

Fairness in Machine Learning

Fairness in ML

- Typically a property of a **model** (ML algo output)
- Exceptions: online decision-making, RL, bandit settings
- Multiple **types** of fairness definitions

Types of Model Fairness

- Group fairness
(most common)
- Individual fairness
- Interpolations between
the two
- Others (causal, fair
representations, ...)

Group Fairness Notions

Start by identifying:

- groups or attributes we wish to "protect" (e.g. race, gender)
- what constitutes harm (e.g. error, false pos/neg)

Choices are subjective & domain-specific

Then seek to equalize rates of harm across groups.

Example:

- domain: consumer lending
- groups: male & female
- harm: false rejection (negs)

Want to find model $h(x)$ s.t.

$$FN(h, \text{male}) \approx FN(h, \text{female})$$

↗
↖ allows for optimization of overall error

Note: We can achieve
= FN rates by
randomization.

If individual x , predict
 $\hat{y} = +$ with prob. p

If $y = -$, can't be a FN

If $y = +$, $\hat{y} = -$ w.p. p

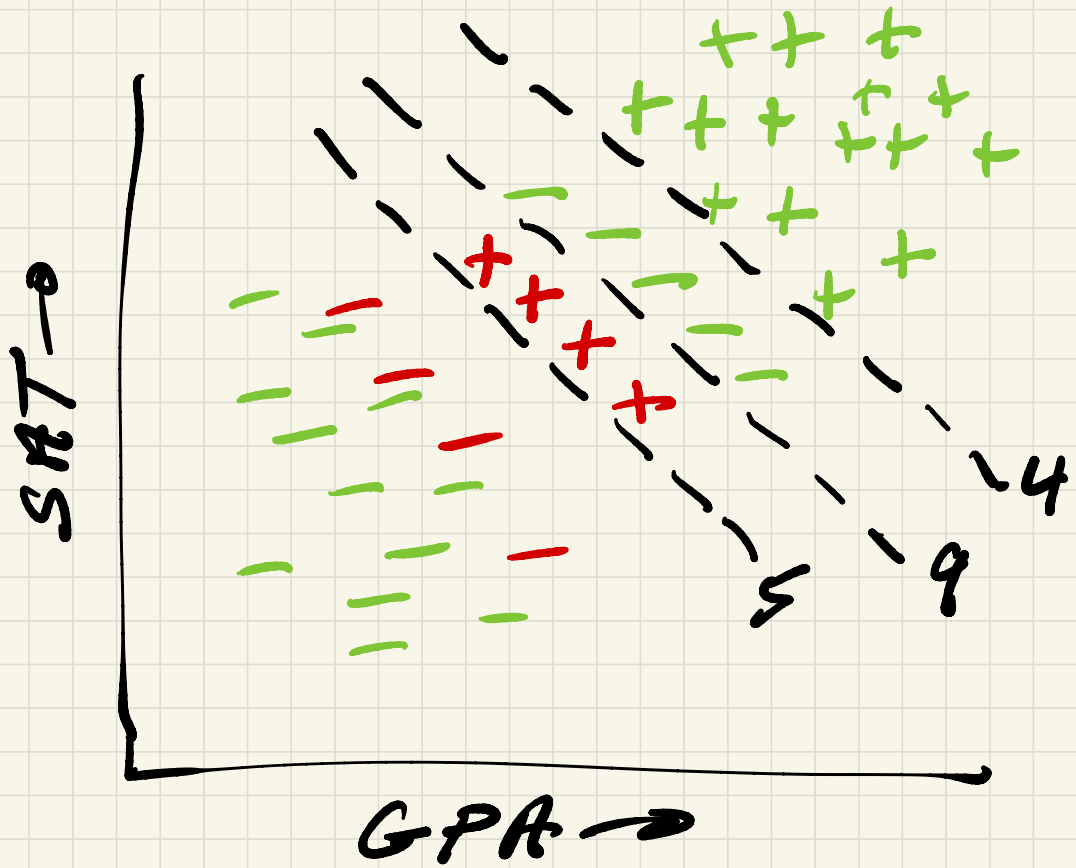
$\therefore FN(p, *) = p.$

If we are given a model $h(x)$ & have access to group membership, easy to audit $h(x)$ for fairness.

How can we learn a fair model $h(x)$?

