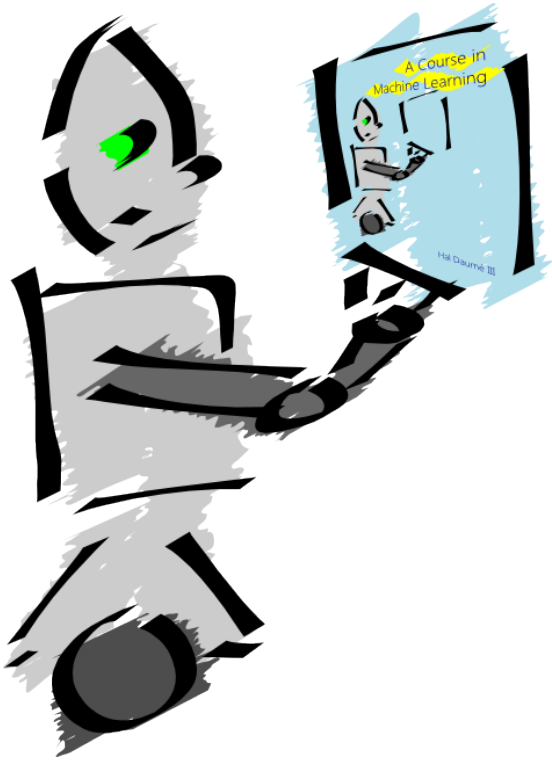
Machine Learning III



HAL DAUMÉ III, UMD
me@hal3.name
http://hal3.name
@haldaume3

Credit: much content based on material from
He He, Marc Tulio Ribeiro, Stéphane Ross, Jacob Steinhardt

What is this course about?

Machine learning studies algorithms for learning to do stuff

- By finding (and exploiting) patterns in data
- Sometimes in ways we'd rather they didn't
- Theory helps us understand this!

Last two times….

- What does it mean to learn?
  - Inductive bias
  - Linear models
  - Overfitting & underfitting

- When can learning fail?
  - Mismatched train/test
  - Sample selection bias

Today

- Attacking and defending ML systems
  - Data poisoning
  - Test data manipulation
- Understanding blackbox (ML) systems
  - LIME: Reduction to locally weighted linear regression
  - TLDR: Text categorization with minimal inputs

Data poisoning

- Setting:
  - Suppose you train a model on clean data D_C
  - But an adversary injects a small amount of poisoned data D_P into your system
- How bad can this impact your test performance?
- How can you prevent this?

Data poisoning: 2 player game

Jacob Steinhardt, Pang Wei Koh, Percy Liang. **Certified defenses for data poisoning attacks.** ICML 2017

$$max_{D_P} min_{\theta} L\left(\theta, D_C \cup D_P\right) + R\left(\theta\right)$$

Thm (Liu&Zhu, '16): in the worst case, even with *only one* poisoned example, you can arbitrarily worsen the model!

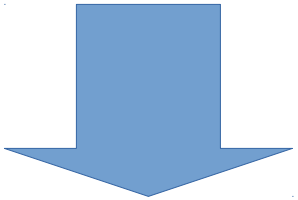
Biggio et al:

- To pursue security in the context of an arms race it is not sufficient to react to observed attacks, but it is also necessary to proactively anticipate the adversary by predicting the most relevant, potential attacks through a what-if analysis; this allows one to develop suitable countermeasures before the attack actually occurs, according to the principle of security by design.

Defense attempt: filtering

- Define *F* over X*Y as a set of "feasible examples" an only train on those

$$max_{D_P} min_{\theta} L(\theta, D_C \cup D_P) + R(\theta)$$



$$max_{D_P} min_{\theta} L(\theta, (D_C \cup D_P) \cap F) + R(\theta)$$

Types of filters

- **Static filters.** eg., only allow some features (words) through
- **Oracle filters.** eg., F is allowed to depend on true data distribution *P*
- **Data-dependent filters.** eg., F depends on the data (and thus on adversary)

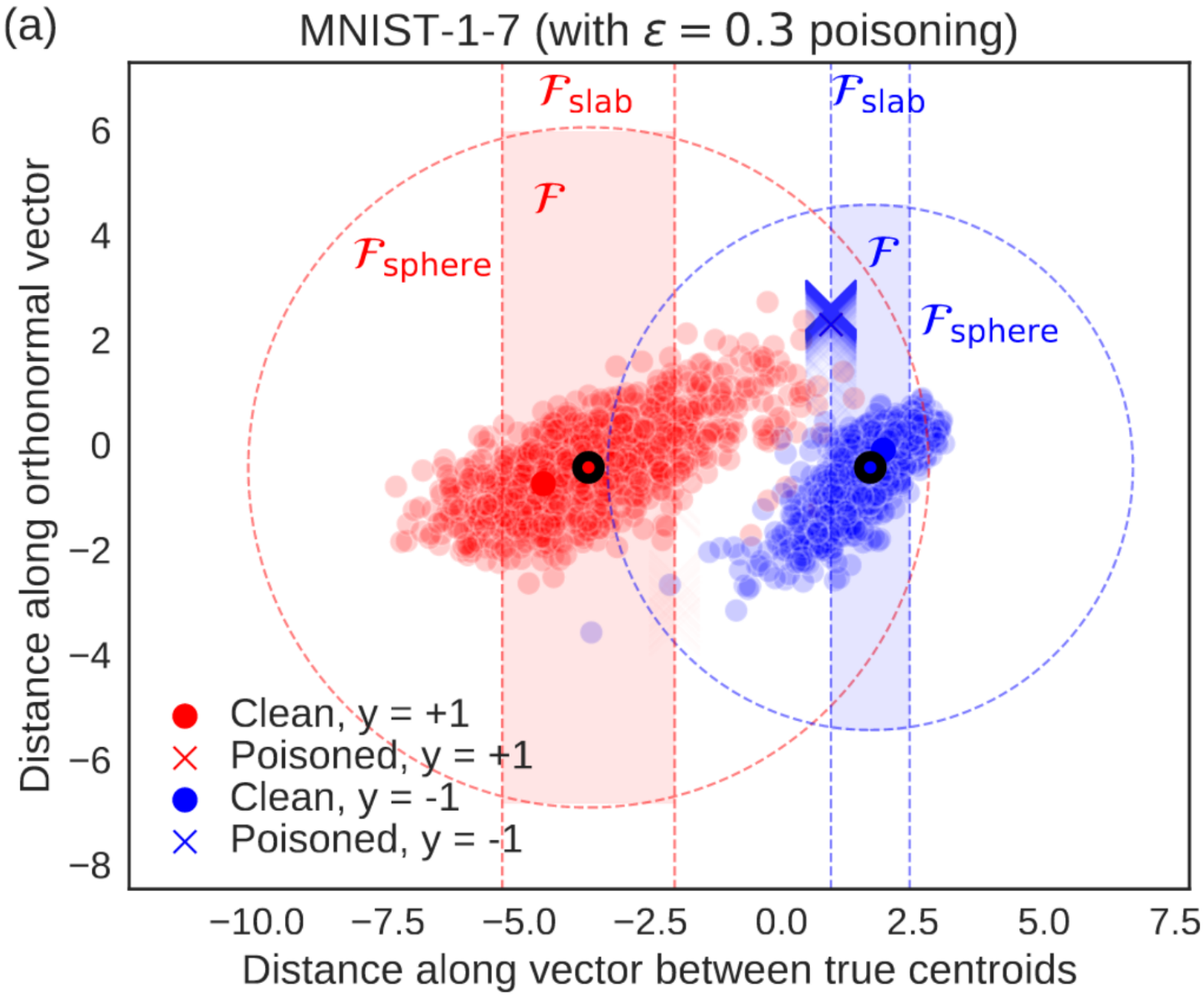
$$max_{D_P} min_{\theta} L\left(\theta, \left(D_C \cup D_P\right) \cap F\right) + R\left(\theta\right)$$

Example oracle filters

Let $\mu_+ \stackrel{\text{def}}{=} \mathbb{E}[x \mid y = +1]$ and $\mu_- \stackrel{\text{def}}{=} \mathbb{E}[x \mid y = -1]$

$$\mathcal{F}_{\text{sphere}} \stackrel{\text{def}}{=} \{(x, y) : \|x - \mu_y\|_2 \leq r_y\}$$

$$\mathcal{F}_{\text{slab}} \stackrel{\text{def}}{=} \{(x, y) : |\langle x - \mu_y, \mu_y - \mu_{-y} \rangle| \leq s_y\}$$

