t(i) = w_t(i) / \sum_{j \in A} w_t(j)$.
2. Observe $\ell_t(\cdot) \in [0,1]$ and incur $\ell_t(i_t)$.
3. Update weights: $w_{t+1}(i) = w_t(i) \cdot (1 - \eta)^{\ell_t(i)}$

Regret

For $\eta = \sqrt{\frac{\log(|A|)}{T}}$, MWU incurs regret: $\mathbb{E}[R(T)] \leq O\left(\sqrt{T \cdot \log(|A|)}\right)$.

Multiplicative Weights Update -- Proof

Algorithm

Initialize weights: $w_0(i) = 1, \forall i \in A$.

Fix learning step size $\eta \in (0, ½)$.

For round $t = 1 \ldots T$:

1. Choose action $i_t \in A$ from $\pi_t(i) = w_t(i) / \sum_{j \in A} w_t(j)$.
2. Observe $\ell_t(\cdot) \in [0,1]$ and incur $\ell_t(i_t)$.
3. Update weights: $w_{t+1}(i) = w_t(i) \cdot (1 - \eta)^{\ell_t(i)}$.

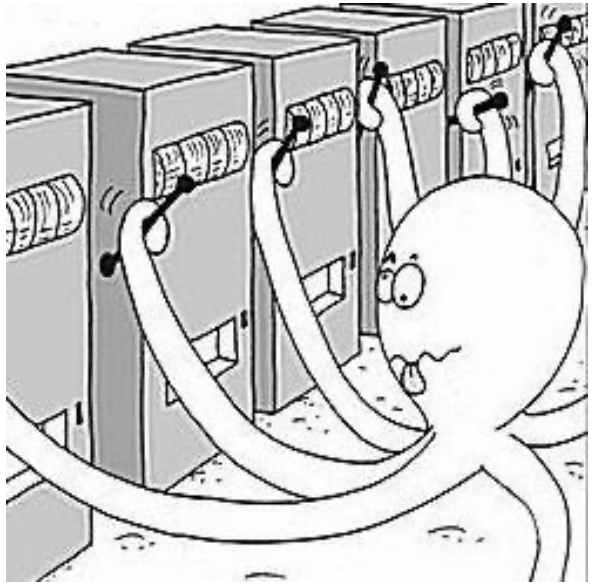
Regret

For $\eta = \sqrt{\frac{\log(|A|)}{T}}$, MWU incurs regret:

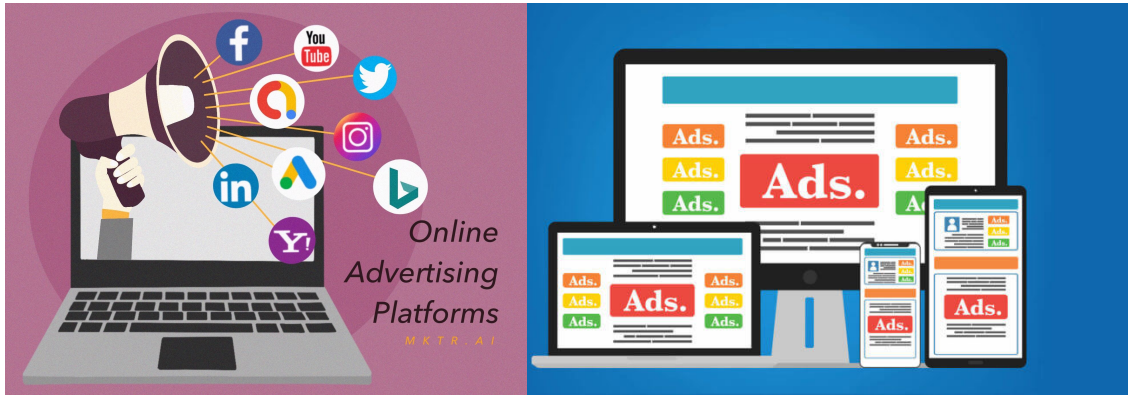
$$\mathbb{E}[R(T)] \leq O\left(\sqrt{T \cdot \log(|A|)}\right).$$

Proof on the board here: https://tinyurl.com/nwe9c4xs

Multi-Armed Bandits



Application: Online Advertising



- Set of actions (aka **arms**) A is discrete ($|A| = K$).
- Learner adaptively chooses an arm to play and observes **only its** realized reward.
- Reward distributions are **not known** a priori.

