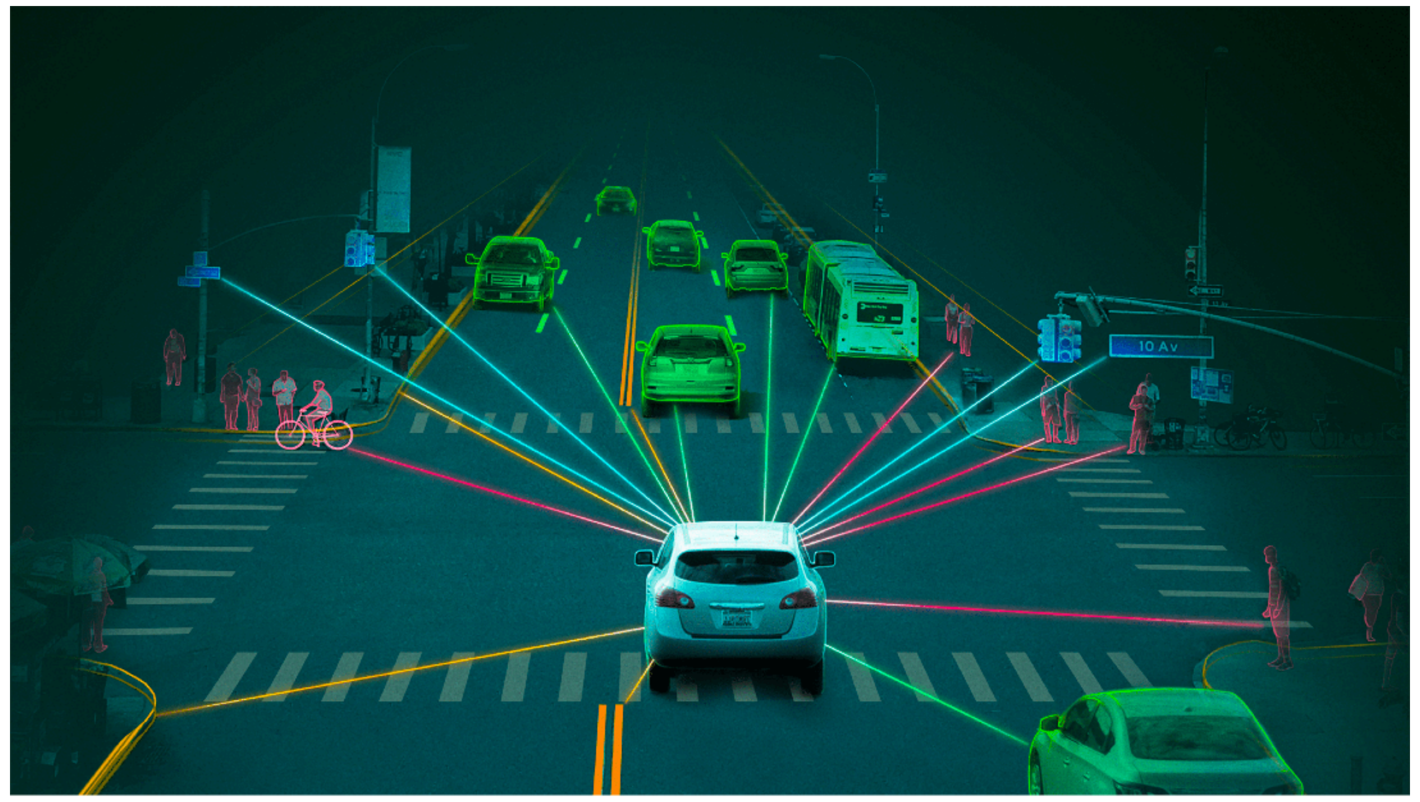
Emerging Generalization Settings
Simons Institute
September 2024

KNOW WHEN YOU KNOW

HANDLING ADVERSARIAL DATA BY ABSTAINING

Surbhi Goel
University of Pennsylvania

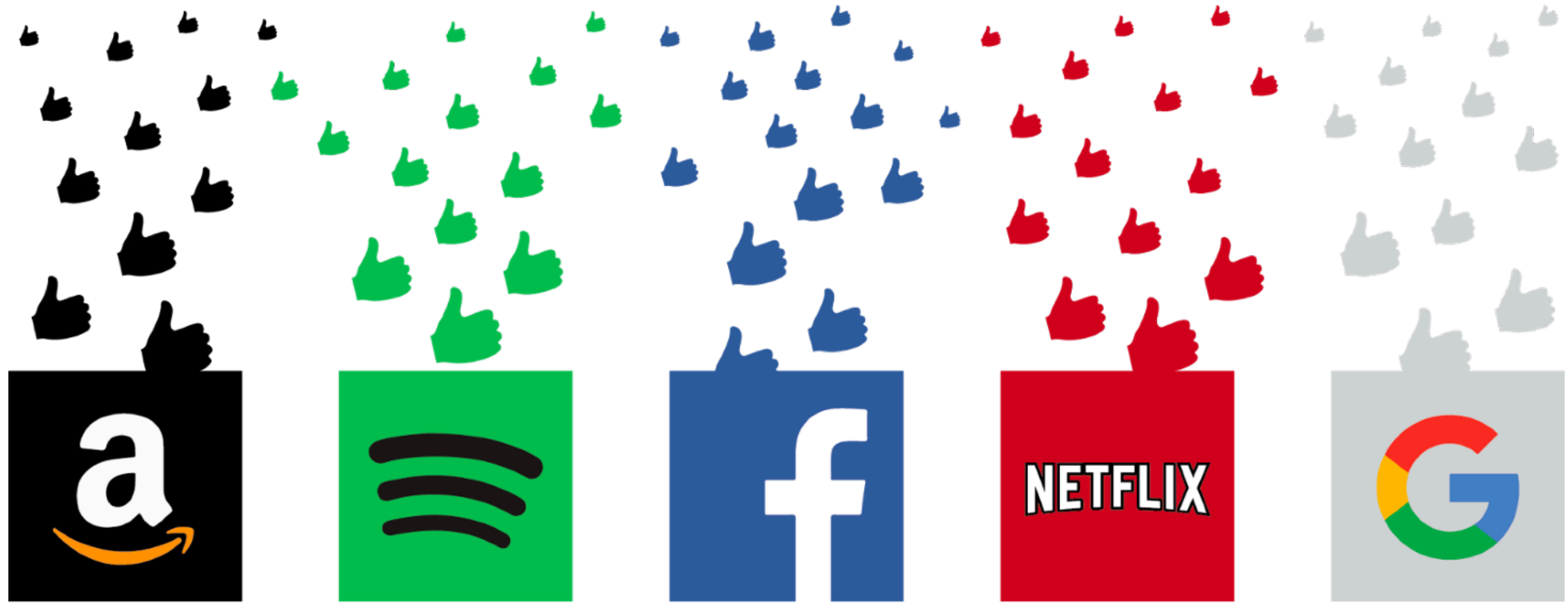
DECISION MAKING TODAY



Self-driving

Spam detection

Medical diagnosis/monitoring



Recommender systems

College admission decisions

LLM-powered ChatBots

DECISION MAKING TODAY

Report: Tesla Autopilot Involved in 736 Crashes since 2019

The self-driving technology was also implicated in 17 deaths, according to a Washington Post investigation.

BY SEBASTIAN BLANCO PUBLISHED: JUN 13, 2023

AI May Be More Prone to Errors in Image-Based Diagnoses Than Clinicians

New research indicates that AI may be more prone to making mistakes than humans in image-based medical diagnoses because of the features they use for analysis.

nages

What happens when test $\neq$ train?

NHTSA deepens its probe into Tesla collisions with stationary emergency vehicles

The agency added six more incidents since the investigation sta

ARTIFICIAL INTELLIGENCE

How AI Bias Impacts Medical Diagnosis

AI models that are good at predicting race/gender are less accurate in diagnosis.

'My Watch Thinks I'm Dead'

Dispatchers for 911 are being inundated with false, automated distress calls from Apple devices owned by skiers who are very much alive.