Why won't standard ML algos work? •

A more subtle example



Ways Things Go Wrong

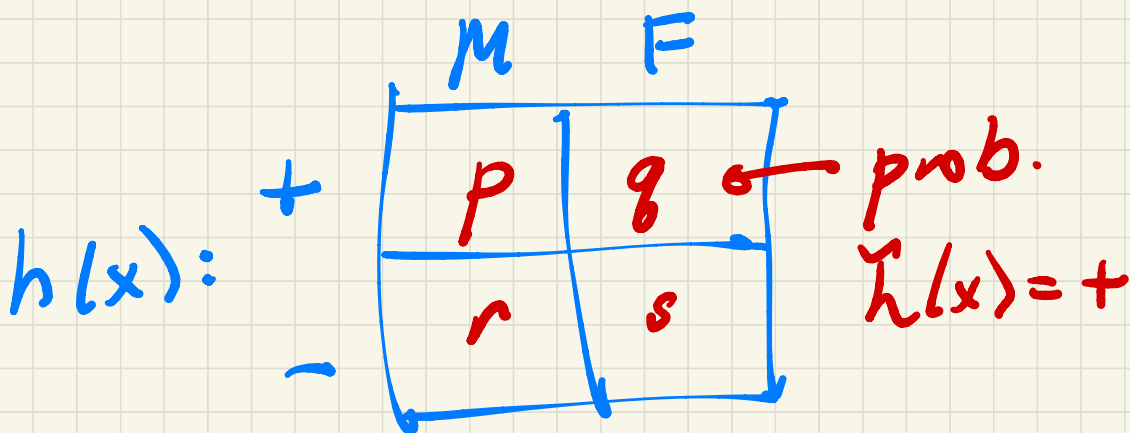
- Have much less data on some group (fine if groups all "same")
- Different groups have different distributions
- Our features are less predictive on some group
- Some group inherently less predictable
- Our data is biased in the first place

Algos for Fair ML:

Bias Mitigation

A Post-Processing Approach ("bolt on")

- start with non-fair $h(x)$,
want to \approx M/F error rates
- build a probabilistic classifier on top of $h(x)$:



$\tilde{h}(x)$

(closed under mixtures)