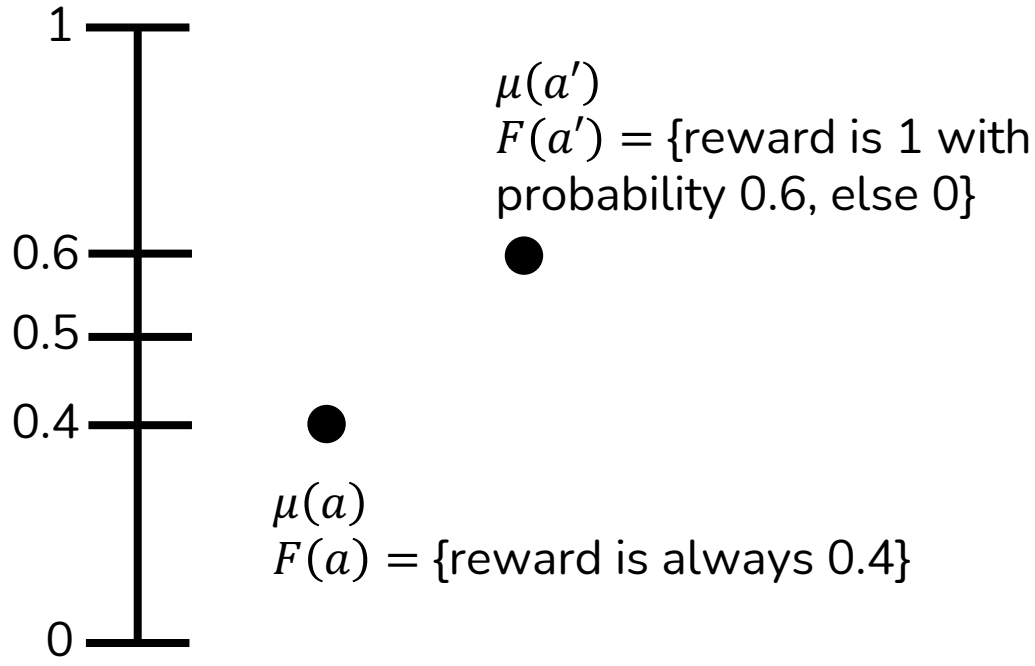
- Arms = advertisers
- Each arm has Click-Through-Rate (CTR) of getting clicked.
- Platform adaptively selects the ads to display, but ads' CTR is unknown.

Stochastic MABs

Arm $a \in A$: reward distribution $F(a)$ (support in $[0,1]$) with mean $\mu(a)$.

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

1. Learner selects arm a_t (possibly randomly).
2. Rewards realized $\forall a \in A$: $r_t(a) \sim F(a)$.
3. Learner earns and only observes reward $r_t(a_t)$.



Greedy Algorithm

1. Play each arm once.
2. Play the highest-observed-reward arm forever.

Greedy Algorithm Fails

Incurs linear regret $R(T) = \Omega(T)$

Exploration-Exploitation Tradeoff

Maintain confidence bounds for the arms' mean rewards and make decisions based on these.

e.g., stop considering an arm after it is **confidently** an underdog

Algorithms for Stochastic MAB

Arm $a \in A$: reward distribution $F(a)$ (support in $[0,1]$) with mean $\mu(a)$.

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

1. Learner selects arm a_t (possibly randomly).
2. Rewards realized $\forall a \in A$: $r_t(a) \sim F(a)$.
3. Learner earns and only observes reward $r_t(a_t)$.

estimated mean

$$UCB_t(a) = \tilde{\mu}_t(a) + \sqrt{\frac{\log\left(\frac{2KT}{\delta}\right)}{2N_t(a)}}$$

$$LCB_t(a) = \tilde{\mu}_t(a) - \sqrt{\frac{\log\left(\frac{2KT}{\delta}\right)}{2N_t(a)}}$$

times played arm a

Active Arm Elimination (AAE)

[Even-Dar, Mannor, Mansour, JMLR06]

1. Originally, $A_0 = A$.
2. Maintain adaptive confidence bounds $\forall a \in A_t$: $UCB_t(a), LCB_t(a)$.
3. Play arms in round robin fashion.
4. Eliminate arm a from A_t if $UCB_t(a) < \max_{a'} LCB_t(a')$

UCB1

[Auer, Cesa-Bianchi, Fischer, ML02]

1. Play arm with max UCB.

Adversarial MAB – EXP3

Sequence of arms' rewards: $\{r_t(a)\}_{t\in[T]}, \forall a \in A$ chosen **adversarially**.

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

1. Learner selects arm a_t (possibly randomly).
2. Rewards $\forall a \in A$: $r_t(a)$.
3. Learner earns and only observes reward $r_t(a_t)$.