Artificial intelligence + Add to myFT

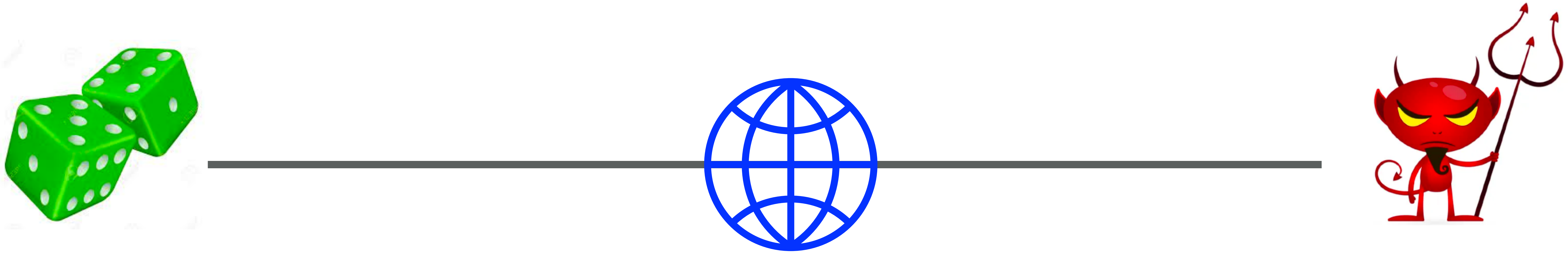
Hackers 'jailbreak' powerful AI models in global effort to highlight flaws

Experts join forces in search for vulnerabilities in large language models made by OpenAI, Google and Elon Musk's xAI

GM's Cruise recalling 950 driverless cars after pedestrian dragged in crash

Challenge: Errors on adversarial or out-of-distribution (OOD) data can be very costly

STOCHASTIC VS WORST-CASE SEQUENTIAL PREDICTION



stochastic sequences

Data is drawn from some unknown fixed distribution at each round

Easy to get strong guarantees

Too simplistic to model real world

worst-case adversarial sequences

Data is generated by an adversary who knows the algorithm and history

Too pessimistic to get any guarantees

Very robust to changes/attacks

Real world is not worst-case, but somewhere in between two extremes

Can we achieve strong guarantees while **handling** any amount of adversarial/OOD data?

Potential Fix:

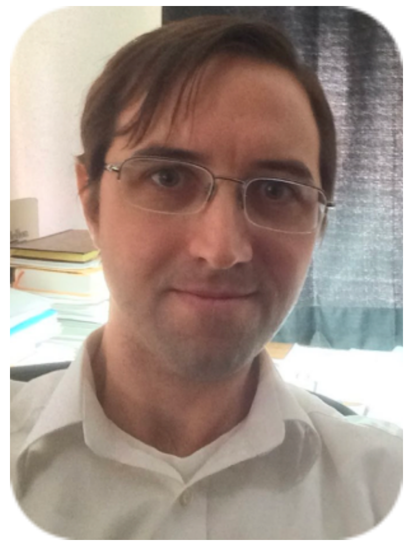
- Allow the model to **abstain** on adversarial/OOD data

Can use human-in-the-loop to label unsure examples in high stakes applications

Today:

- New **framework** that incorporates abstention in sequential prediction
- **Algorithms** in this framework that achieve strong guarantees

Joint work with:



Steve Hanneke
Purdue

Shay Moran
Technion & Google Research

Abhishek Shetty
UC Berkeley

Paper:

Acknowledgement: *Thodoris Lykouris (MIT) and Adam Tauman Kalai (OpenAI)*

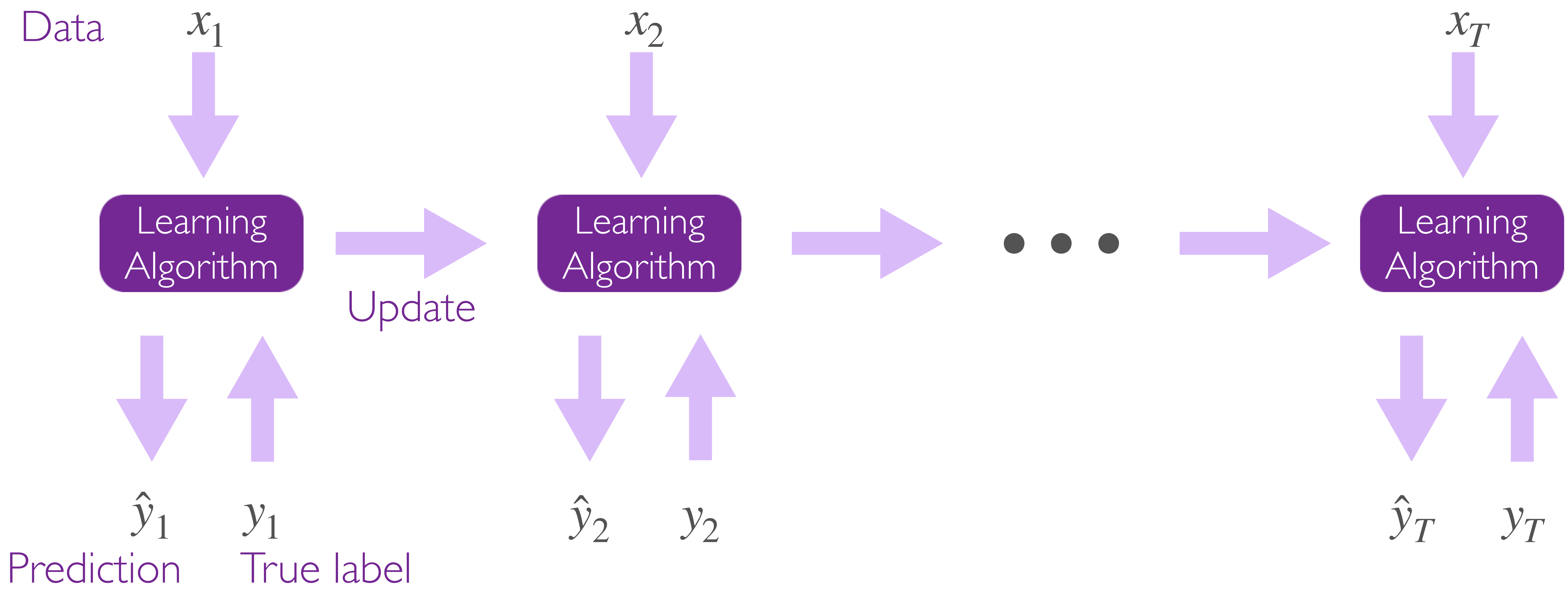
Builds on: *[Goldwasser-Kalai-Kalai-Montasser'20] (will discuss after)*

Part I:

Sequential prediction with Abstentions

SEQUENTIAL PREDICTION

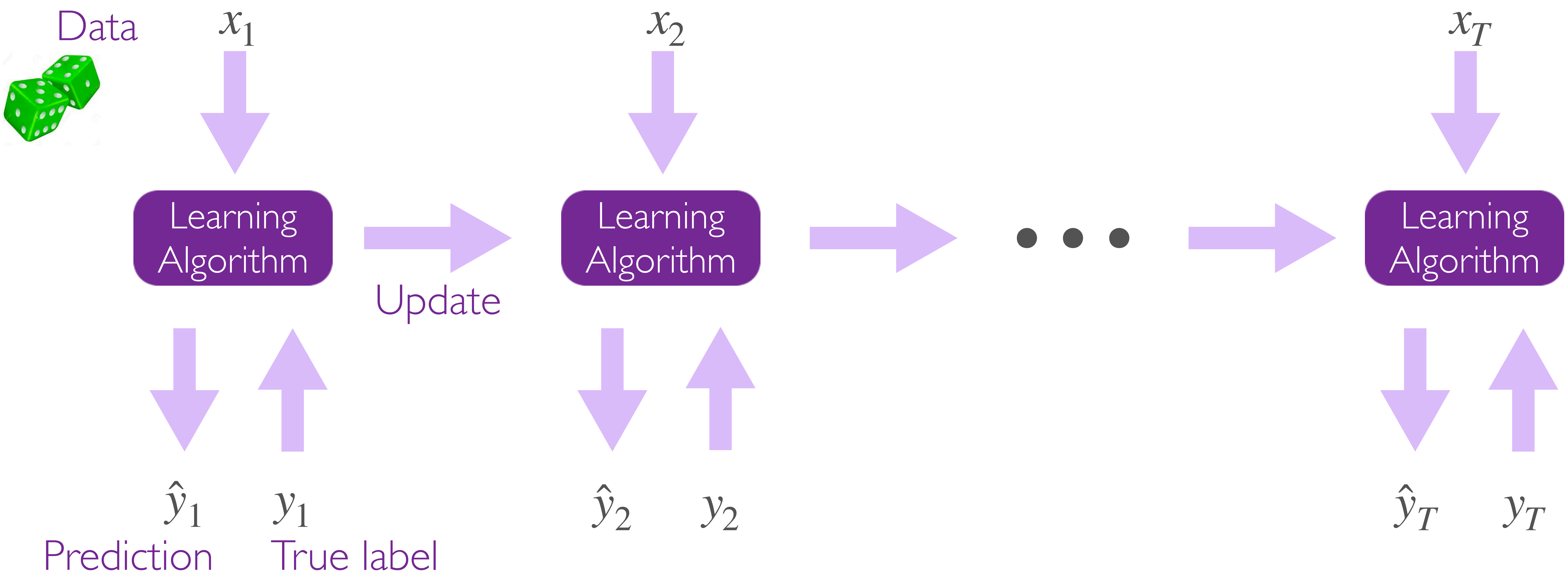
Assume true label follows some fixed unknown function from a binary-valued class $\mathscr{F}$