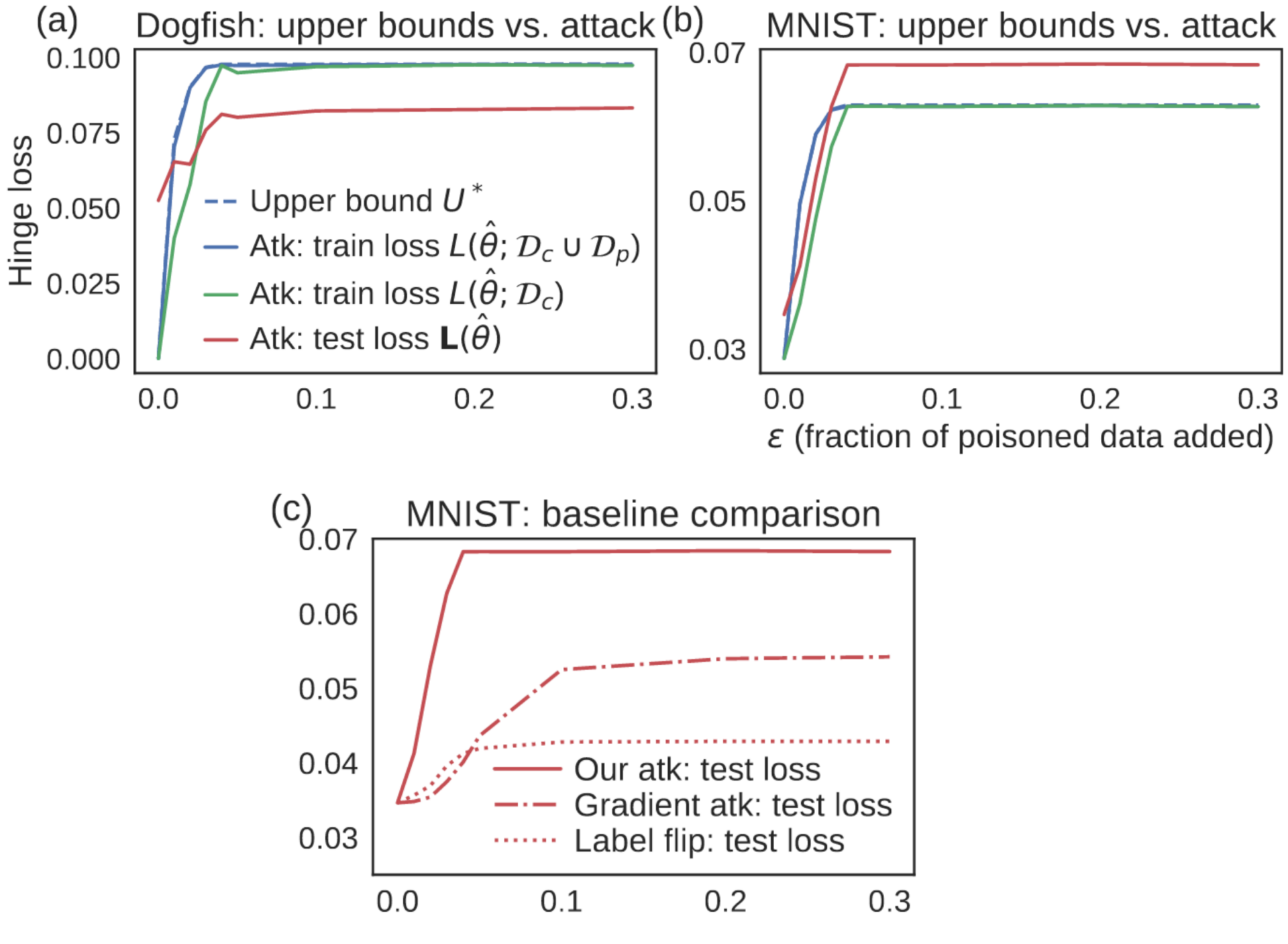

(a)
MNIST-1-7 (with $\varepsilon = 0.3$ poisoning)
$\mathcal{F}_{\text{slab}}$
$\mathcal{F}_{\text{slab}}$
$\mathcal{F}$
$\mathcal{F}$
$\mathcal{F}_{\text{sphere}}$
$\mathcal{F}_{\text{sphere}}$
Clean, y = +1
Poisoned, y = +1
Clean, y = -1
Poisoned, y = -1
Distance along orthonormal vector
6
4
2
0
−2
−4
−6
−8
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
Distance along vector between true centroids

Algorithm for optimal attack

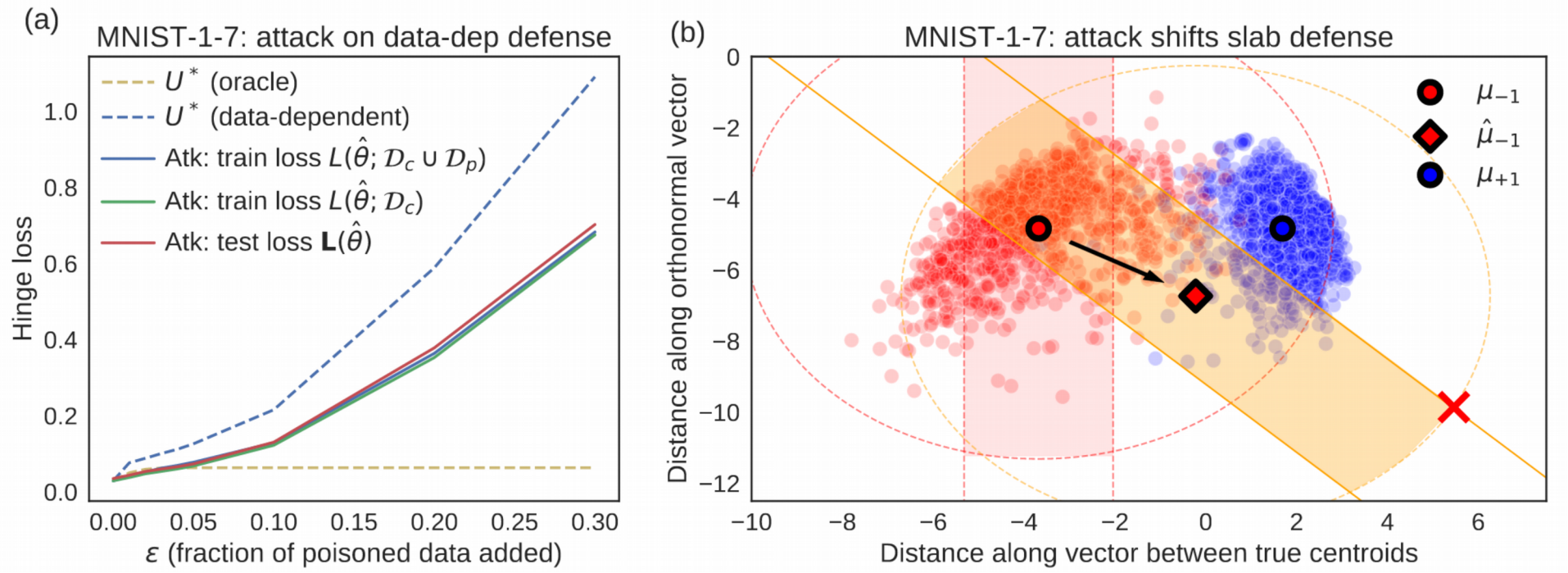
$$max_{D_P} min_{\theta} L(\theta, (D_C \cup D_P) \cap F) + R(\theta)$$

- Roughly, iterate the following:
  - Train a model on current data
  - Find (x,y) pair in F with highest possible loss
  - Add that to poisoned data

- Guarantee: finds a near-optimal (low regret) solution to the *attacker problem*


(a)
Dogfish: upper bounds vs. attack
Hinge loss
0.100
0.075
0.050
0.025
0.000
0.0
0.1
0.2
0.3
Upper bound U^*
Atk: train loss $L(\hat{\theta}; \mathcal{D}_c \cup \mathcal{D}_p)$
Atk: train loss $L(\hat{\theta}; \mathcal{D}_c)$
Atk: test loss $\mathbf{L}(\hat{\theta})$
(b)
MNIST: upper bounds vs. attack
0.07
0.05
0.03
0.0
0.1
0.2
0.3
ε (fraction of poisoned data added)
(c)
MNIST: baseline comparison
0.07
0.06
0.05
0.04
0.03
0.0
0.1
0.2
0.3
Our atk: test loss
Gradient atk: test loss
Label flip: test loss

But that all assumes we know P!



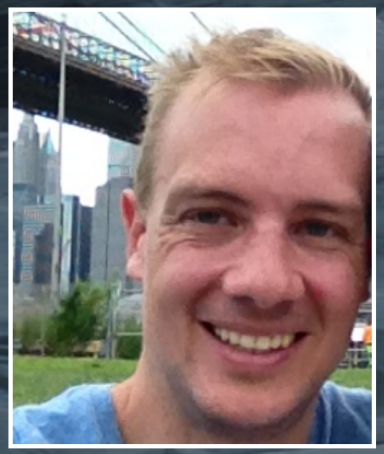
Discussion

- If *P* is known, are quite robust to attacks
  - Both in practice and provable
- If *P* is unknown, life is much worse
  - Connections to robust statistics

- Is the the thing we really want to defend against?