Idea

Exponential weights for an unbiased estimate of the rewards.

$$R(T) \leq O\left(\sqrt{KT\log K}\right)$$

EXP3 (Exponential weights for Exploration and Exploitation)

[Auer, Cesa-Bianchi, Freund, and Schapire, FOCS95]

Parameter: $\eta \in (0, ½)$.

Initialize weights: $w_0(a) = 1, \forall a \in A$.

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

1. Learner selects arm $a_t \sim \pi_t(\cdot)$, where $\pi_t(a) = w_t(a) / \sum_{a' \in A} w_t(a')$
2. Compute unbiased loss estimates: $\forall a \in A$: $\hat{\ell}_t(a) = \frac{\ell_t(a) \cdot 1\{a = a_t\}}{\pi_t(a)}$.
3. Learner updates weights: $\forall a \in A$: $w_{t+1}(a) = w_t(a) \cdot \exp\left(-\eta \hat{\ell}_t(a)\right)$.

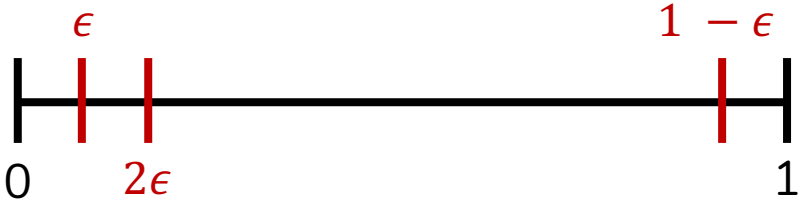
Lipschitz Bandits -- Discretization

- So far: arms' reward functions didn't have to be *correlated* and set of arms was **discrete**.
- What if $\boldsymbol{A = [0, 1]}$ and $\ell_t(\cdot)$ is unknown, **Lipschitz** and sequence $\{\ell_t(\cdot)\}_{t\in[T]}$ can be **adversarially/stochastically chosen**?

[Kleinberg, NIPS04]

→ Uniform discretization of $A = [0,1]$ in ϵ-grid.
→ Treat grid points as arms.
→ Apply standard MAB algorithms for $K = \frac{1}{\epsilon}$ arms.
→ **Discretization error** is bounded.

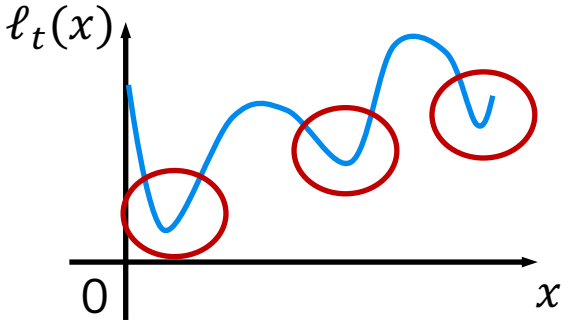
Proof on the board here: https://tinyurl.com/nwe9c4xs



$R(T) = \tilde{\Theta}(T^{2/3})$ both for stochastic and adversarial case.

- Easy implementation.
- Worst-case optimal.
- **Wasteful** for "nicer" instances.

[Kleinberg, Slivkins, Upfal, STOC08]
[Bubeck, Munos, Stoltz, Szepesvari, NIPS08]
[**Podimata** and Slivkins, COLT21]



Ideally, want to place more probes around red circles → this is where the optimal arm is.

Still worst case optimal, but (much) for nice instances.

Adaptive Discretization

Adaptively discretize the arms' space based on feedback received so far.

Between Stochastic and Adversarial

Stochastic World

- In the worst case $R(T) \leq O(\sqrt{KT})$.
- If rewards are **not stochastic** → stochastic MAB algos $R(T) = \Omega(T)$.

Adversarial World

- In the worst case $R(T) \leq O(\sqrt{KT})$.
- If rewards are **stochastic** → adversarial MAB algos: no enhanced bounds

[Lykouris, Mirrokni, Paes Leme STOC18]

Best-of-both worlds bounds

[Bubeck and Slivkins, COLT12], [Seldin and Slivkins, ICML14], [Auer, Chiang, ICML16], [Seldin, Lugosi, COLT17], [Wei, Luo, COLT18], [Zimmert, Seldin, AISTATS19]

Idea

Start with stochastic (resp. adversarial) and switch to adversarial (resp. stochastic) when it no longer holds.

MAB with corruptions

Arm $a \in A$: reward distribution $F(a)$ (support in $[0,1]$) with mean $\mu(a)$.