Goal: Minimize regret/error - total number of mistakes made by the algorithm

As $T \to \infty$, average number of mistakes $\to 0$

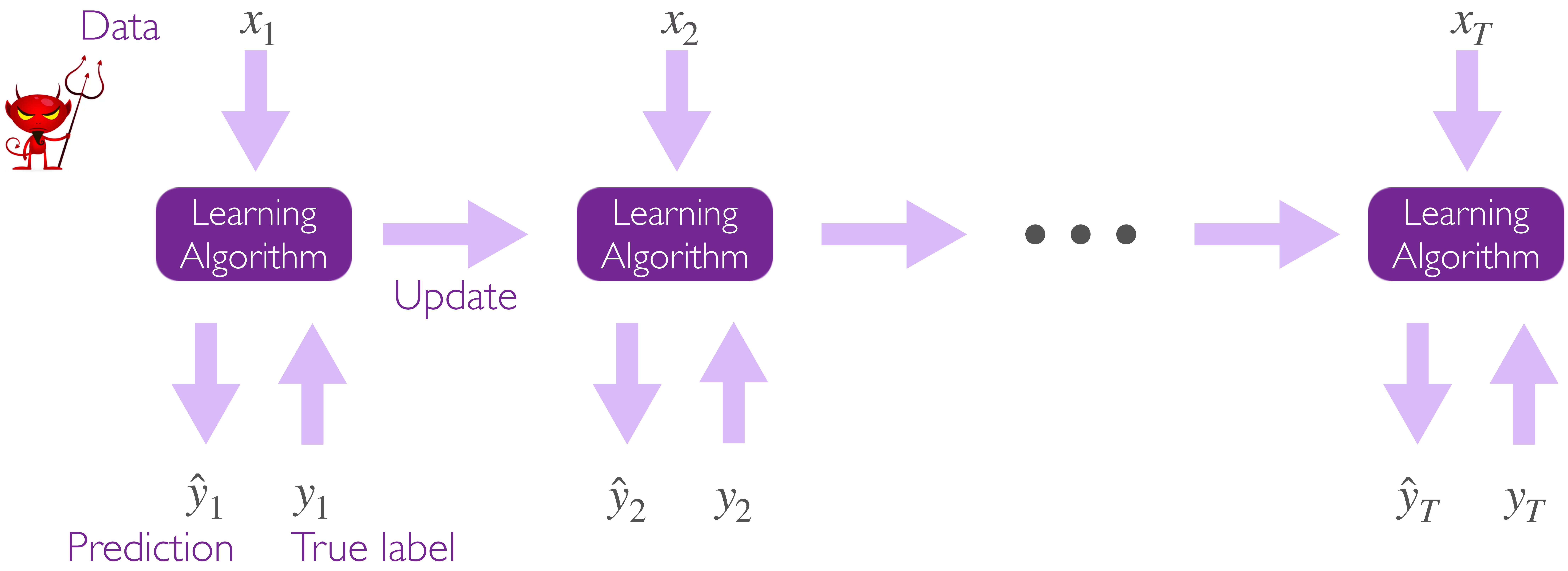
SEQUENTIAL PREDICTION - STOCHASTIC



If data is purely stochastic then regret depends only on the VC dimension of $\mathscr{F}$

Complexity notion for offline learning

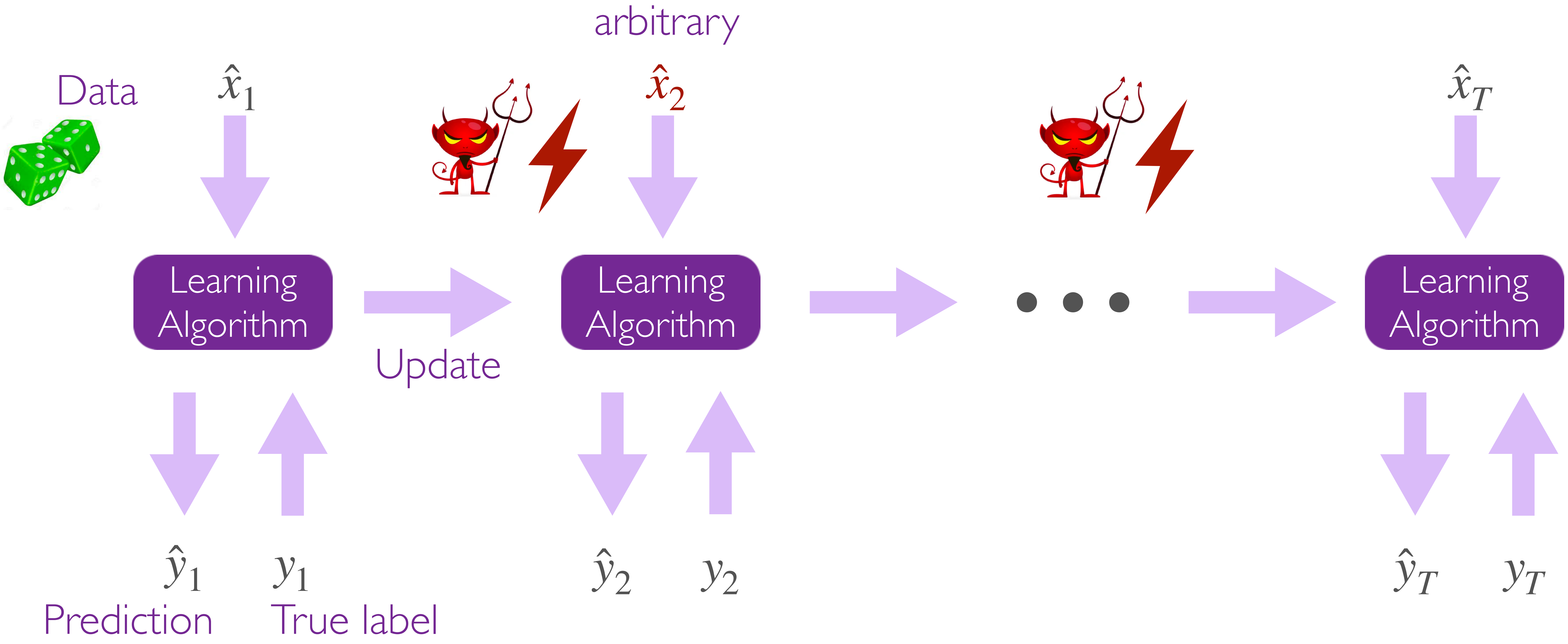
SEQUENTIAL PREDICTION - ADVERSARIAL



If data is fully adversarial then regret depends on the Littlestone dimension of $\mathcal{F}$

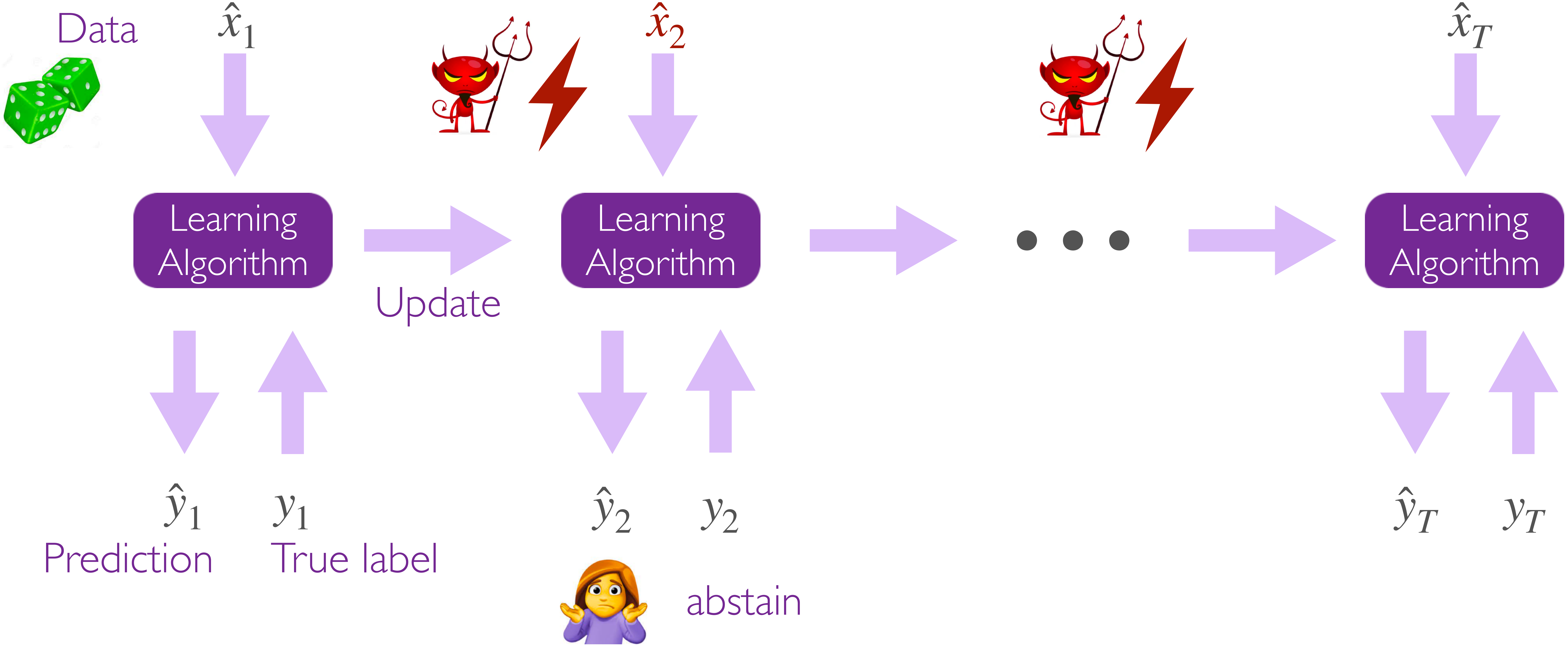
Littlestone dimension can be infinite even when VC dimension is 1