Test data manipulation

Emily M. Bender



Hal Daumé III

Allyson Ettinger

Harita Kannan

Sudha Rao

Ephraim Rothschild

Sentiment analysis task

- You get training data like:

```
+1 The acting is superb.
-1 This movie tried to be entertaining.
```

- You then get unlabeled test data

```
?  Every actor in this movie is horrible.
?  I love this movie!
```

- and must construct “minimal pairs”

```
+1 Every actor in this movie is wonderful.
+1 I’m mad for this movie!
```

Results

System	average F1
Strawman	**0.528**
Phrase-based CNN	0.518
Bag-of-ngrams	0.510
Sentence-based CNN	0.490
DCNN	0.483
RNTN	0.457

Table 1: Builder team scores: Average F1 across all breaker test cases, and percent of breaker test cases that broke the system

Example minimal pairs

ID	Minimal Pairs	Label	Rationale
Utrecht 1a	Through elliptical and seemingly oblique methods, he forges moments of staggering **emotional power**	+1	*Emotional pain can be positive*
Utrecht 1b	Through elliptical and seemingly oblique methods, he forges moments of staggering **emotional pain**	+1	
Utrecht 2a	[Bettis] has a smoldering, humorless intensity that's **unnerving**.	-1	*Funny can be positive & negative*
Utrecht 2b	[Bettis] has a smoldering, humorless intensity that's **hilarious**.	+1	
OSU 1a	A bizarre (and sometimes repulsive) exercise that's **a little too willing** to swoon in its own weird embrace.	-1	*Comparative*
OSU 1b	A bizarre (and sometimes repulsive) exercise that's **just willing enough** to swoon in its own weird embrace.	+1	
OSU 2a	Proves that **fresh new work** can be done in the horror genre if the director follows his or her own shadowy muse.	+1	*Sarcasm (single cue)*
OSU 2b	Proves that **dull new work** can be done in the horror genre if the director follows his or her own shadowy muse.	-1	

Example minimal pairs II

Melbourne 1a	Exactly the kind of **unexpected delight** one hopes for every time the lights go down.	+1	*(Not provided)*
Melbourne 1b	Exactly the kind of **thrill** one hopes for every time the lights go down.	+1	
Melbourne 2a	**American drama** doesn't get any more meaty and muscular **than this**.	+1	*(Not provided)*
Melbourne 2b	**This** doesn't get any more meaty and muscular **than American drama**.	-1	
Team4 1a	Rarely have good intentions been wrapped in such a **sticky** package.	-1	*(Not provided)*
Team4 1b	Rarely have good intentions been wrapped in such a **adventurous** package.	+1	

Today

- Attacking and defending ML systems
  - Data poisoning
  - Test data manipulation

- Understanding blackbox (ML) systems
  - LIME: Reduction to locally weighted linear regression
  - TLDR: Text categorization with minimal inputs

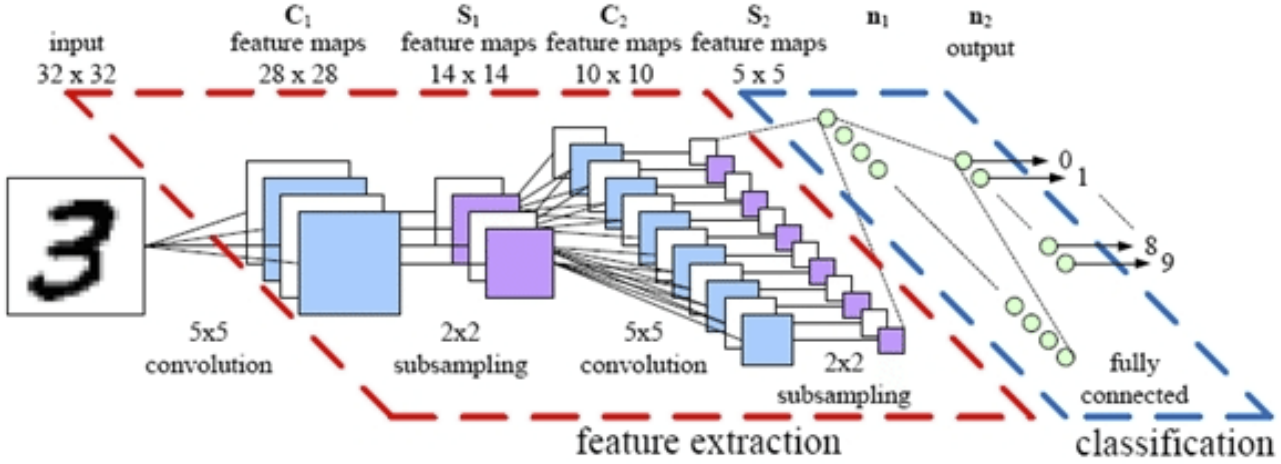
Black box → Linear model

Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin. **"Why should I trust you?": Explaining the predictions of any classifier.** KDD 2016.

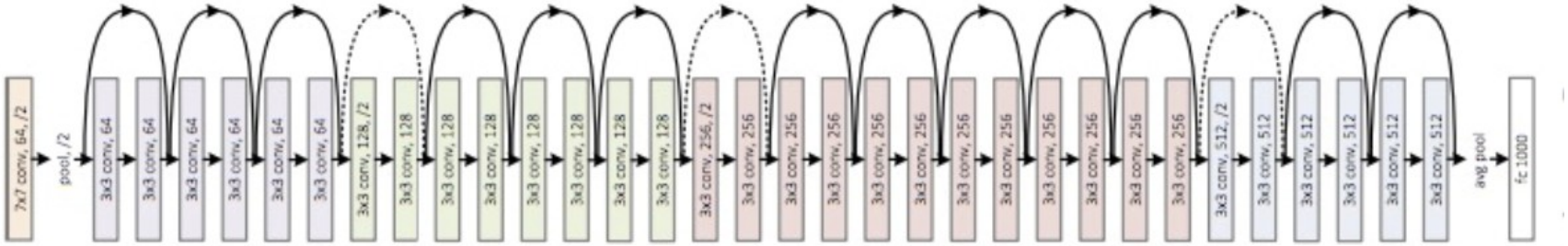
- Key observations:
  - Black box systems are hard to understand
  - Sparse linear models are easy to understand
- Ergo: explain decisions of black box systems *as if* they were linear models

Modern object detectors

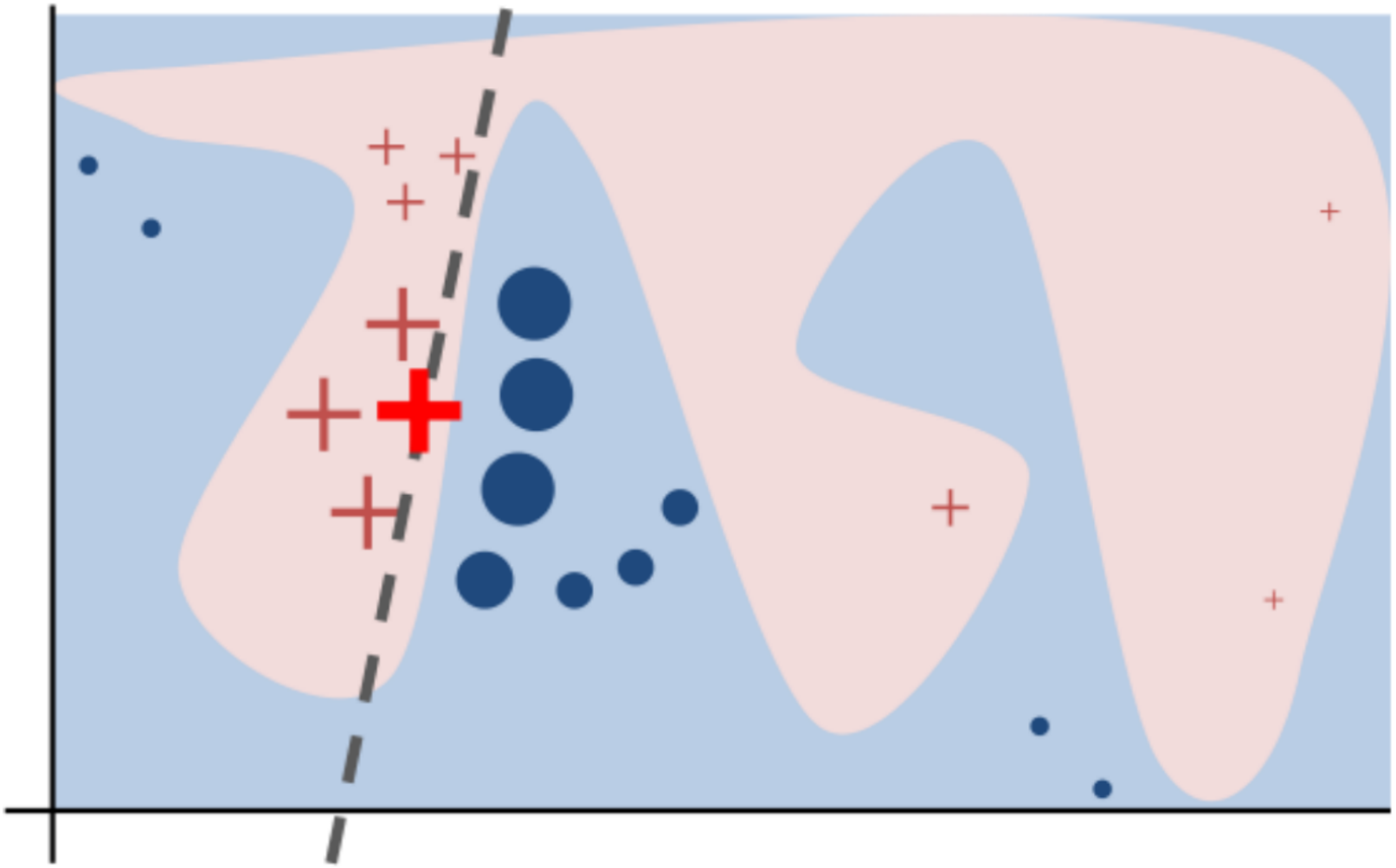
- AlexNet:



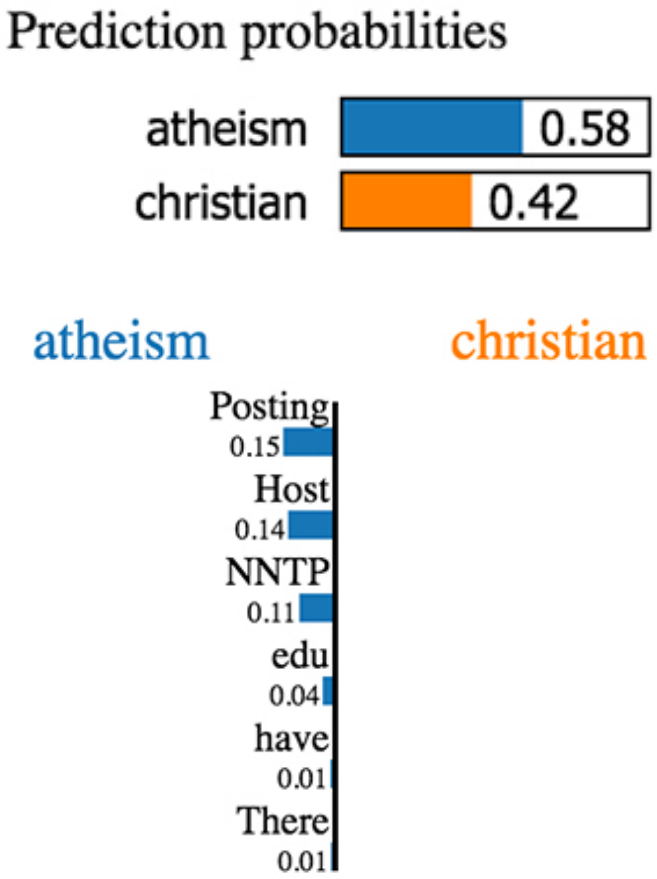
- ResNet:



Key idea in LIME



Example for text categorization



Text with highlighted words

From: johnchad@triton.unm.edu (jchadwic)
Subject: Another request for Darwin Fish
Organization: University of New Mexico, Albuquerque
Lines: 11
NNTP-Posting-Host: triton.unm.edu

Hello Gang,

There have been some notes recently asking where to obtain the DARWIN fish.
This is the same question I have and I have not seen an answer on the
net. If anyone has a contact please post on the net or email me.

Example for object detection

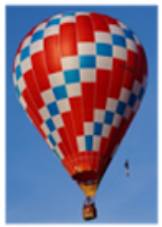
Original Image

Interpretable Components

Example for object detection

P() = 0.54

P() = 0.07

P() = 0.05

Can be used to diagnose bias

(a) Husky classified as wolf

(b) Explanation

Human studies

- Can users select the best model?
- Can non-experts improve the model?
- Do users trust the model?
  - See also

Marianna Martindale (UMD)

and

Rediet Abebe (Cornell/MSR)

Today

- Attacking and defending ML systems
  - Data poisoning
  - Test data manipulation

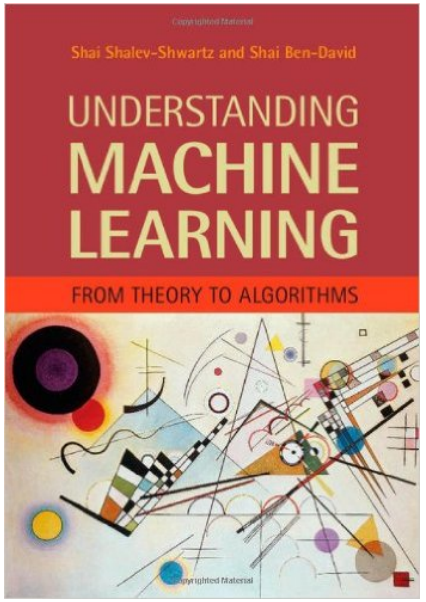
- Understanding blackbox (ML) systems
  - LIME: Reduction to locally weighted linear regression
  - TLDR: Text categorization with minimal inputs

Text categorization w/ minimal inputs

He He, Paul Mineiro, Nikos Karampatziakis. **Active information acquisition.** arxiv 1602.02181

- Key idea:
  - Jointly learn to select relevant information & classify
  - Hopefully what is useful to machine is same as what is useful to human
- Learn the selection process dynamically

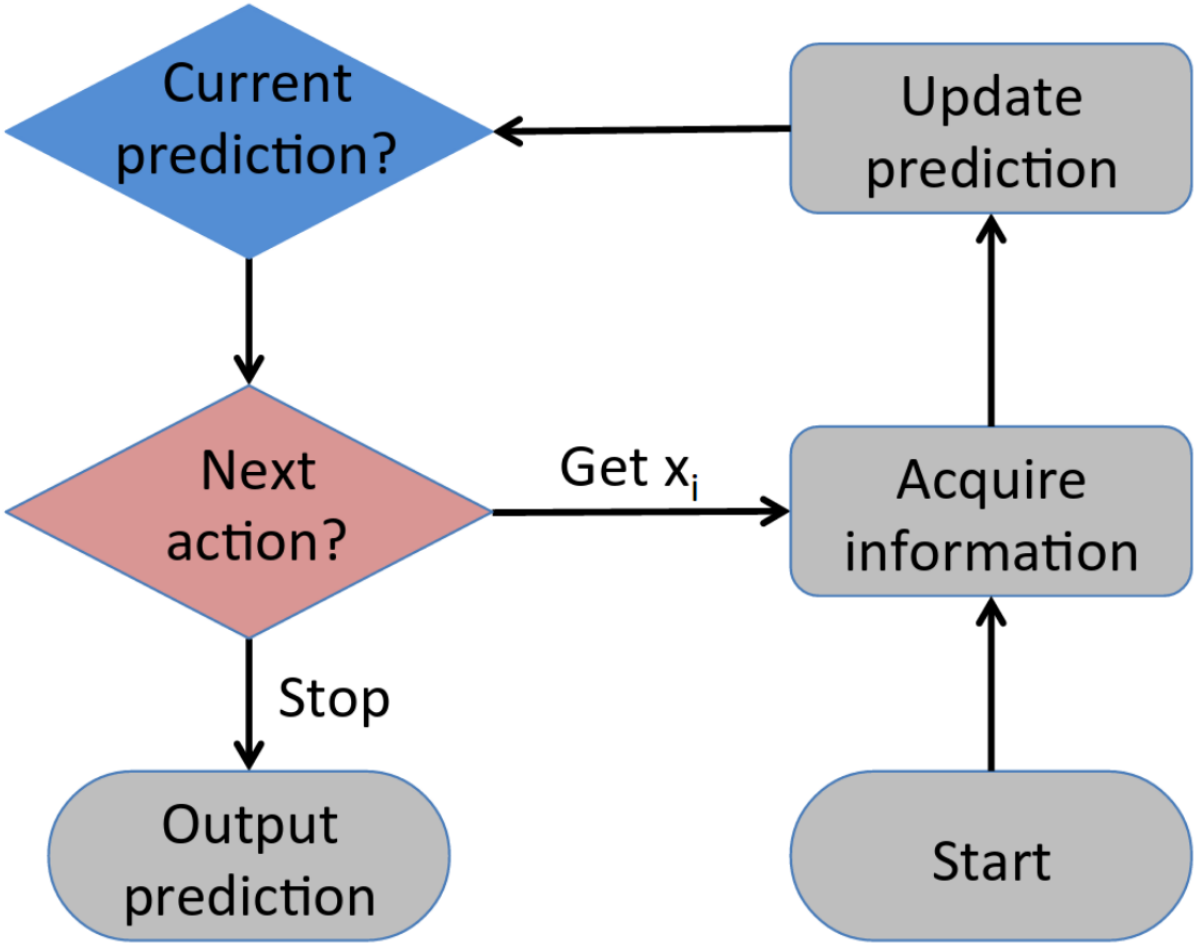
Amazon review

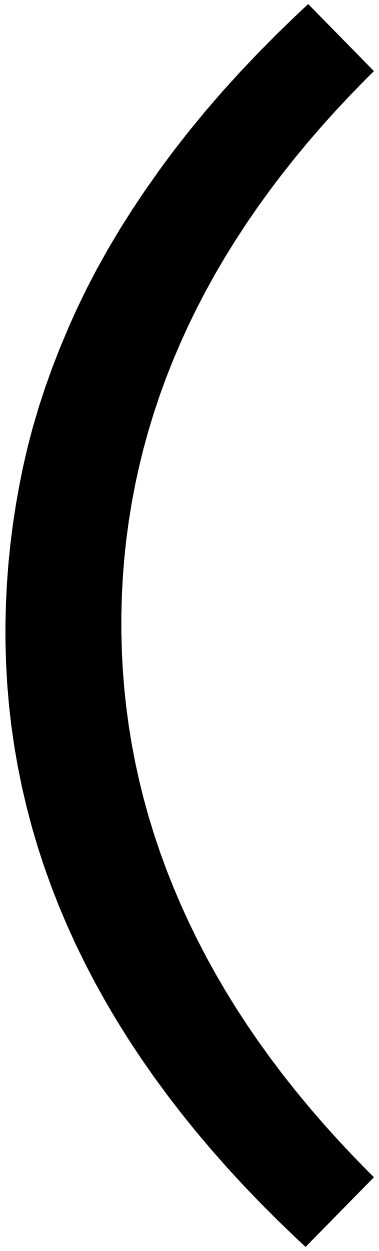


This book provides a great story line along with solid proofs of machine learning theories and algorithms. Each chapter is rather short (15-20 pages), yet is well written to convey the topic in detail, making the book comfortable to read. Moreover, the connection among consecutive chapters is strong, giving an excellent coarse-to-fine introduction on sophisticated theories.

Over the past few years, I have read several machine learning books, and this is the one solidly based on "statistical learning theory". Compared to other books that give only brief description to this aspect, this book does a good job not only on providing the basic proofs, but also on extending the theories to well-known practical algorithms, supporting the success of these algorithms and showing how theories can be used to design or analyze practical algorithms. For whom eager to know more about learning theory, this is a must-read book.