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

1. Learner commits to distribution $\pi_t(\cdot)$ over arms.
2. Rewards realized $\forall a \in A$: $r_t(a) \sim F(a)$.
3. Adversary corrupts $r_t(a)$ (total corruption budget: C).
4. Learner draws $a_t \sim \pi_t(\cdot)$ and observes (potentially corrupted) reward.

Multi-Layering Race

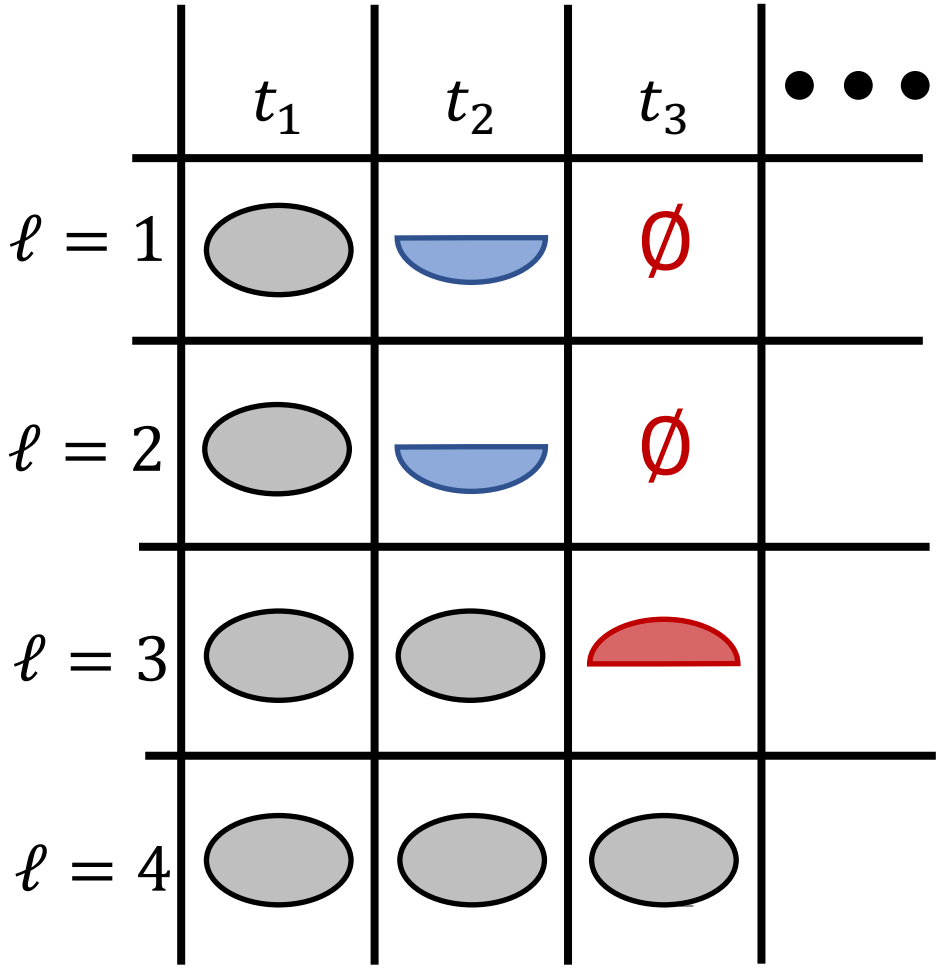
Original: [Lykouris, Mirrokni, Paes Leme STOC18]
Adaptation to continuous action spaces: [Krishnamurthy, Lykouris, **Podimata**, Schapire, STOC21