SEQUENTIAL PREDICTION WITH ADVERSARIAL INJECTIONS



Sequence is stochastic, but the adversary can decide at each time if they want to inject an adversarial input (*before the stochastic input is drawn*)

SEQUENTIAL PREDICTION WITH ADVERSARIAL INJECTIONS AND ABSTENTIONS



The learner gets an extra option to abstain from predicting ($\hat{y}_t \in \{0,1,\underbrace{\perp}_{\text{abstain}}\}$)

SEQUENTIAL PREDICTION WITH ADVERSARIAL INJECTIONS AND ABSTENTIONS

Protocol 1 Sequential Prediction with Adversarial Injections and Abstentions

Adversary (or nature) initially selects distribution $\mathfrak{D} \in \Delta(\mathcal{X})$ and $f^\star \in \mathcal{F}$. The learner does not have access to $f^\star$. The learner may or may not have access to $\mathfrak{D}$.

for $t = 1, \ldots, T$ **do**

Adversary decides whether to inject an adversarial input in this the round ($c_t = 1$) or not ($c_t = 0$).

if $c_t = 1$ **then** Adversary selects any $\hat{x}_t \in \mathcal{X}$

else Nature selects $\hat{x}_t \sim \mathfrak{D}$

Learner receives $\hat{x}_t$ and outputs $\hat{y}_t \in \{0, 1, \perp\}$ where $\perp$ implies that the learner abstains.

Learner receives clean label $y_t = f^\star(\hat{x}_t)$. *May not receive label when abstaining*

- **Realizable** model, fixed $f^\star$ at the start *(can handle adaptive $f^\star$ in some cases)*
- **Insertion-only** adversarial examples before seeing the i.i.d. example
- Same as stochastic if **no** adversarial injections and same as fully adversarial if **all** examples are injected

SEQUENTIAL PREDICTION WITH ADVERSARIAL INJECTIONS AND ABSTENTIONS

Simultaneously want to minimize

- **Incorrect prediction** whenever the learner decides to predict
- **Incorrect abstentions** whenever the learner abstains on non-adversarial data

$$\text{Error} := \underbrace{\sum_{t=1}^{T} \mathbb{1}[\hat{y}_t = 1 - f^\star(\hat{x}_t)]}_{\text{MisclassificationError}} + \underbrace{\sum_{t=1}^{T} \mathbb{1}[c_t = 0 \wedge \hat{y}_t = \perp]}_{\text{AbstentionError}}$$

Abstentions on adversarial examples are free

Error does not need to scale with the number of adversarial injections as long as we predict with certainty

SEQUENTIAL PREDICTION WITH ADVERSARIAL INJECTIONS AND ABSTENTIONS

Why is it challenging?

- Adversary can add **any** number of adversarial injections
- Learner **does not know** which examples in the history were injected, and hence can't compute its own loss
- Adversary could insert examples that cause downstream errors later
- Need to known when you know

Note that we do not need to solve one example adversarial detection (which may be impossible) as long as we can predict correctly on the example

CONNECTIONS TO OTHER FRAMEWORKS

- **Abstention-based learning:**
  - **Offline classification:**
    - Fast rates via abstenion [Chow'70, Herbei-Wegkamp'06, Bousquet-Zhivotovskiy'20]
    - **Transductive robust learning** [Goldwasser-Kalai-Kalai-Montasser'20, Kalai-Kanade'21]
    - Testable Distribution shift [Klivans-Stavropoulos-Vasilyan'24]
- **Online classification:**
    - KWIK *(know what it knows)* [Li-Littman-Walsh'08, Sayedi et al.'10, Zhang-Chaudhuri'16]
    - Fast rates via abstention [Neu-Zhivotovskiy'20]

Mostly focus on fully adversarial or purely stochastic setting

TRANSDUCTIVE ROBUST LEARNING

[Goldwasser-Kalai-Kalai-Montasser'20]

Clean labelled training data

$(x_1, y_1), \ldots, (x_n, y_n)$

Corrupted unlabelled test data

$\hat{x}_1, \ldots, \hat{x}_m$

Want to output prediction set $S \subseteq [m]$ and classifier h such that you minimize both

Incorrect predictions: $\frac{1}{m} \sum_{i \in S} 1[h(\hat{x}_i) \neq \hat{y}_i]$

Incorrect abstentions: $\frac{1}{m} \sum_{i \in [m] \setminus S} 1[\hat{x}_i$ **was not corrupted**$]$

Our setup is an online version of this

CONNECTIONS TO OTHER FRAMEWORKS

- **Beyond-worst case:**
  - Assumption on adversary such as smoothed adversary [Rakhlin-Sridharan-Tewari'11, Haghtalab-Roughgarden-Shetty'20, …]
  - Assumption on future sequences such as predictable sequences [Rakhlin-Sridharan'13], learning with hints [Bhaskara-Cutkosky-Kumar-Purohi'13]
- **Adversarially robust learning:**
  - Test-time attacks [Szegedy et al.'13, Biggio et al.'13, Goodfellow et al.'15, Feige et al.'18, Attias et al.'19, Montasser et al.'19,20,21,22]
  - Training time attacks [Valiant'85, Kearns and Li'93, Bshouty et al.'02, Biggio et al.'12, Awasthi et al.'17, Steinhardt et al.'17, Shafahi et al.'18, Levine and Feizi'21, Gao et al.'21, Hanneke et al.'22, Balcan et al.'22]

We need to handle both test time and training time attacks

CONNECTIONS TO OTHER FRAMEWORKS

- **Testable Learning:** *(see Arsen's Talk at Meet the Fellows)*
  - Our framework can be viewed as an online version of testable learning [Rubinfeld-Vasilyan'20]
  - Testable learning tests an assumption on the data, and must succeed when the assumption is satisfied
  - Abstention acts as the tester

$$\text{Error} := \underbrace{\sum_{t=1}^{T} \mathbb{1}[\hat{y}_t = 1 - f^\star(\hat{x}_t)]}_{\text{MisclassificationError}} + \underbrace{\sum_{t=1}^{T} \mathbb{1}[c_t = 0 \wedge \hat{y}_t = \perp]}_{\text{AbstentionError}}$$

Soundness *Completeness*

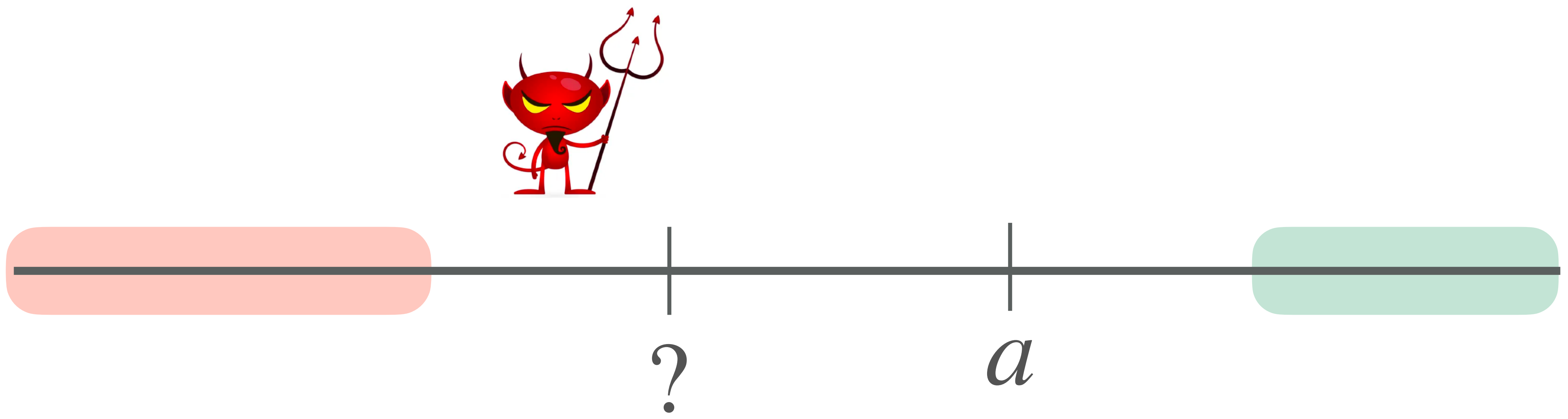
Part II:

Robust Algorithms via Abstention

EXAMPLE: LINEAR THRESHOLDS

Class of linear thresholds in one dimension

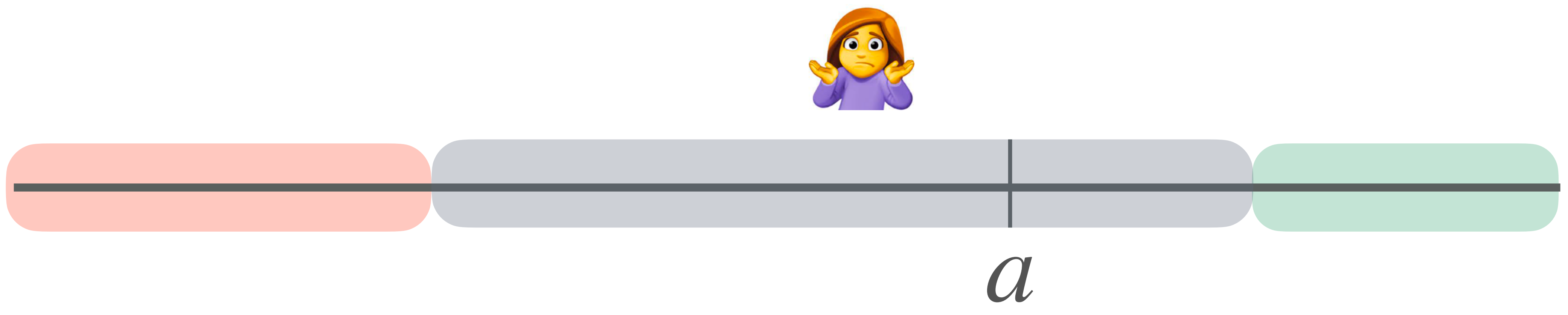
$$f_a(x) = \begin{cases} 1 & \text{if } x \geq a \\ 0 & \text{otherwise.} \end{cases}$$



- The adversary chooses a random a
- The adversary can inject a random point between the closest seen positive and negative example so far
- If the learner must predict, it will make a mistake on this with probability $1/2$, leading to $T/2$ mistakes in expectation

EXAMPLE: LINEAR THRESHOLDS

$$f_a(x) = \begin{cases} 1 & \text{if } x \geq a \\ 0 & \text{otherwise.} \end{cases}$$

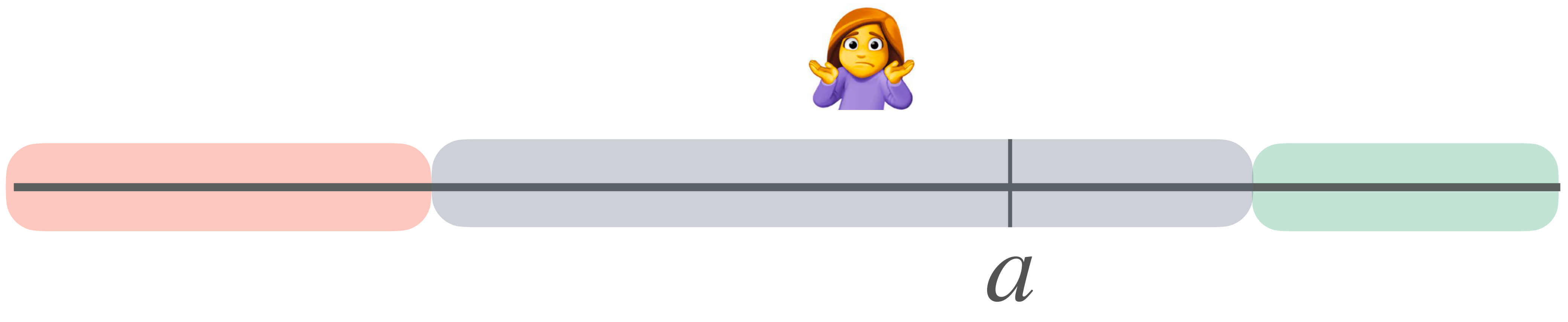


Algorithm: Learner can instead abstain between the closest positive and negative example, and predict everywhere else

$$\text{Error} := \underbrace{\cancel{\sum_{t=1}^{T} \mathbb{1}[\hat{y}_t = 1 - f^\star(\hat{x}_t)]}}_{\text{MisclassificationError}} + \boxed{\underbrace{\sum_{t=1}^{T} \mathbb{1}[c_t = 0 \wedge \hat{y}_t = \perp]}_{\text{AbstentionError}}}$$

EXAMPLE: LINEAR THRESHOLDS

$$f_a(x) = \begin{cases} 1 & \text{if } x \geq a \\ 0 & \text{otherwise.} \end{cases}$$



After seeing $t-1$ i.i.d. examples, the probability of a new i.i.d. example falling in between the closest positive and negative is $\leq 1/t$ *Exchangeability argument*

Total abstentions on i.i.d. examples $\leq \sum_{t=1}^{T} \frac{1}{t} \leq 2 \log T$

$$\text{Error} := \underbrace{\sum_{t=1}^{T} \mathbb{1}[\hat{y}_t = 1 - f^\star(\hat{x}_t)]}_{\text{MisclassificationError}} + \underbrace{\sum_{t=1}^{T} \mathbb{1}[c_t = 0 \wedge \hat{y}_t = \perp]}_{\text{AbstentionError}} \qquad \leq 2 \log T$$

Can extend to non-oblivious adversary since adversarial injections only reduce the probability of abstention

ABSTAINING WHENEVER UNCERTAIN

- **Disagreement-based learning:** Abstain on all points in the disagreement region* and predict according to consistent hypothesis everywhere else *(common approach in active learning and perfect selective classification)*

- Strategy always gets 0 misclassification error

- Abstention error is well-understood and quantified by the **star number** of the hypothesis class [Hanneke'16]

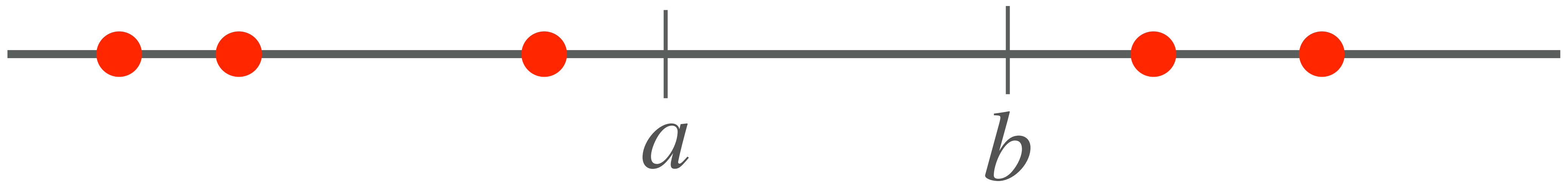
Star number is infinite for most hypothesis class of interest!

**disagreement region is the subset of the input space on which two different hypotheses disagree on the label*

EXAMPLE: INTERVALS

Class of intervals in one dimension

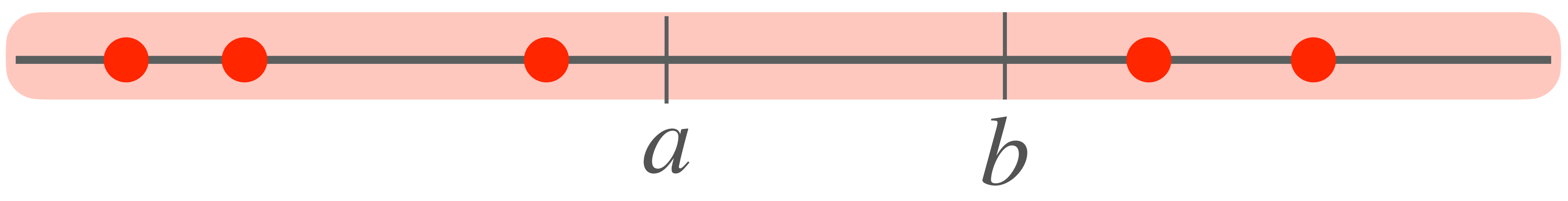
$$f_{a,b}(x) = \begin{cases} 1 & \text{if } a \leq x \leq b \\ 0 & \text{otherwise.} \end{cases}$$



- Suppose the i.i.d. distribution has very low mass on the positive part and we have only seen negative examples so far
- Then a disagreement-based learner will abstain on every example since every example is in the disagreement region
- However, it is better to predict 0 everywhere

EXAMPLE: INTERVALS

$$f_{a,b}(x) = \begin{cases} 1 & \text{if } a \leq x \leq b \\ 0 & \text{otherwise.} \end{cases}$$



Algorithm: Learner predicts negative (resp. positive) if the closest labelled examples to the left and right are both negative (resp. positive), else abstains.