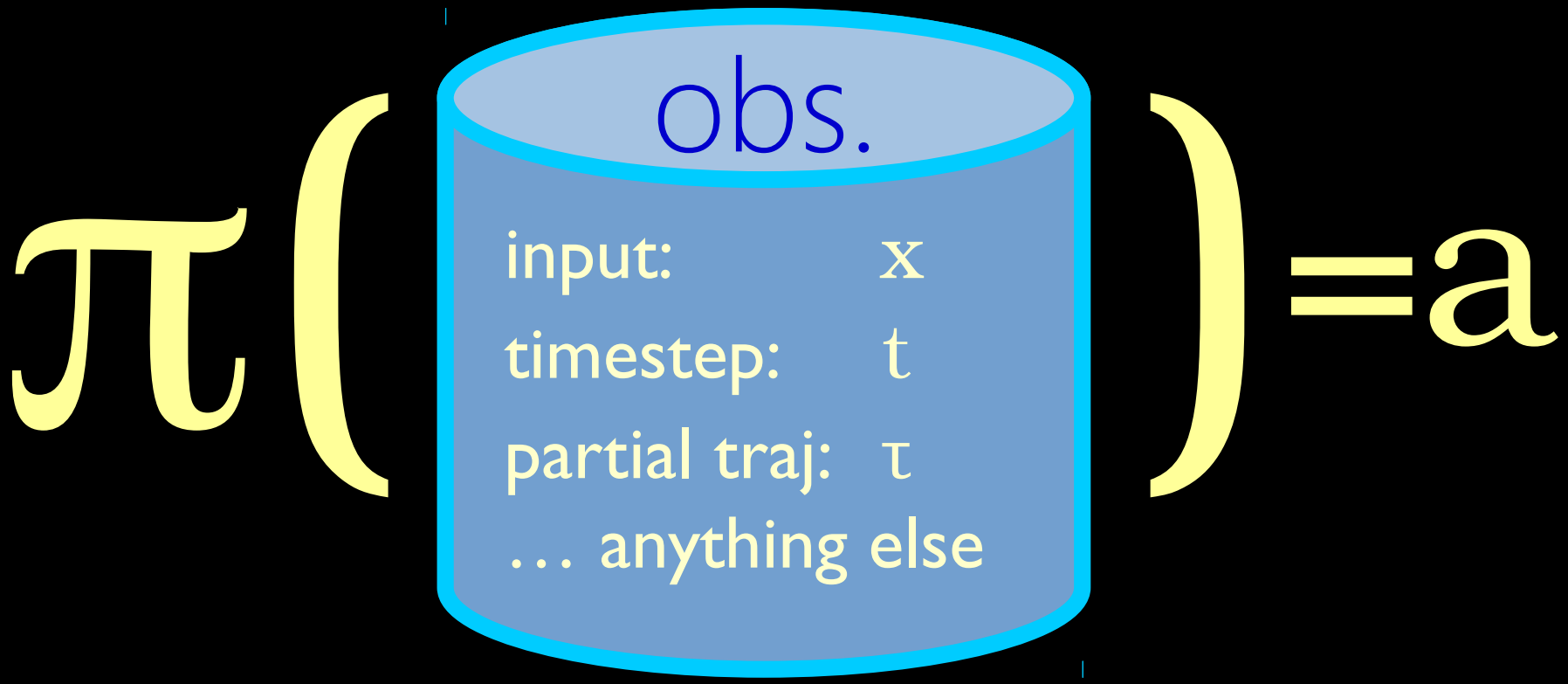
Information access model





Policies

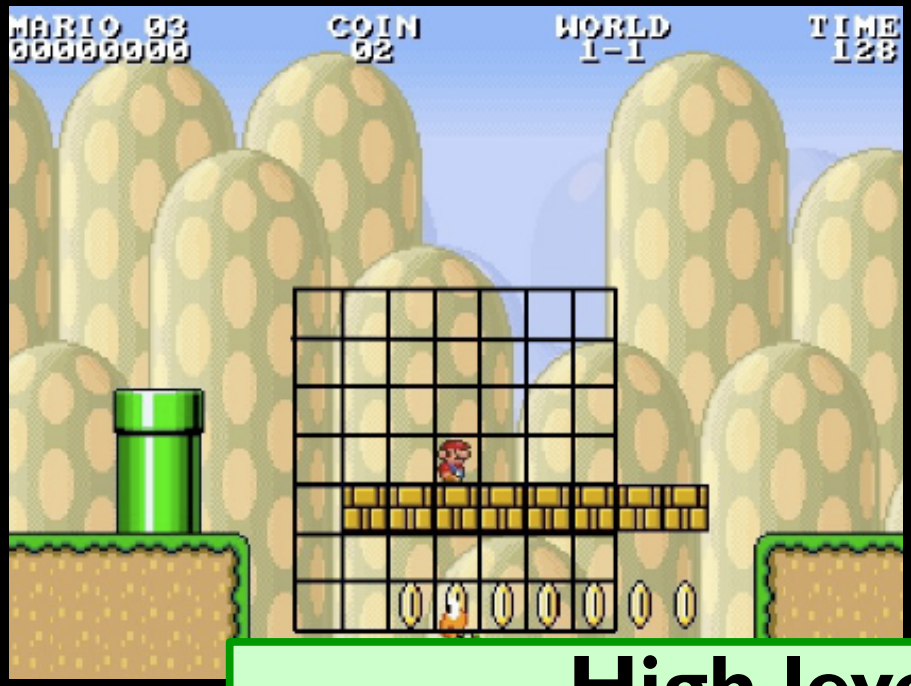
- A policy maps observations to actions



An analogy from playing Mario

From Mario AI competition 2009

Input:



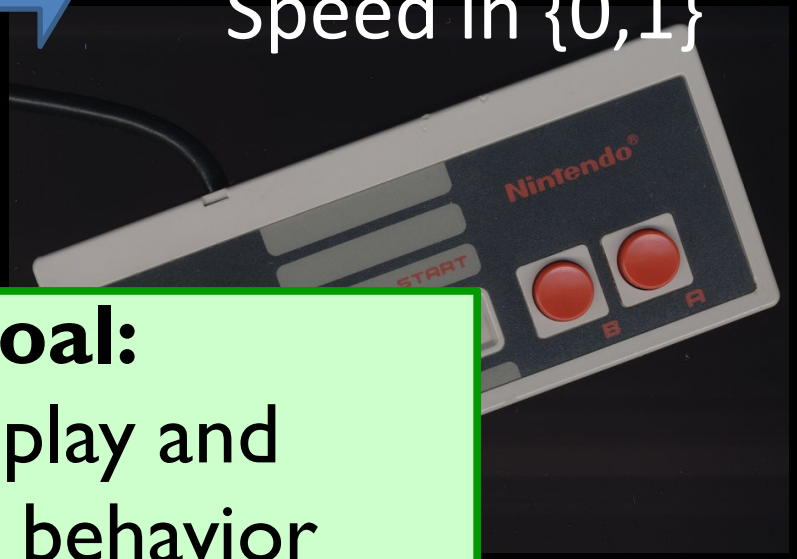
Output:

Jump in {0,1}

Right in {0,1}

Left in {0,1}

Speed in {0,1}



High level goal:

Watch an expert play and learn to mimic her behavior

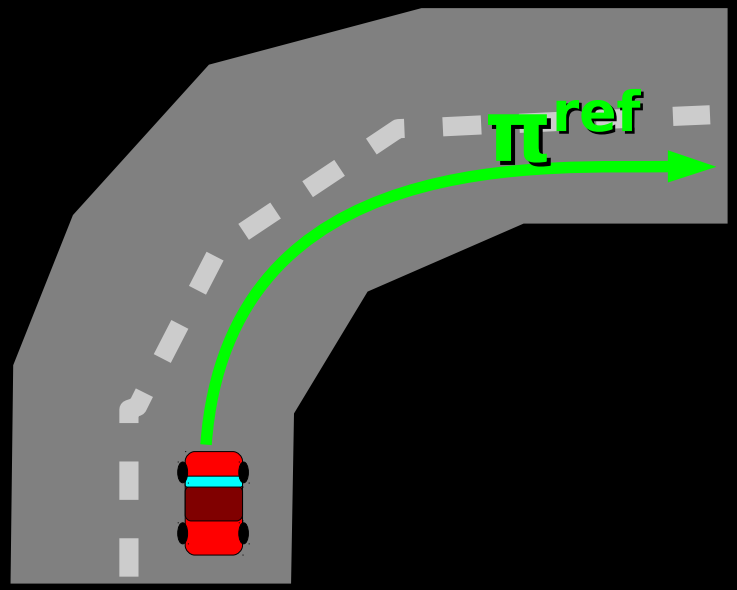
Training (expert)

Video credit: Stéphane Ross, Geoff Gordon and Drew Bagnell

Warm-up: Supervised learning

1. Collect trajectories from expert π^{ref}
2. Store as dataset $\mathbf{D} = \{ (o, \pi^{ref}(o,y)) \mid o \sim \pi^{ref} \}$
3. Train classifier π on $\mathbf{D}$