$$p = q = 1, r = s = 0:$$

$$\tilde{h} \equiv h, \varepsilon(\tilde{h}) = \varepsilon(h)$$

$$p = q = r = s = 1/2:$$

$$\varepsilon(\tilde{h}) = 1/2$$

perfectly
fair

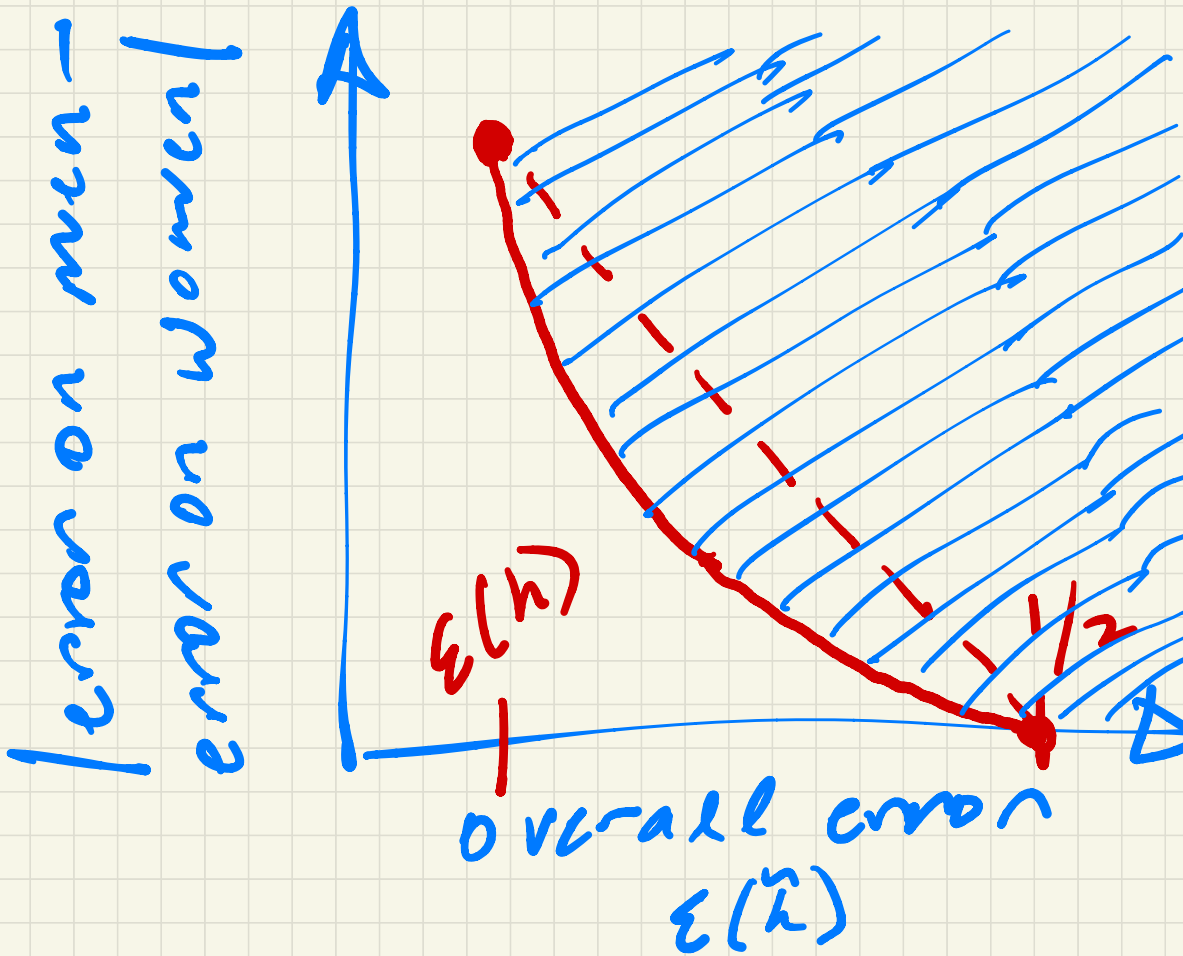
$$p = r = 1/2, q = s = 1:$$

error on men = 1/2

error on women = same
as h

etc.

Set of all $\langle p, q, r, s \rangle$ gives
Pareto frontier of h :



Algorithm

- Problem of finding \hat{h} that minimizes $\varepsilon(\hat{h})$

subject to

y-axis $\leq \gamma$

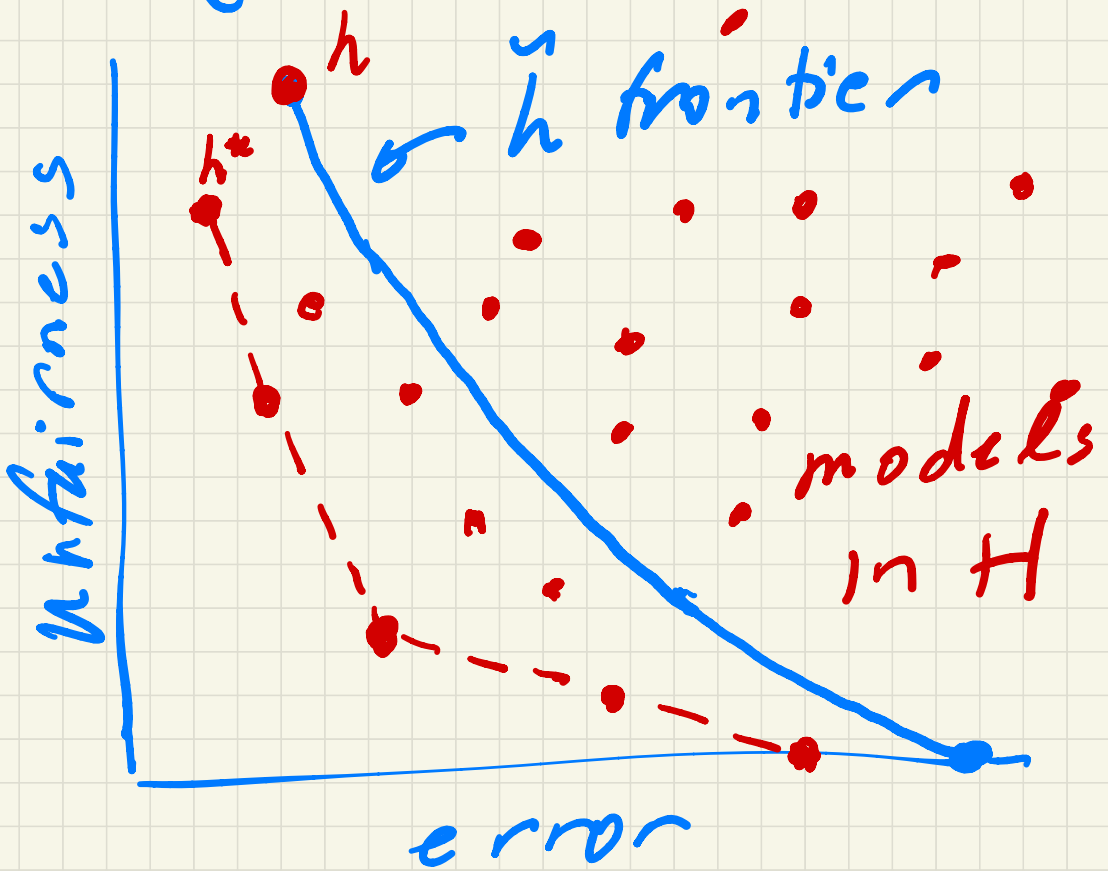
is a linear program

in \mathbb{R}^n .