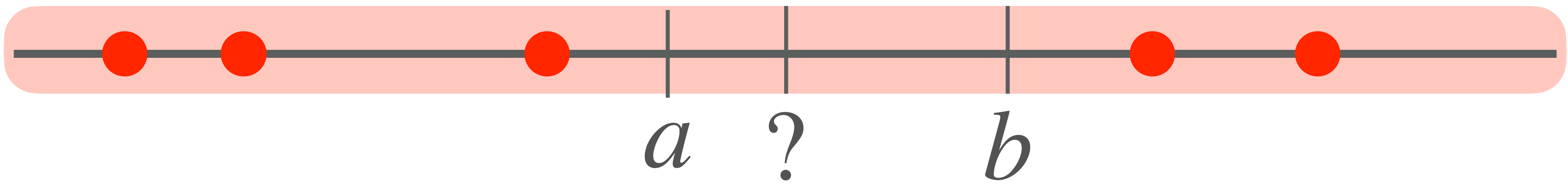
$$\text{Error} := \underbrace{\sum_{t=1}^{T} \mathbb{1}[\hat{y}_t = 1 - f^\star(\hat{x}_t)]}_{\text{MisclassificationError}} + \underbrace{\sum_{t=1}^{T} \mathbb{1}[c_t = 0 \wedge \hat{y}_t = \perp]}_{\text{AbstentionError}}$$

$$\neq 0$$

EXAMPLE: INTERVALS

$$f_{a,b}(x) = \begin{cases} 1 & \text{if } a \le x \le b \\ 0 & \text{otherwise.} \end{cases}$$

Algorithm: Learner predicts negative (resp. positive) if the closest labelled examples to the left and right are negative (resp. positive), else abstains.

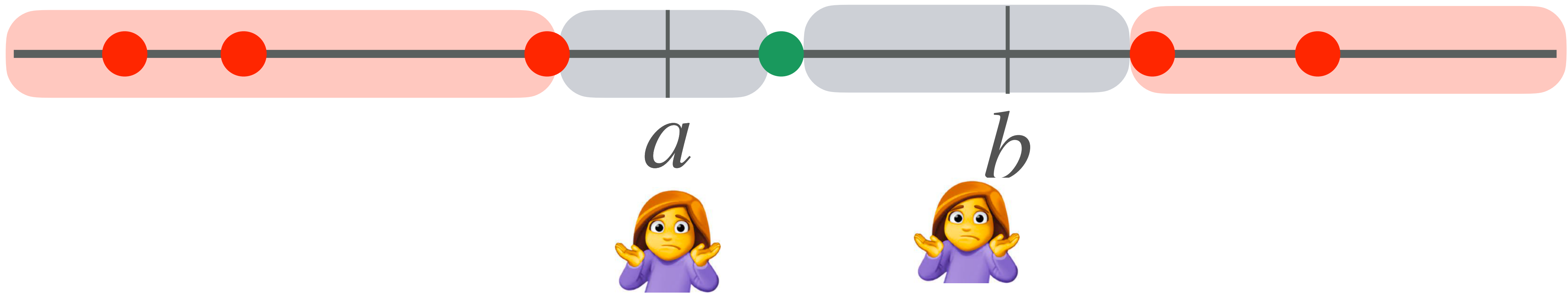


- If the algorithm makes a mistake, then it learns a positive label
- The problem reduces to two thresholds, and the algorithm reduces to disagreement-based learner for thresholds
- Can only make one misclassification mistake and at most $O(\log T)$ abstentions on i.i.d. examples

EXAMPLE: INTERVALS

$$f_{a,b}(x) = \begin{cases} 1 & \text{if } a \leq x \leq b \\ 0 & \text{otherwise.} \end{cases}$$

Algorithm: Learner predicts negative (resp. positive) if the closest labelled examples to the left and right are negative (resp. positive), else abstains.



- If the algorithm makes a mistake, then it learns a positive label
- The problem reduces to two thresholds, and the algorithm reduces to disagreement-based learner for thresholds
- Can only make one misclassification mistake and at most $O(\log T)$ abstentions on i.i.d. examples

Make mistakes as long as they help with learning!

HIGHER-ORDER DISAGREEMENT-BASED LEARNER

Theorem:
Assuming access to the marginal distribution $\mathscr{D}$ over the i.i.d. inputs, there is an algorithm that for any class with VC dimension d achieves,

$$\mathbb{E}[\text{MisclassificationError}] \le d^2 \log T$$
$$\mathbb{E}[\text{AbstentionError}] \le 6d$$

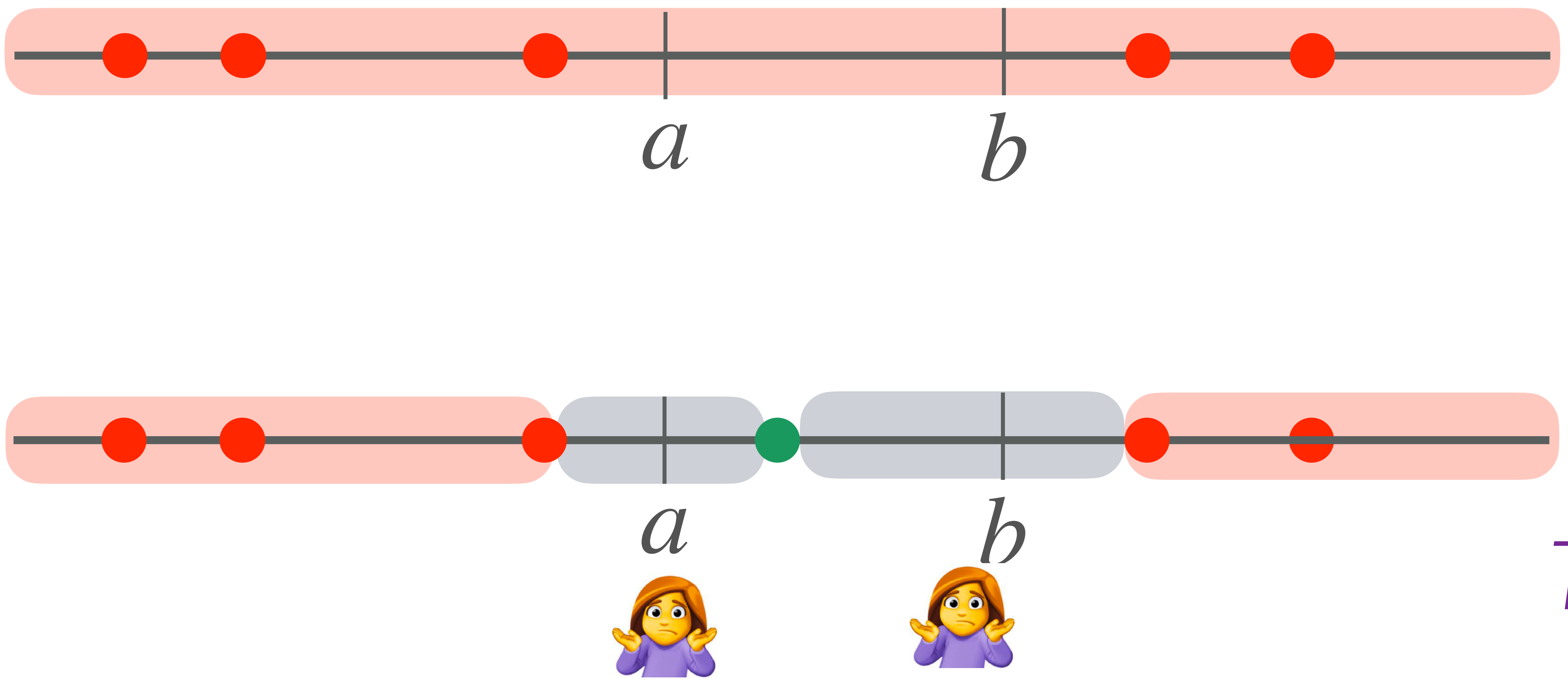
Can improve to $\tilde{O}(d \log T)$ [Narayanan]

Works even when $f^\star$ is adaptively decided as long as realizability holds
Can also allow for no labels when abstaining

HIGHER-ORDER DISAGREEMENT

Goal: You want to predict in a way that mistakes will help you

Recall: In the interval example, we predicted negative, since if we made an error we would gain information, and reduce the "dimensionality" of the problem