Subsampling

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

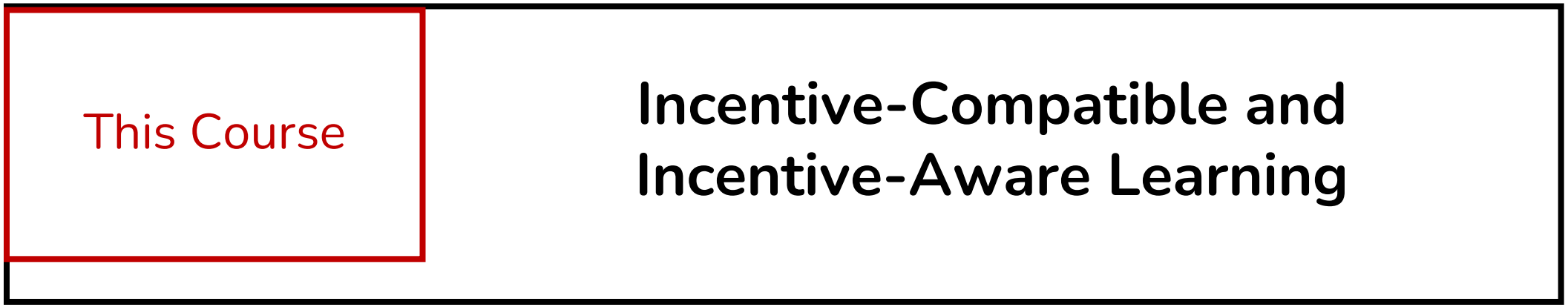
Global Eliminations

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



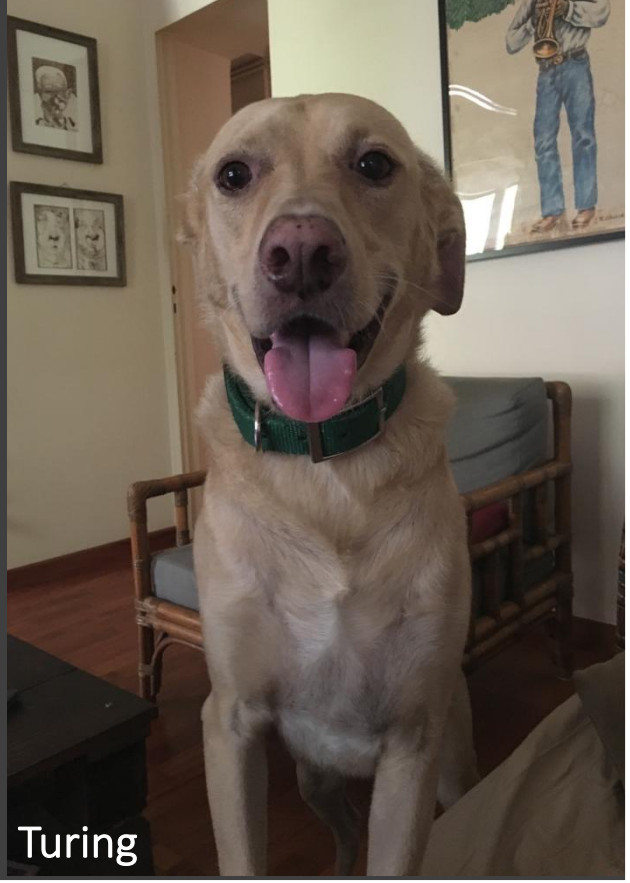
The Broader Picture

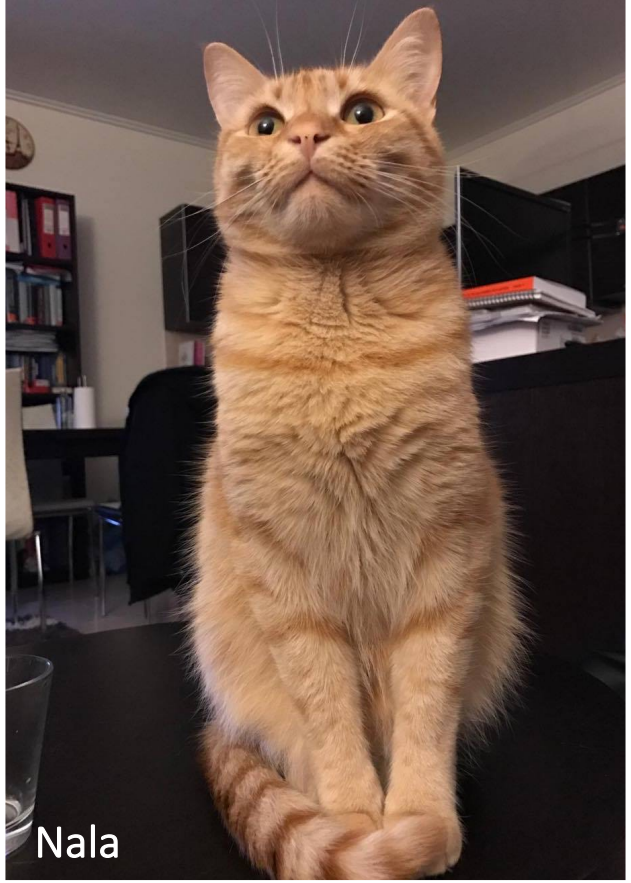
We have only scratched the surface!



If interested in learning more about Incentive-compatible ML:

[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!