- **Let π play the game!**



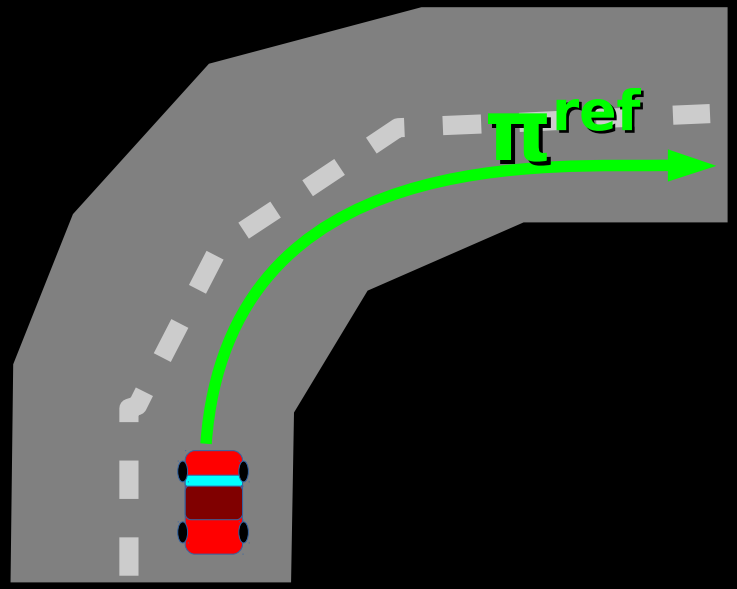
Test-time execution (sup. learning)

Video credit: Stéphane Ross, Geoff Gordon and Drew Bagnell

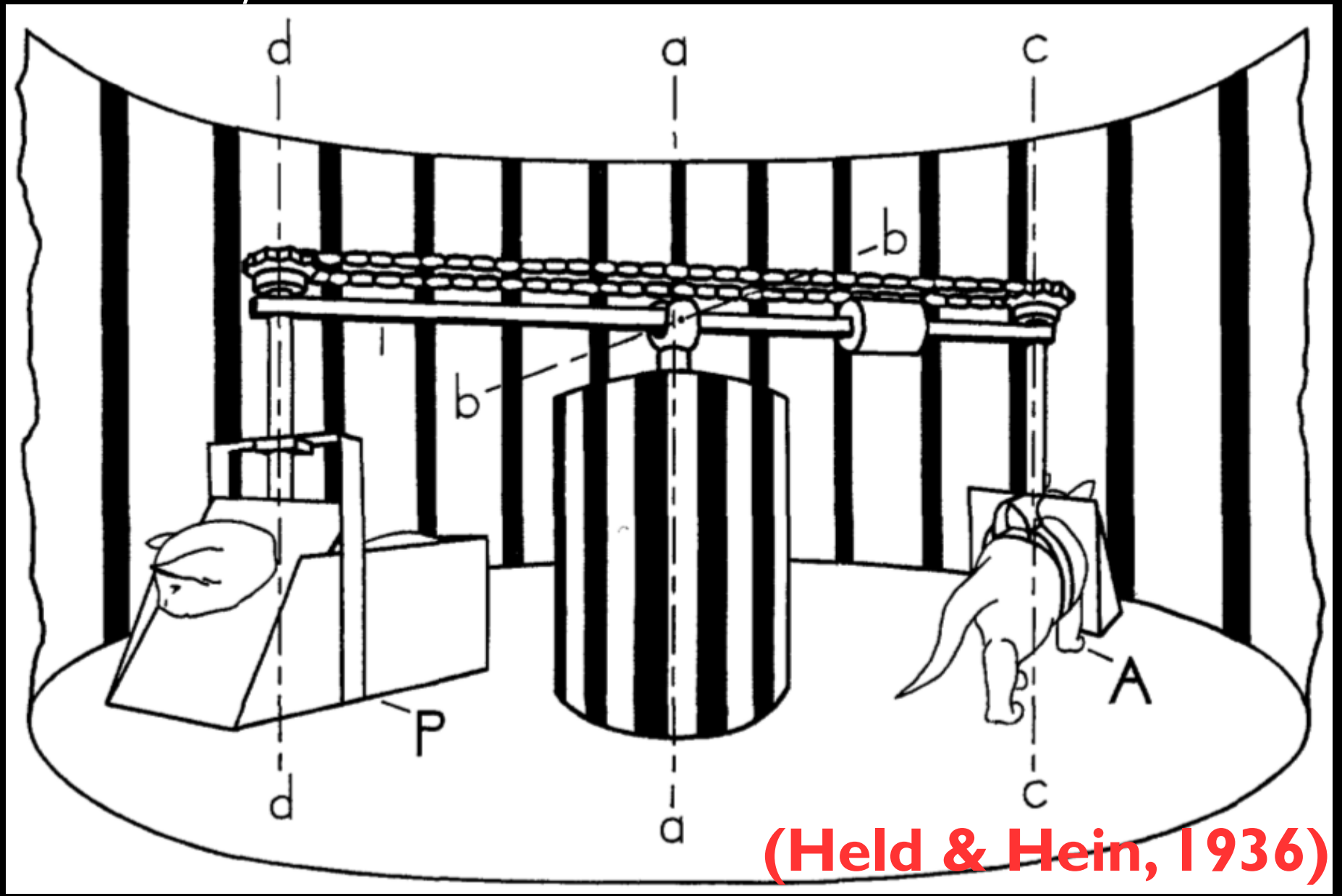
What's the (biggest) failure mode?

The expert never gets stuck next to pipes

⇒ Classifier doesn't learn to recover!



Kittens, revisited.



(Held & Hein, 1936)

Warm-up II: Imitation learning

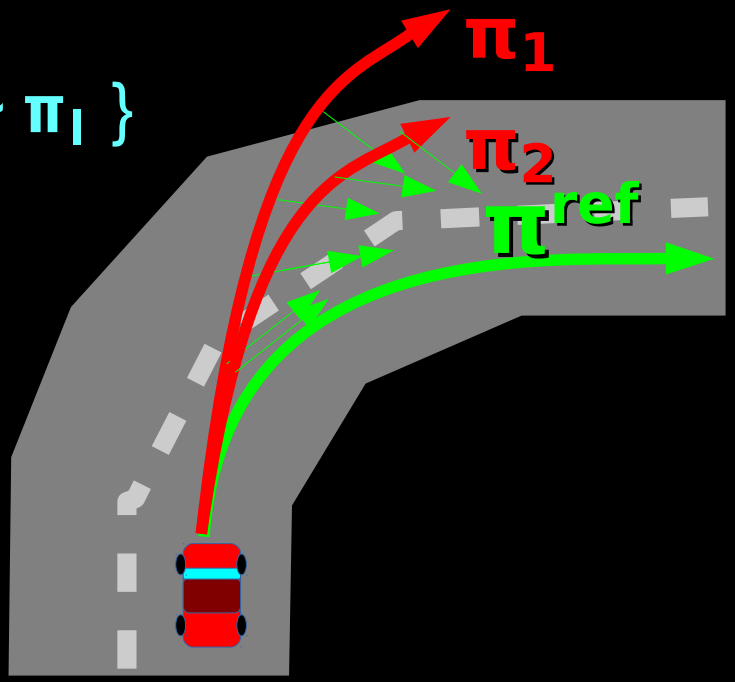
1. Collect trajectories from expert π^{ref}
2. Dataset $D_0 = \{ (o, \pi^{ref}(o,y)) \mid o \sim \pi$
3. Train π_1 on D_0
4. Collect new trajectories from π_1
   - But let the *expert* steer!
5. Dataset $D_1 = \{ (o, \pi^{ref}(o,y)) \mid o \sim \pi_1 \}$
6. Train π_2 on $D_0 \cup D_1$

- In general:
  - $D_n = \{ (o, \pi^{ref}(o,y)) \mid o \sim \pi_n \}$
  - Train π_{n+1} on $\cup_{i \le n} D_i$

If N = T log T,

$L(\pi_n) < T \epsilon_N + O(1)$

for some n

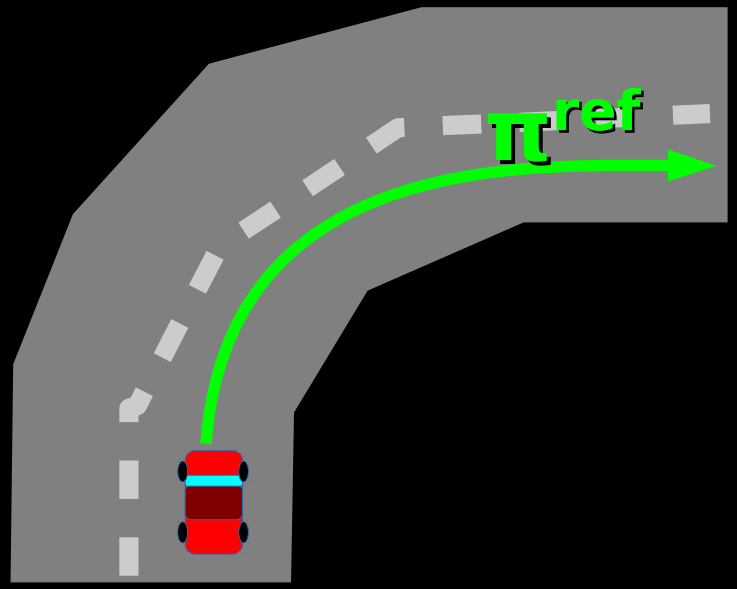


Test-time execution (DAgger)

Video credit: Stéphane Ross, Geoff Gordon and Drew Bagnell

What's the (biggest) failure mode?