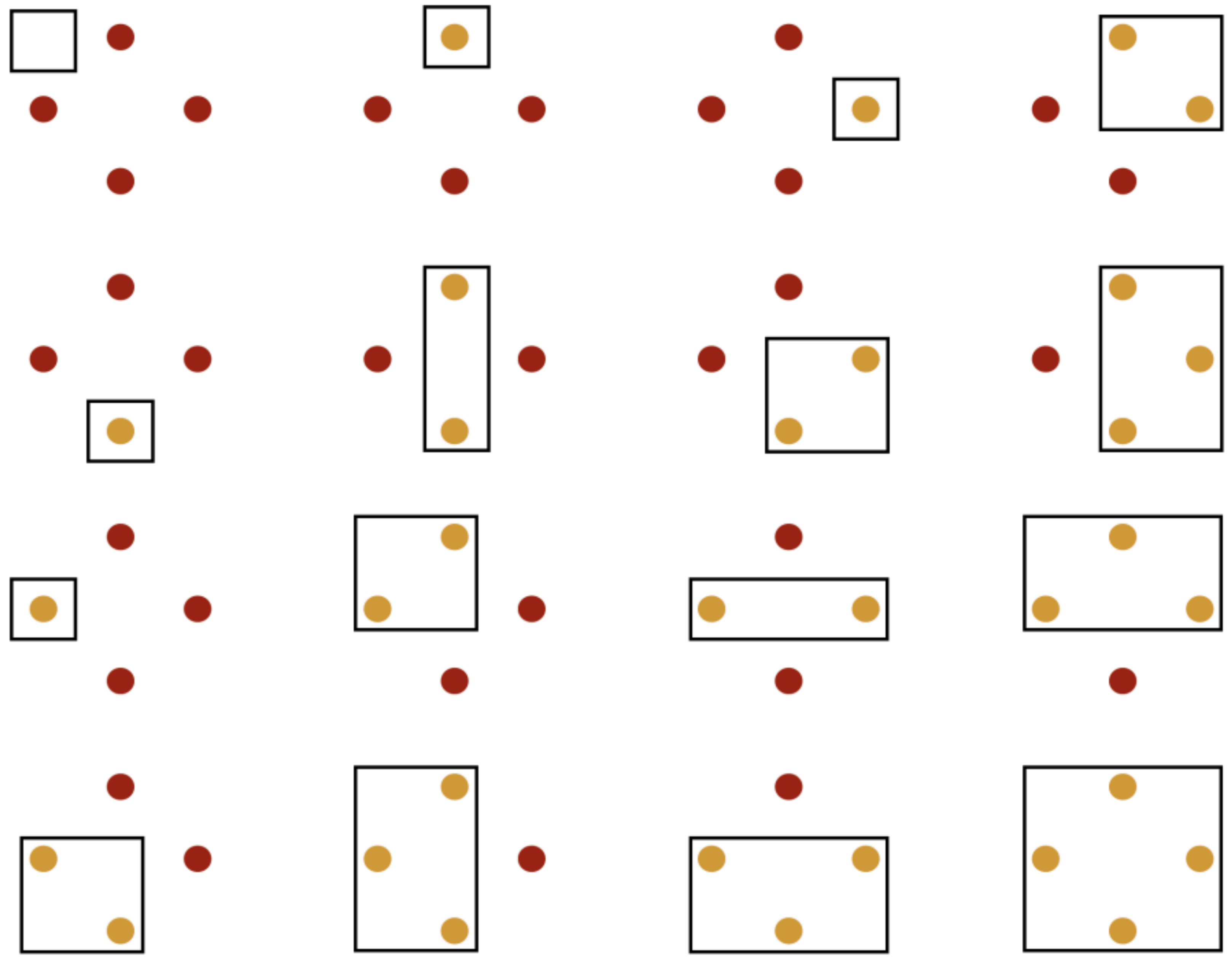
Interval (VC dim 2)

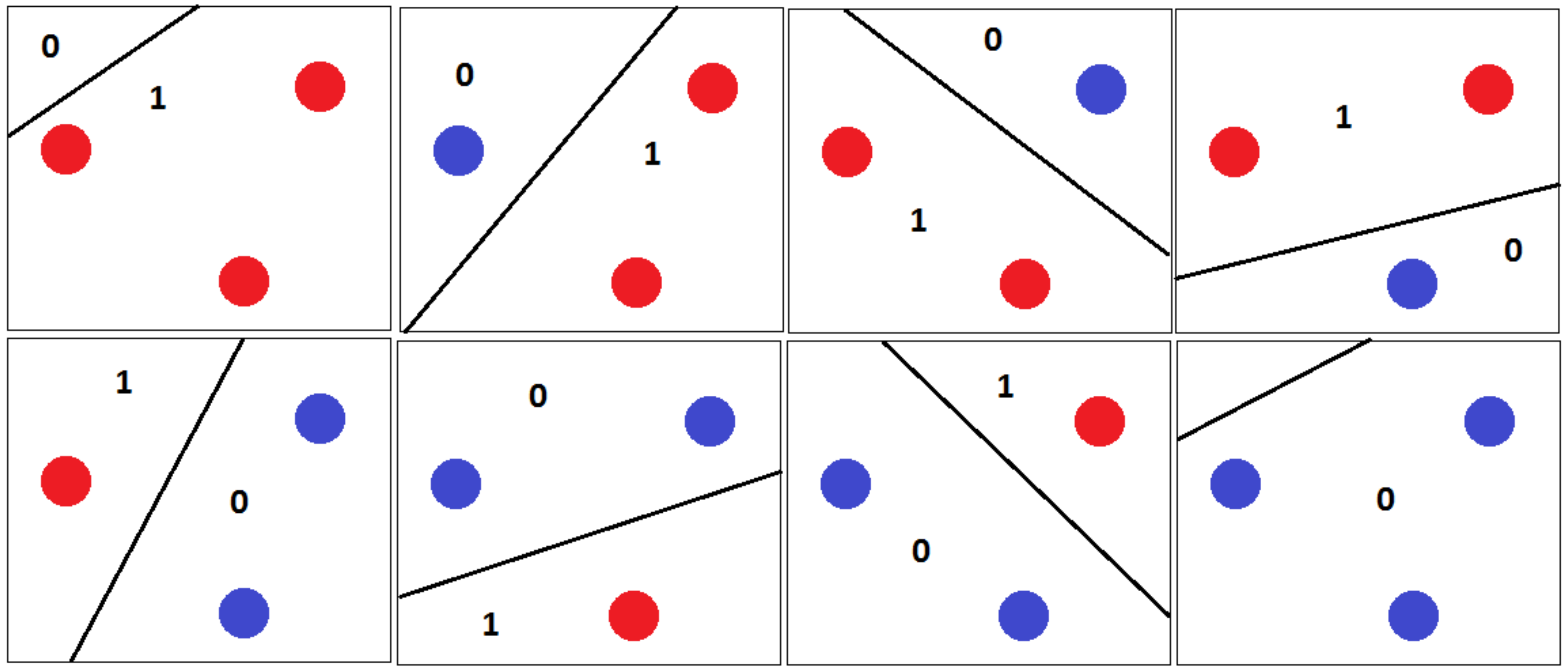
Threshold (VC dim 1)

HIGHER-ORDER DISAGREEMENT

Key idea: Use probability of shattering as an estimate of "dimensionality"

Recall: A set of size k is shattered by a function class $\mathscr{F}$ if for all possible labelings in $\{0,1\}^k$ of the set of points, there exists a $f \in \mathscr{F}$ that exactly matches it





HIGHER-ORDER DISAGREEMENT

Key idea: Use probability of shattering as an estimate of "dimensionality"

Higher-order disagreement based on shattering: [Hanneke'09,12]

At level k, the shattering probability is defined as the probability that a random set of k points drawn from the distribution is shattered by the function class

$$\rho_k(\mathcal{F}) = \Pr_{x_1,\ldots,x_k \sim \mathfrak{D}^{\otimes k}} \left[\{x_1, \ldots, x_k\} \text{ is shattered by } \mathcal{F}\right]$$

- ρ_1 is the density of the disagreement region over the distribution $\mathfrak{D}$
- $\rho_k \geq \rho_{k+1}$ for all k
- ρ_{d+1} is 0 since no $d+1$ points can be shattered (*by definition of VC dimension*)

HIGHER-ORDER DISAGREEMENT-BASED LEARNER

- Set current level $k = d$
- To make a prediction on $\hat{x}_t$:
  - Compute the k-shattering probability for the current version space with restriction on $\hat{x}_t$ being labelled 0 and 1
  - If both quantities are large, we abstain
  - Else we predict according to the larger one

$$\min\left\{\rho_k\left(\mathcal{F}_t^{\hat{x}_t\to 1}\right), \rho_k\left(\mathcal{F}_t^{\hat{x}_t\to 0}\right)\right\} \geq 0.6\rho_k\left(\mathcal{F}_t\right)$$

  - If both quantities are small ($\approx T^{-k}$), go down a level
  - Update version space after receiving label
- Once at level 0, then abstain on any point in the disagreement region

WHY DOES IT WORK?