(Framework & result due to Hardt, Price, Srebro.)

What more could we want?

- Imagine $h \in H$ (NNs, DTs, ...) by some learning algo



Can we find H -frontier?

Well...

- even finding $h^* \in H$ is intractable
in worst case
- but we do have effective non-fair heuristics

The Reductions/Oracle Approach

• Assume we have a black-box subroutine L for learning $h \in \mathcal{H}$ w.r.t. $\epsilon(h)$ only (non-fair)

• But L is "pretty good" & general (can solve weighted class. problems in \mathcal{H})

Show we can use L for fair learning.

Constrained optimization

$$\min_{h \in \Delta(H)} \{ \epsilon(h) \} \text{ s.t.}$$

fairness constraints:

$$(1) |\epsilon(h, \text{white}) - \epsilon(h, \text{black})| \leq \gamma$$

$$(2) |\epsilon(h, \text{white}) - \epsilon(h, \text{hispanic})| \leq \gamma$$

$$(3) |\epsilon(h, \text{black}) - \epsilon(h, \text{hispanic})| \leq \gamma$$

⋮

(k) (usually small, but...)

Introduce variables for weights
in $\Delta(H)$ & constraints \Rightarrow

huge LP.

- Learner picks $h \in H$
- Regulator picks a pair of groups, e.g. $A \neq B$
- Payoff to Regulator:
 $\hat{\epsilon}(h) + \max(0, |\hat{\epsilon}(h, A) - \hat{\epsilon}(h, B)| - \gamma)$
- Payoff to Learner:
 $-(\text{Payoff to Regulator})$

Game Theory Formulation

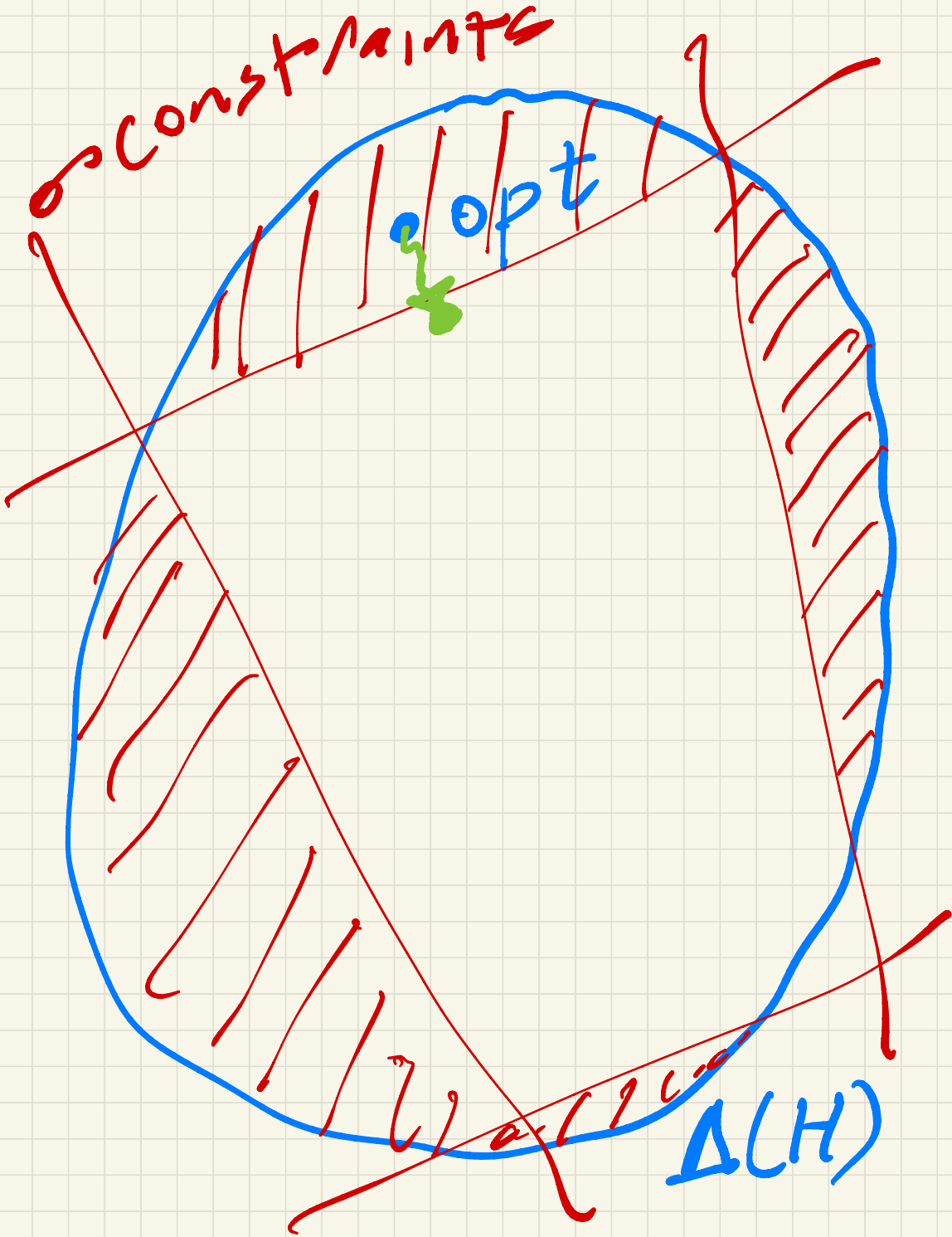
- Learner plays mixed strategy $p \in \Delta(H)$
- Regulator plays mixed strategy q over fairness constraints