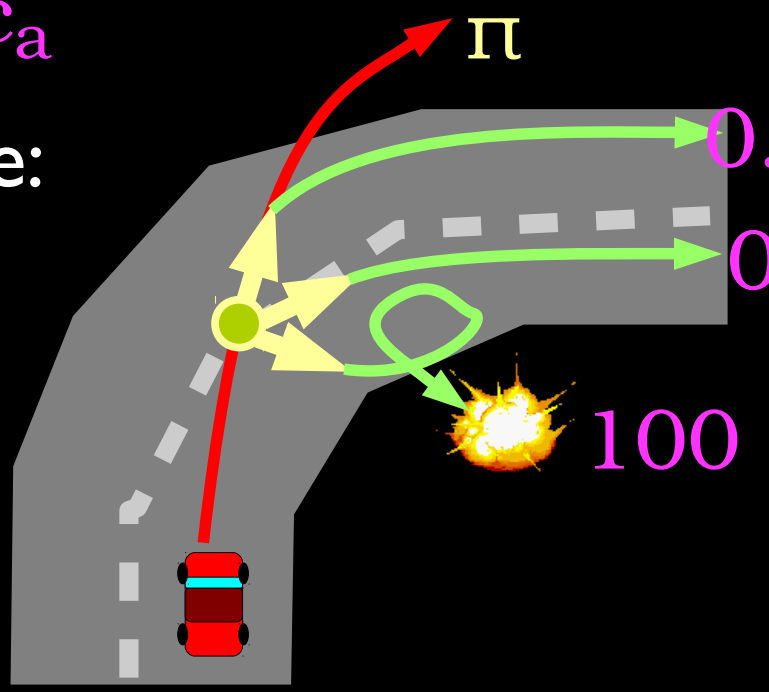
Not all errors are created equally



Learning to search: AggraVaTe

1. Let learned policy π drive for t timesteps to obs. o
2. For each possible action a:
   - Take action a, and let expert π^{ref} drive the rest
   - Record the overall loss, c_a
3. Update π based on example:

   $(o, \langle c_1, c_2, \ldots, c_K \rangle)$
4. Goto (1)



(Ross+Bagnell, 2014)

maçarico

built on pytorch

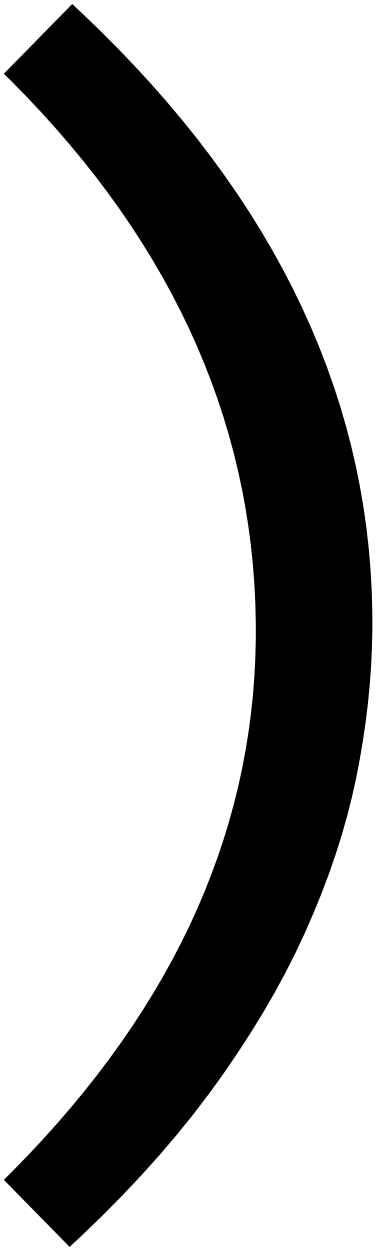
with Tim Vieira

release soon:

github.com/hal3/macarico

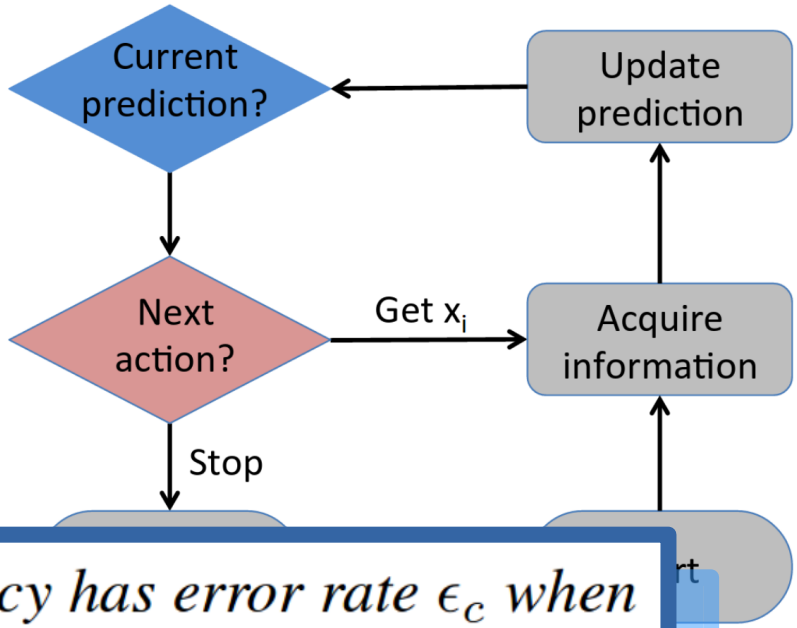
```python
import macarico

class DAgger(macarico.Learner):
    def __init__(self, expert, policy, rollin_expert):
        self.rollin_expert = rollin_expert
        self.policy = policy
        self.expert = expert
        self.objective = 0.0

    def __call__(self, state):
        exp = self.expert(state)
        pol = self.policy(state)
        self.objective += self.policy.forward(state, exp)
        return exp if self.rollin_expert() else pol

    def update(self, _):
        self.objective.backward()
```

PS 2016



Amazon review

This book provides a great story line along with solid proofs of machine learning theories and algorithms. Each chapter is rather short (15-20 pages), yet is well written to convey the topic in detail, making the book comfortable to read. Moreover, the connection among consecutive chapters is str
introd