- To make a prediction on $\hat{x}_t$:
  - Compute the k-shattering probability for the current version space with restriction on $\hat{x}_t$ being labelled 0 and 1
  - If both quantities are large, we abstain
  - Else we predict according to the larger one
  - If both quantities are small ($\approx T^{-k}$), go down a level
  - Update version space after receiving label

$$\boxed{\min\left\{\rho_k\left(\mathcal{F}_t^{\hat{x}_t\to 1}\right), \rho_k\left(\mathcal{F}_t^{\hat{x}_t\to 0}\right)\right\} \geq 0.6\rho_k\left(\mathcal{F}_t\right)}$$

For i.i.d. points, these quantities can't both be large very often, so low abstentions

(next slide)

We reduce ρ_k by a constant factor (0.6) at every mistake, so low misclassifications

Similar to halving for finite sized classes but we do it for each level

HIGHER-ORDER DISAGREEMENT: KEY LEMMA

Lemma:

Given any $\eta > 1/2$, for an i.i.d. example, the probability that both quantities are large is bounded by above as follows:

$$\Pr_{x \sim \mathfrak{D}}\left[\rho_k\left(\mathcal{F}^{x\to 1}\right) + \rho_k\left(\mathcal{F}^{x\to 0}\right) \geq 2\eta\rho_k(\mathcal{F})\right] \leq \frac{1}{2\eta - 1} \cdot \frac{\rho_{k+1}(\mathcal{F})}{\rho_k(\mathcal{F})}.$$

$\eta = 0.6$

High-level intuition:

- Consider a set of k i.i.d. points, if they are shattered by both $\mathscr{F}^{x\to 0}$ and $\mathscr{F}^{x\to 1}$, then we have a set of $k+1$ i.i.d. points that are shattered
- So this can not both happen too often if $\rho_{k+1}(\mathscr{F})$ is small

LIMITATIONS

- Requires access to the exact marginal distribution $\mathscr{D}$
- To simulate with unlabelled samples, naive implementation would need $T^{\Omega(d)}$ samples, which is too high for large VC dimension
- Checking shattering can be computationally inefficient

However, it wasn't even clear we could get VC-like guarantees!

WITHOUT ACCESS TO THE DISTRIBUTION

Theorem:

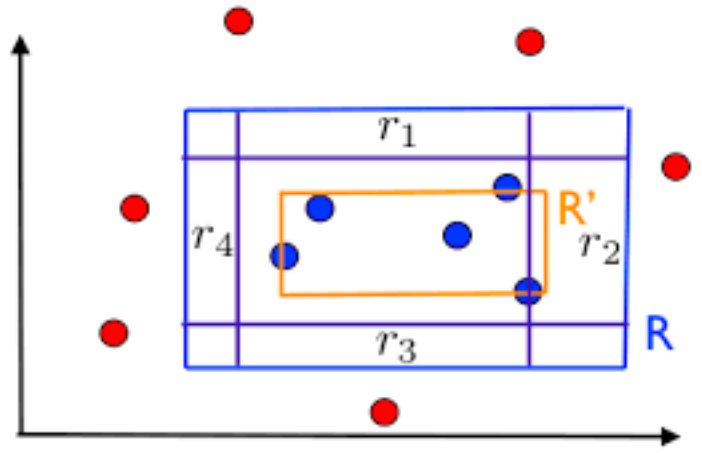
Without any access to the marginal distribution, for VC dimension 1 classes, there is an algorithm that achieves,

$$\mathbb{E}[\text{MisclassificationError}] \leq O(\sqrt{T \log T})$$

$$\mathbb{E}[\text{AbstentionError}] \leq O(\sqrt{T \log T})$$

Note: *The error scales as* $\sqrt{T \log T}$ *compared to* $\log T$ *in the known marginals case*

WITHOUT ACCESS TO THE DISTRIBUTION



Theorem:

Without any access to the marginal distribution, for axis-aligned rectangles in dimension d, there is an algorithm that achieves,

$$\mathbb{E}[\text{MisclassificationError}] \leq O(\sqrt{dT \log T})$$

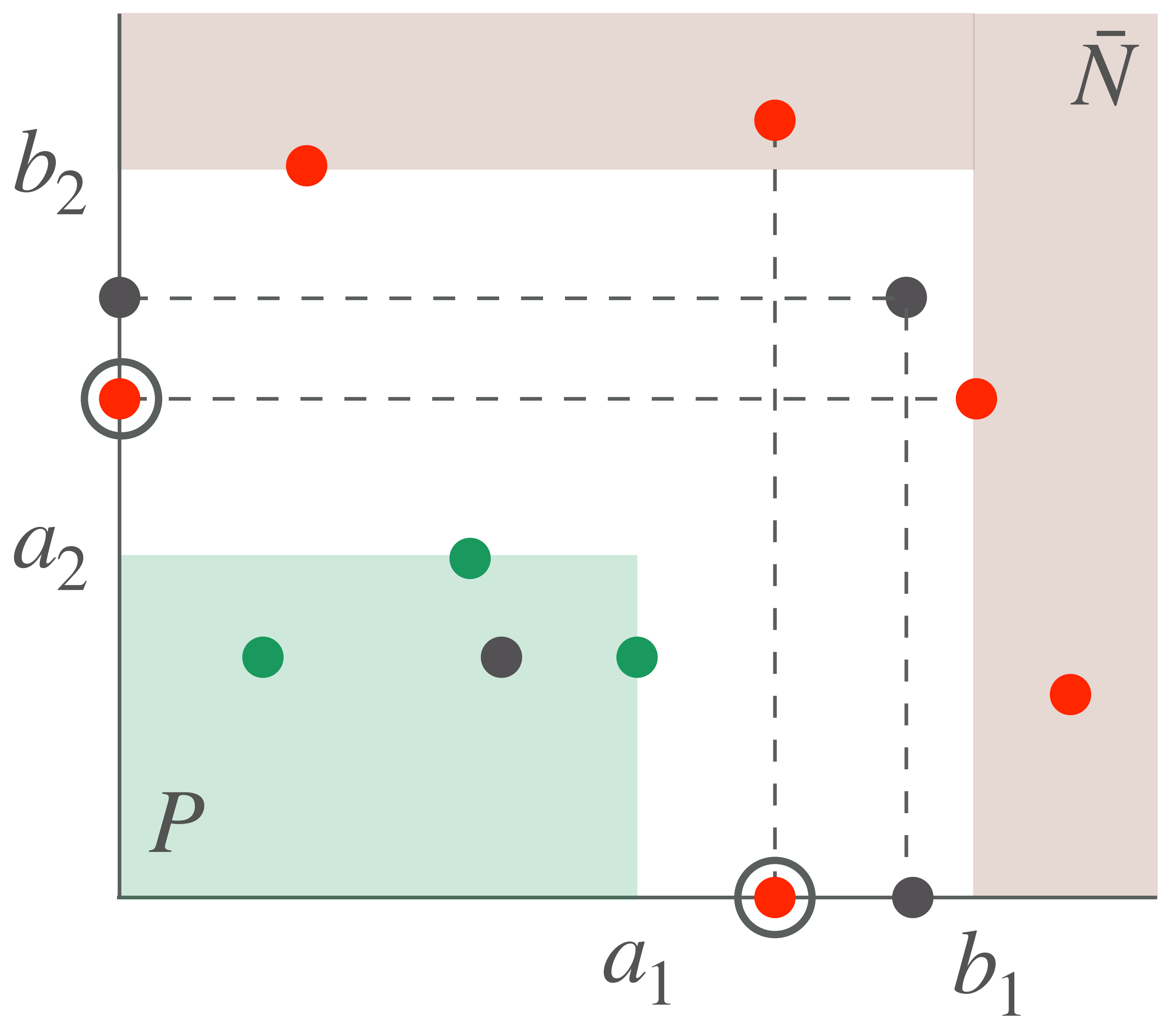
$$\mathbb{E}[\text{AbstentionError}] \leq O(\sqrt{dT \log T})$$

Note: *The error scales as* $\sqrt{dT \log T}$ *compared to* $\tilde{O}(d \log T)$ *in the known marginals*

RECTANGLE LEARNER

- Let P be the smallest rectangle enclosing the positive points and N be such that $\bar{N}$ is the largest rectangle containing no negative points
- To make a prediction on $\hat{x}_t$:
  - If $\hat{x}_t$ is not in the disagreement region, predict according to consistent label
  - Else count all the points $\hat{x}_i$ such that $\exists j \in \{1,2\}$ such that $\hat{x}_{i,j} \in (a_j, \hat{x}_{t,j}]$

    Votes to predict negative
    - If the number of such points is $\geq \alpha$, predict negative, else abstain
- Update P and N after getting the true label



PROOF OVERVIEW

- We make no mistake on positive points

They cannot vote negative in that direction anymore

- Every time we make a mistake on a negative, we can eliminate one direction each of α number of examples from the history, **MisclassificationError** $\leq \frac{2T}{\alpha}$
- The adversary can fool us on a new i.i.d. example with probability at most $2(\alpha + 1)/n$ if we have seen n i.i.d. examples, **AbstentionError** $\leq 2(\alpha + 1)\log T$

For the latter, we show that not many i.i.d. points can be attacked even if the adversary knows all the i.i.d. points

OPEN QUESTIONS

- **General VC classes without distribution access:**
  - Only know results for special classes by exploiting structural properties
- **Heuristics for more complex classes:**
  - Can we test out heuristic algorithms for deep learning setups
- **Beyond realizability:**
  - Benign noise models like random classification? We heavily use realizability
- **Beyond binary classification:**
  - Multi-class, partial concept class? Regression?
- **Computational efficiency:**
  - Are their efficient algorithms? Or statistical-computational trade-offs here?
- **Connections to other problems/techniques:**
  - Testable learning, conformal prediction, SoS style robustness