- Zero-sum game on:

$$u(p) + \text{constraint violations}(p, q)$$

$$= \text{payoff to Regulator} \\ = - \text{payoff to Learner}$$

Nash equil = constrained opt solution



A Classic Theorem (Freund & Schapire)

If L & R play iteratively:

(1) L best responds to g_t

(2) R updates g_{t+1} using
no-regret algo

Then converge to


$1/\sqrt{t}$ -optimal
solution.

(2) usually easy

(1) often reduces to
weighted classification
with wts. given by δ_t

\Rightarrow "oracle" L.
(Agarwal et al.)

Yields "principled
heuristics" that
are implementable.



Towards
Individual
Fairness

Q: Why not treat
each individual x
as their own "group"?

A: Error (or FP, FN, ...) "rate" on x is
either 0 or 1.

But there are
other approaches...

Metric Fairness

- Posit a distance metric $d(x, x')$ between pairs of individuals
- $h(x)$ our real-valued predictor
- Then constrain $h(x)$ to obey $\forall x, x'$:

$$|h(x) - h(x')| \leq \alpha d(x, x')$$

Difficulties

- Where do we get $d(x, x')$?
- Closed form?
- Usually want to threshold $h(x)$, lose fairness
- Practical challenges

Subgroup
Fairness

- Suppose we ask for group fairness by all of race, gender, disability, age, income, ...

- Might still discriminate against disabled Hispanic women over age 55 making $\leq 20K/year$

Framework

- Model class H
- Group membership class G
- For $g \in G$, $g(x) \in \{0, 1\}$ indicates if x is in g (e.g. disabled Hispanic...)
- Now allowing G to be large or infinite

Game Theory II

- Learner plays $h \in H$
- Regulator plays $g \in G$,
finds **most violated**
 g (e.g. h has high
error on g)

Reduce to non-fair
case; L no-regret,
 R best response

Another Approach:

Average

Individual

Fairness

- Suppose we will make many decisions about x over time
- E.g. product rec's
- Then any h has error rate $\epsilon_x(h)$ across problems
- Ask that all $\epsilon_x(h)$ be \approx equal across individuals x
- Game Theory III •

Fairness

Elicitation

- What if fairness isn't "simple"...

- ...but we can elicit empirical fairness judgements.

- E.g.

"Alice & Bob should receive same treatment"

"Alice should be treated at least as well as Bob"

Framework

- Outcome data $S = \{(x_i, y_i)\}$
- Fairness data F of form $x_i = x_j, x_i \geq x_j$
- Find $h \in H$ that min's error on S subject to F
- Generalize to dist's of S & F
- Game Theory IV •

Beyond Equalization

- Problem: may achieve by heedlessly inflating harm to advantaged
- Alternative: minimax group fairness:

$$\min_{h \in H} \max_{\text{groups } g} \{ E_g(h) \}$$

- Game Theory V •

Alternative Approach:

"Bias Bounties"

Motivation

- AI activism & adversarial dynamics
- Can't defend against everything (?)
- Hard to anticipate groups harmed (gerrymandering)
- "Unexpected" data
- Bug bounties in software

First Idea:

- Invite "crowd" to find problems in a trained model
- E.g. on subsets of training data
- On new data
- "Blurry pictures problem"

Second Idea

- Invite crowd to propose improvements to a trained model
- Submit two classifiers:
 - group classifier
 $g(x) \in \{0, 1\}$
 - model classifier
 $h(x) \in \{0, 1\}$