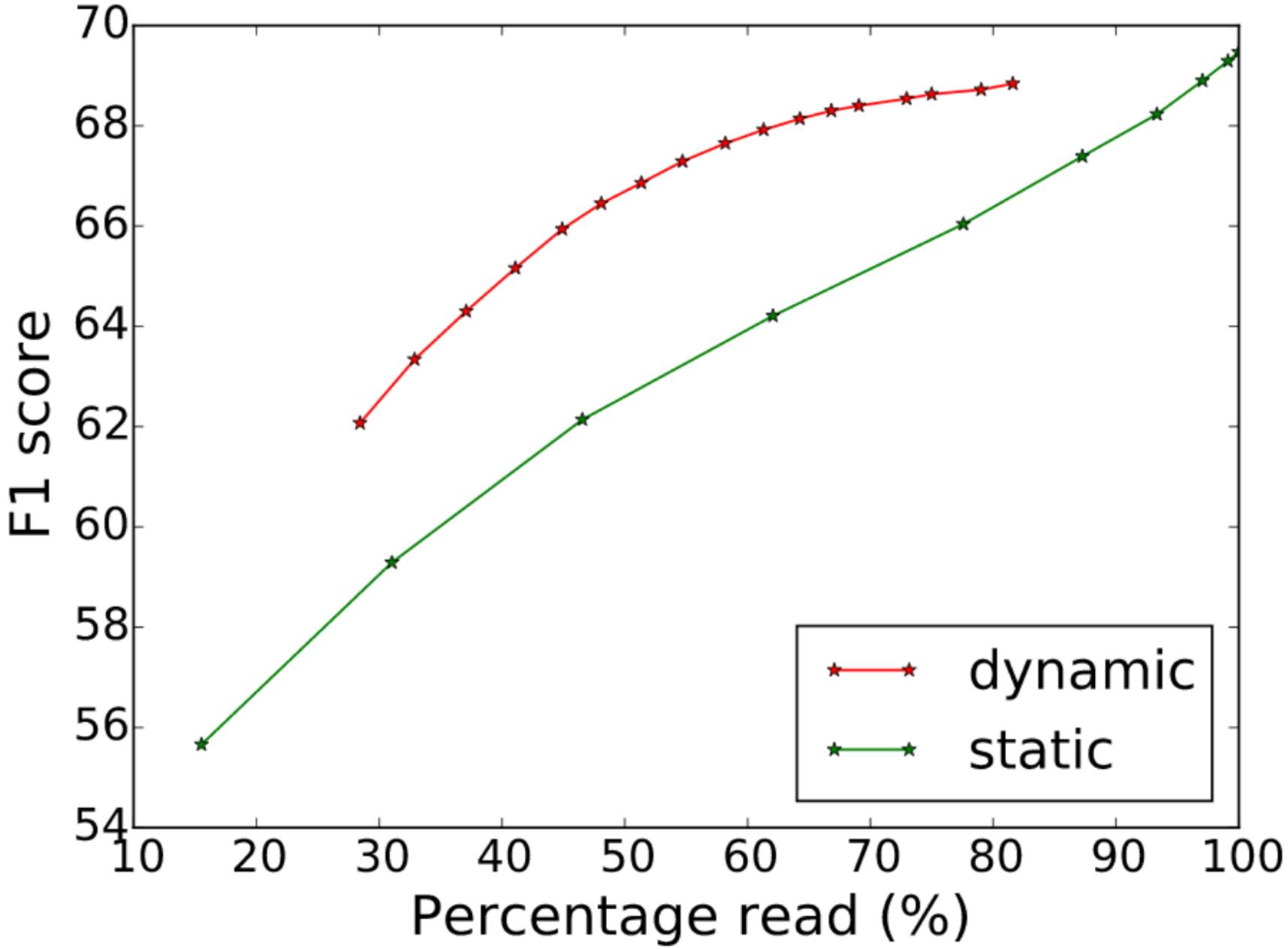


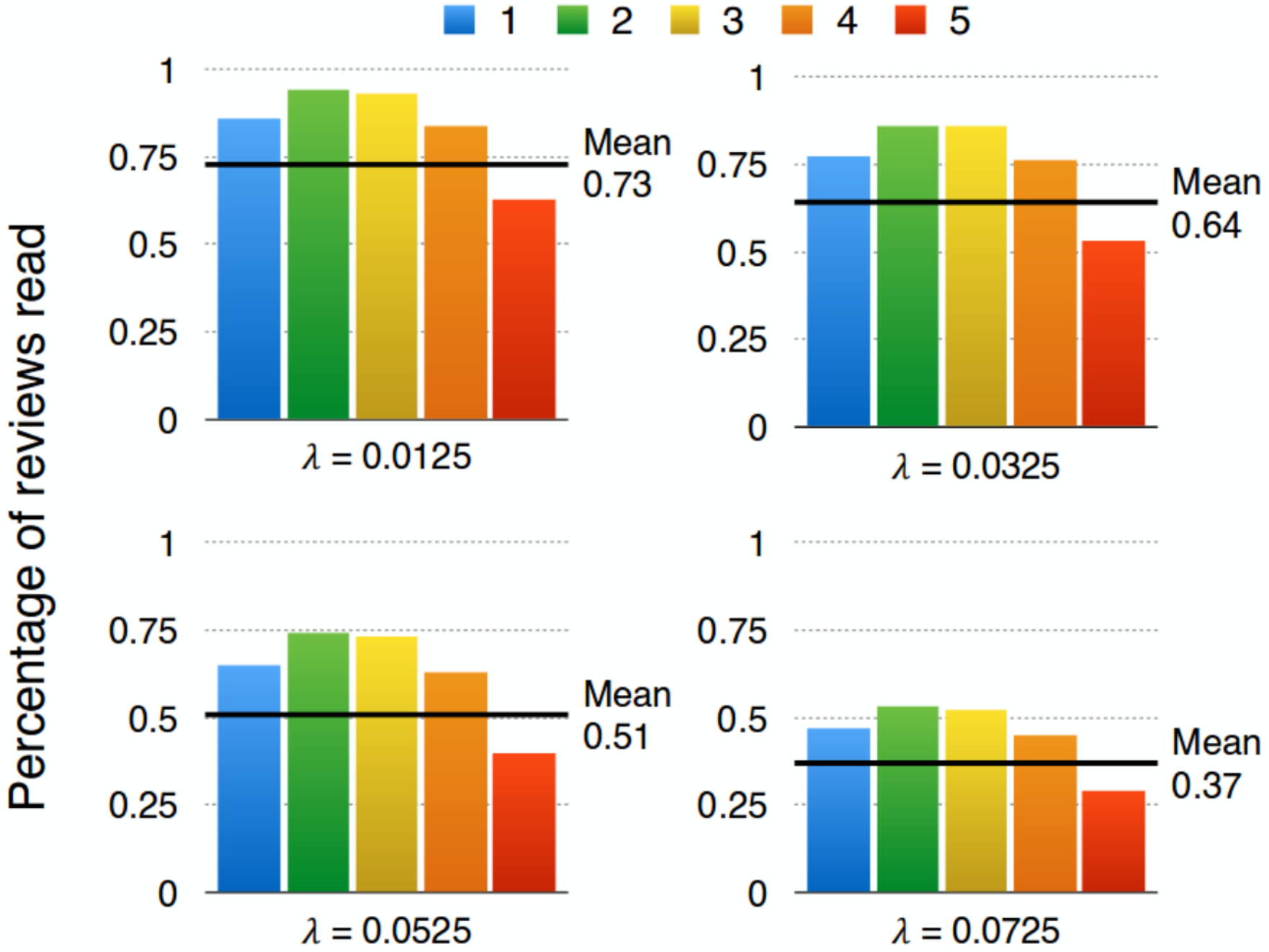
Corollary 1. *If the returned policy has error rate ϵ_c when evaluated in the multiclass classification setting, as an information selector it satisfies*

$$J(\pi) - J(\pi^*) \leq T\delta,$$

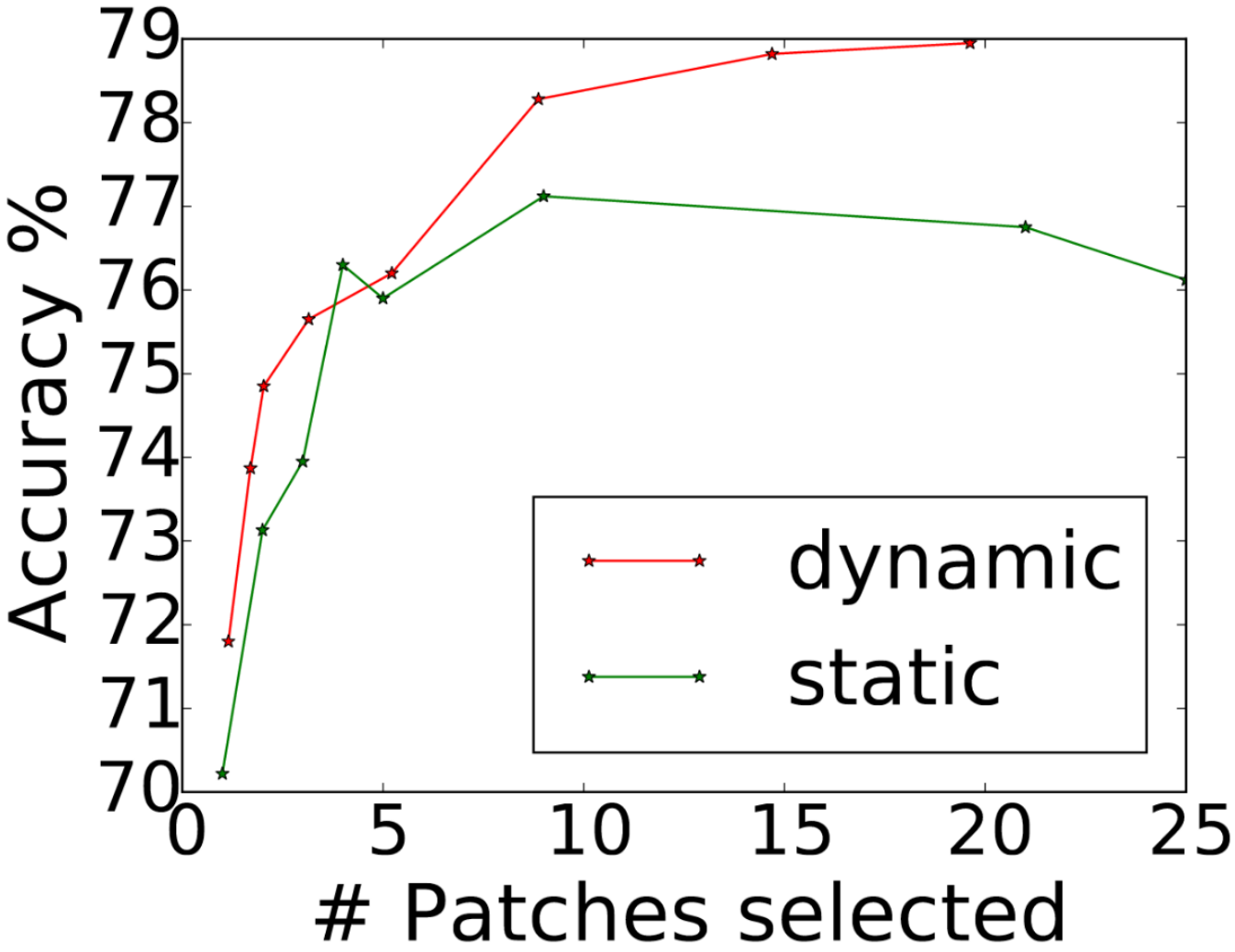
where $\delta = \epsilon_c \left(\Delta_{\max} + \lambda C + (1 - \alpha^{-1})Q^*_{\max}\right)$.

Over
one s
give
provi
pract
theor
know more about learning theory, this is a must-read book.


70
68
66
64
62
60
58
56
54
F1 score
10
20
30
40
50
60
70
80
90
100
Percentage read (%)
dynamic
static


1
2
3
4
5
Percentage of reviews read
1
0.75
0.5
0.25
0
Mean
0.73
λ = 0.0125
Mean
0.64
λ = 0.0325
Mean
0.51
λ = 0.0525
Mean
0.37
λ = 0.0725

Can apply the same ideas to image classification



Today

- Attacking and defending ML systems
  - Data poisoning
  - Test data manipulation
- Understanding blackbox (ML) systems
  - LIME: Reduction to locally weighted linear regression
  - TLDR: Text categorization with minimal inputs

Open questions…?

- Attacking and defending ML systems
  - What other attacks are there?
  - Data manipulation also useful for understanding!
- Understanding blackbox (ML) systems
  - What does it mean to “explain”?
  - How do explanations affect trust?
  - What happens when people start using explanations?

Open questions…?

- Bias and fairness in machine learning
  - What axiomatic notions of fairness exist? Compose?
  - What about less axiomatic notions?

- Learning from interaction
  - How do we minimize reliance on labelers?
  - How should we manage cost/benefit of data?
  - How can we learn from implicit interactions?