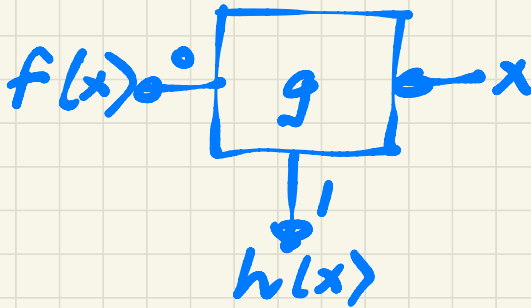
Idea is that $h(x)$ is more accurate on $g(x) = 1$ than current model

More formally:

- $f(x)$ is current model
- wrt $P, \epsilon_g(h) \not\leq \epsilon_g(f)$

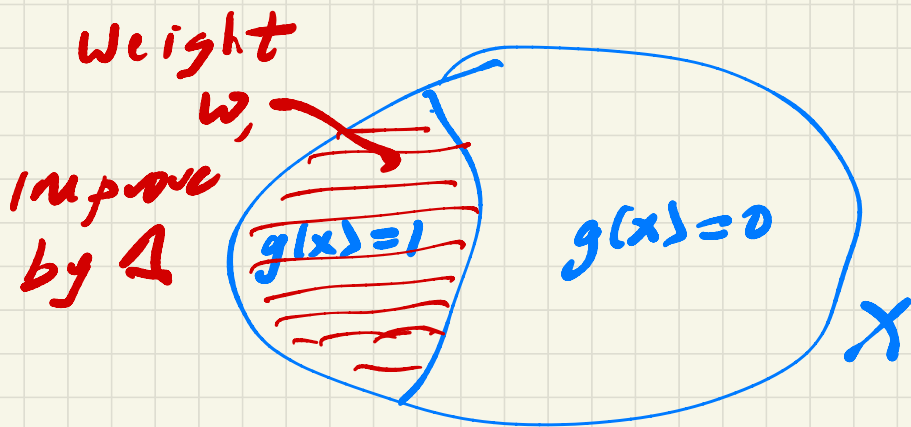
$$P_{x \sim P} [h(x) \neq f^*(x) | g(x) = 1]$$

- What to do?

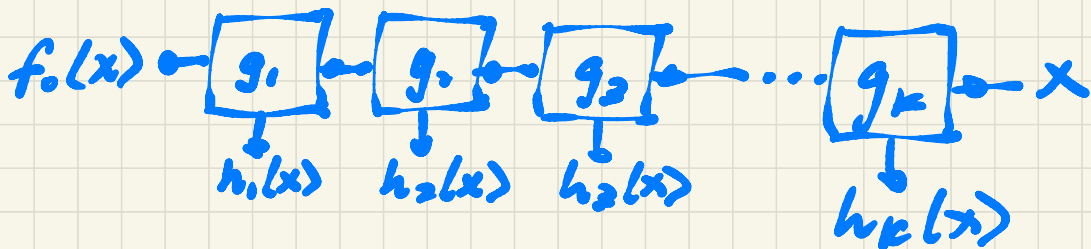


new model
 $f_1(x)$

- $\mathcal{E}_g(f_1) = \mathcal{E}_g(h) \leq \mathcal{E}_g(f)$
- let $\Delta = (\mathcal{E}_g(f) - \mathcal{E}_g(f_1)) \geq 0$
- let $w = P_{X \sim P} [g(x) = 1]$
- $\mathcal{E}(f_1) = (1-w)\mathcal{E}_{\pi_g}(f)$ *no change*
 $+ w\mathcal{E}_g(f_1)$ *better!*
- $\mathcal{E}(f) - \mathcal{E}(f_1) = w \cdot \Delta \geq 0$

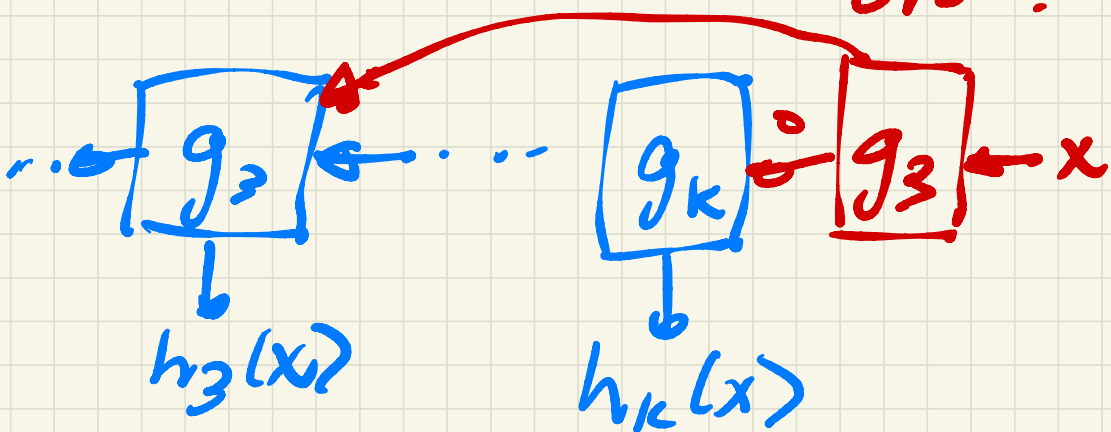


Now repeat/recurse:



Hmm... what if g_k

makes e.g. g_3 worse
off?



"pointer decision list"
(PDL)

Ends in one of three ways:

- $E(f_k) = 0$
- No improving (g, h) exists
- then $f_k = \text{Bayes optimal}$
- Nobody can find improving (g, h)
- effective Bayes optimal

Practicalities

- Must give crowd **data**, not distribution
- What about overfitting?
- One solution:
 - only accept when $w \cdot \Delta \geq \gamma \neq 0$
 - give **no info** when (g, h) rejected
 - can only be $1/\gamma$ accepts!
 - use test set to accept/reject

Algorithmic Framework

- Plug in your favorite algo for finding (g, h) improvements
- Ternary classification
- EM-style algo
- Others?

Trade-offs will return...

Other Learning Settings

Fairness in Bandits

- Ground truth data

$$\langle x, y \rangle$$

loan
app

$\in \mathbb{R}$, prob. of
repayment

- Unknown linear map

$$y = \theta \cdot x + \text{noise}$$

(linear regress.)

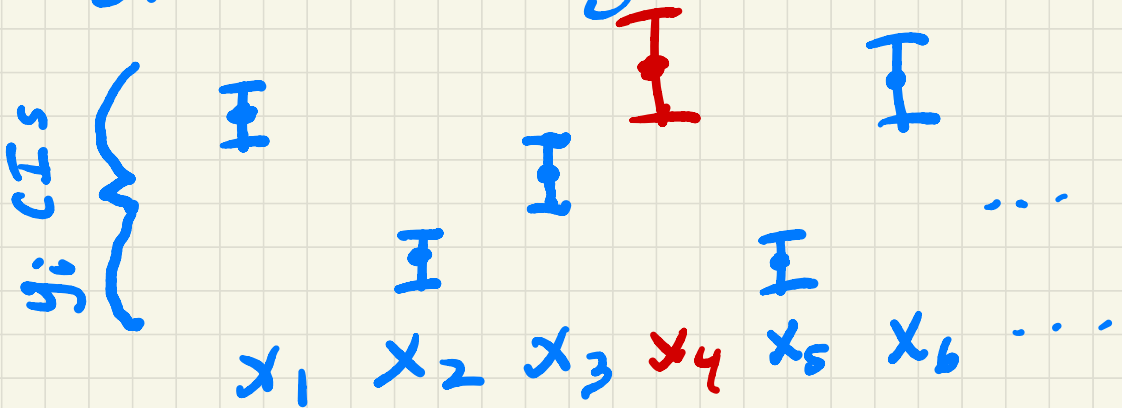
- Meritocratic fairness:

If $y_1 \geq y_2$, must have

prob. of loan to $x_1 \geq$ prob. of loan to x_2

• Bandit setting: each day x_1, \dots, x_k arrive, must choose loans **fairly**

• Standard algo: LIN-UCB

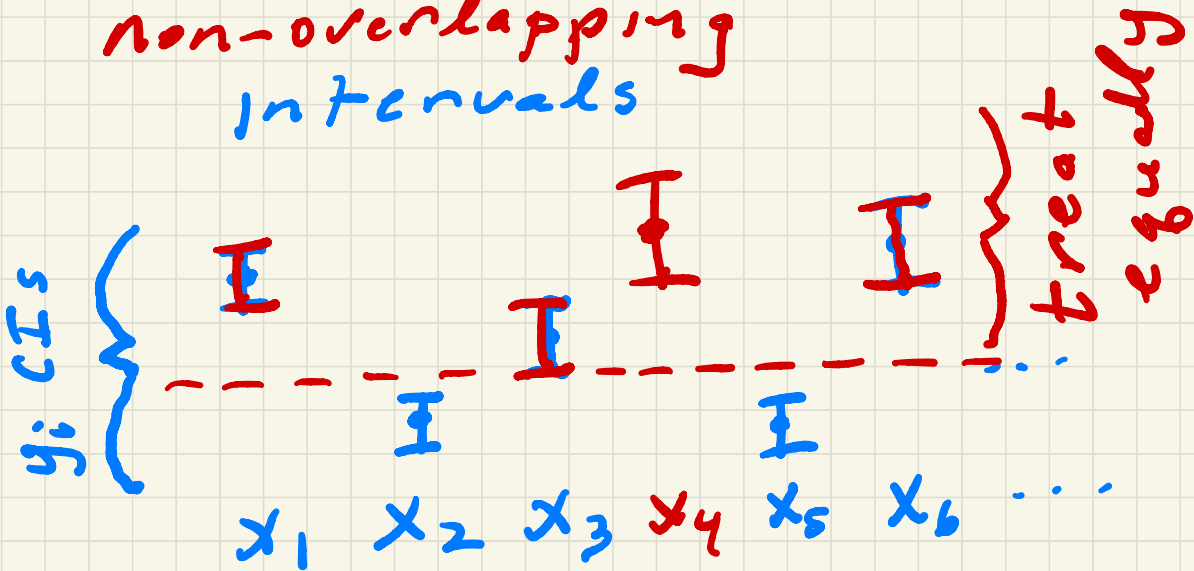


Give loan(s) to highest UCBs \Rightarrow fast convergence to opt

Not fair

Fair Modification

- Interval chaining
- May even choose non-overlapping intervals



- choose interval
⇒ more data
⇒ chaining fragment
⇒ fast convergence to opt

Other Topics

- Fair RL
(e.g. meritocratic
wrt Q-values)
- Fair Representations
- Causal Approaches
- Fair Clustering
- Fair Rankings
-

Some Resources

- "Frontiers of Fairness in Machine Learning"
Chouldechova & Roth
- "Fairness and ML"
Barocas, Hardt, Narayanan
fairmlbook.org
- "The Ethical Algorithm"
Kearns & Roth

Privacy in ML

What Do We Want?

- Not addressing preventing data breaches, unwanted access, etc - domains of cryptography and security
- Rather, prevent inferences and exfiltration from trained model

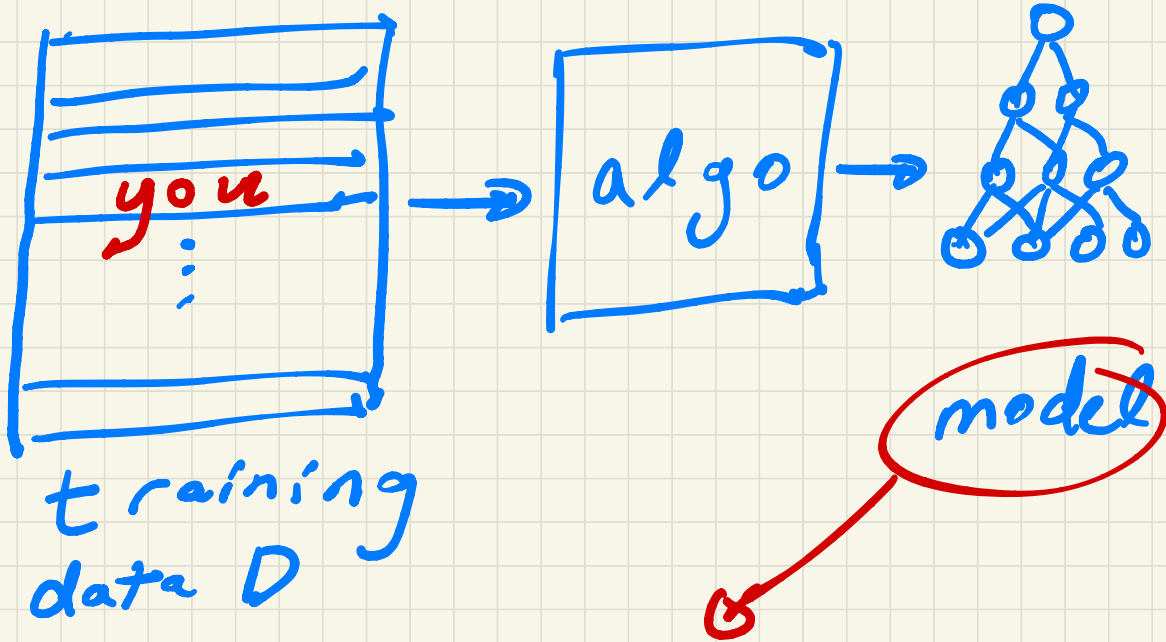
(Bad) Examples

- k-NN models
- SVMs
- Neural Networks
- Any model with confidence ratings

• Even **black-box** access problematic

• "Anonymizing" data **doesn't work**

High-Level Idea



Shouldn't reveal
"anything" about
your data - even
with additional
computation & data

Differential Privacy

Say algo A is ϵ -DP if

\forall neighboring D, D'

\forall set $S \subseteq \text{range}(A)$:

$$\Pr[A(D') \in S] \leq e^\epsilon \Pr[A(D) \in S]$$

\hookrightarrow wrt randomization
of A only

