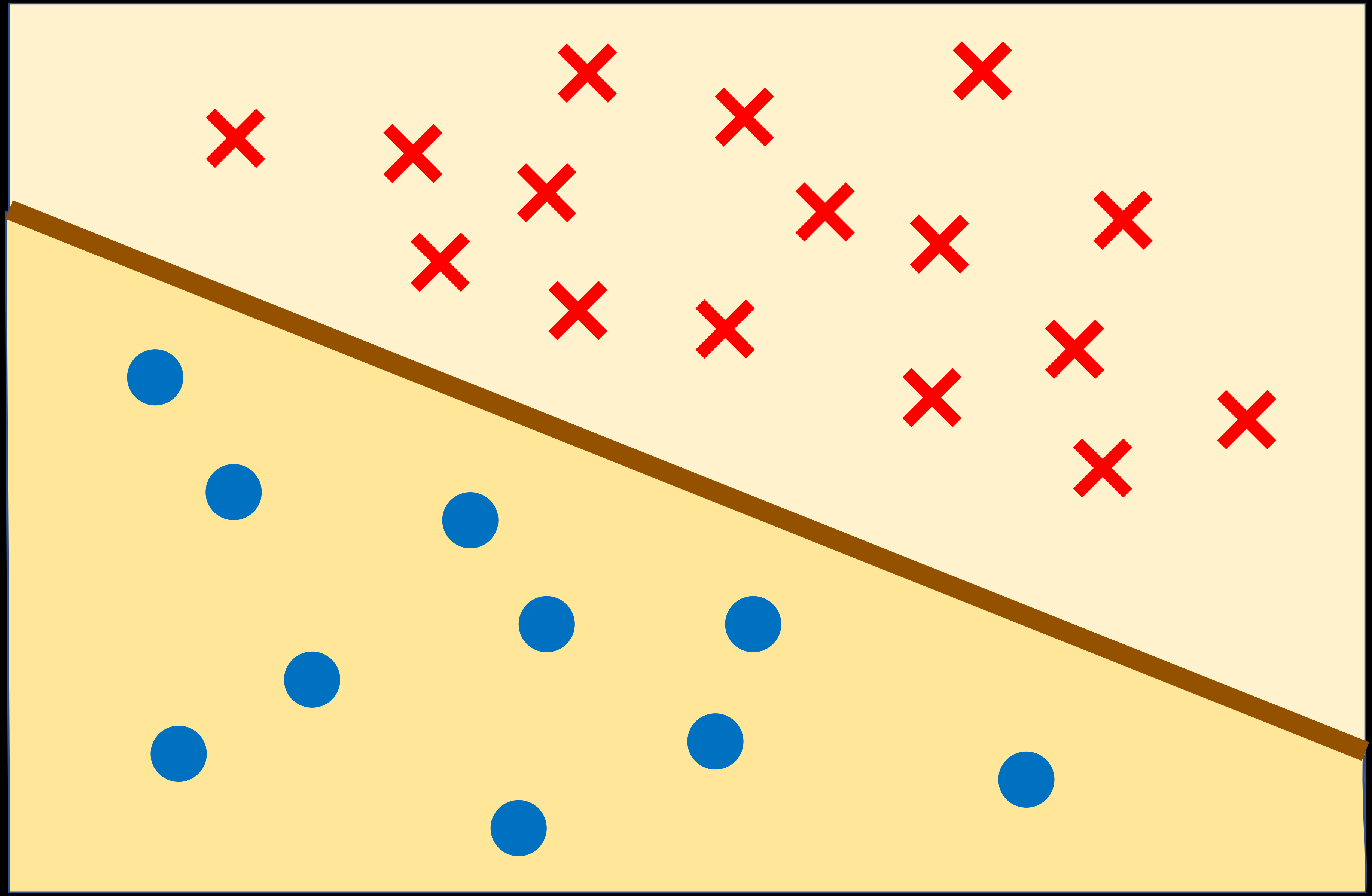
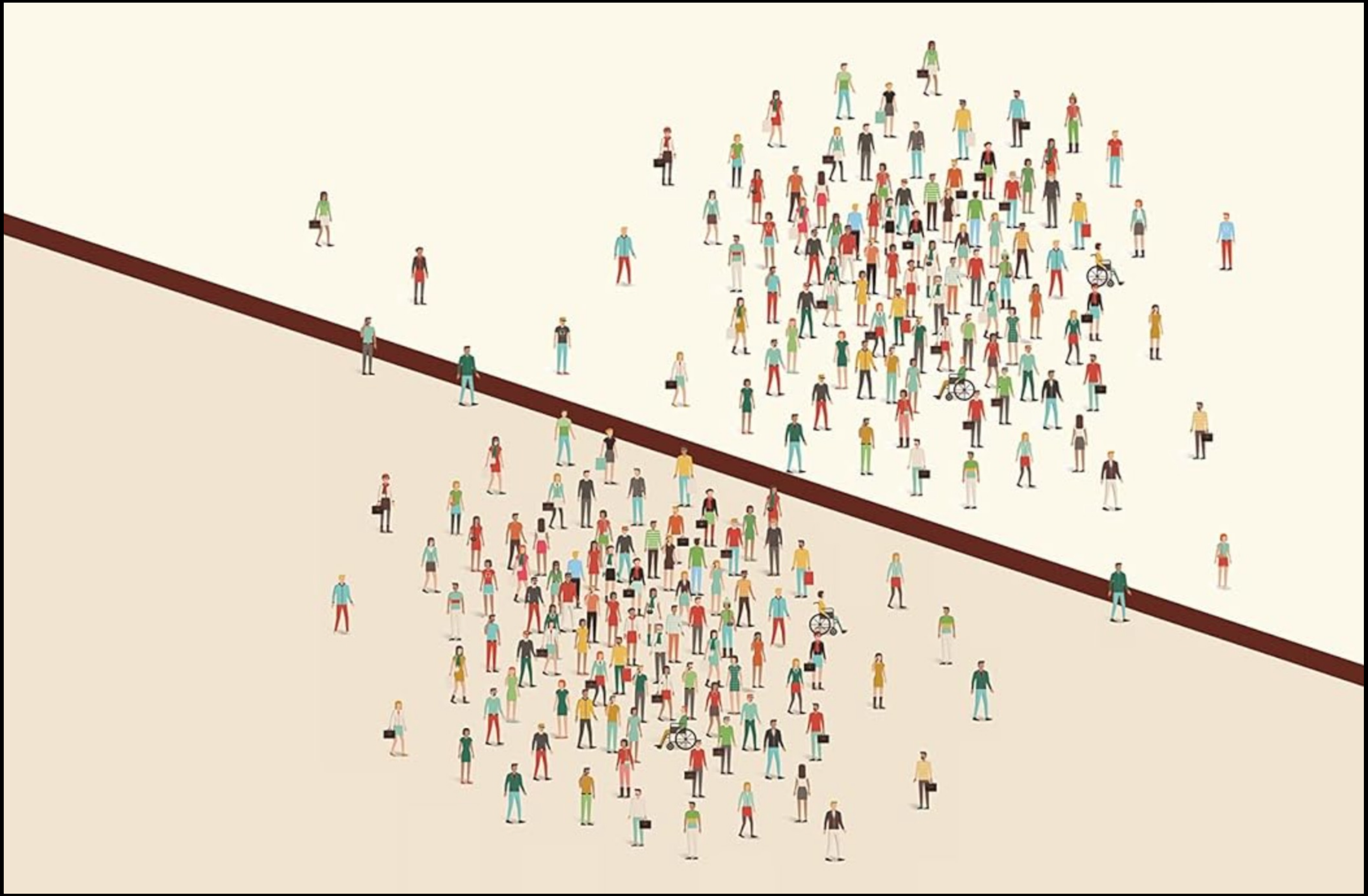
A multigroup indistinguishability perspective to go beyond loss minimization in ML

Vatsal Sharan
USC

Loss minimization: Find predictor to minimize some loss on average

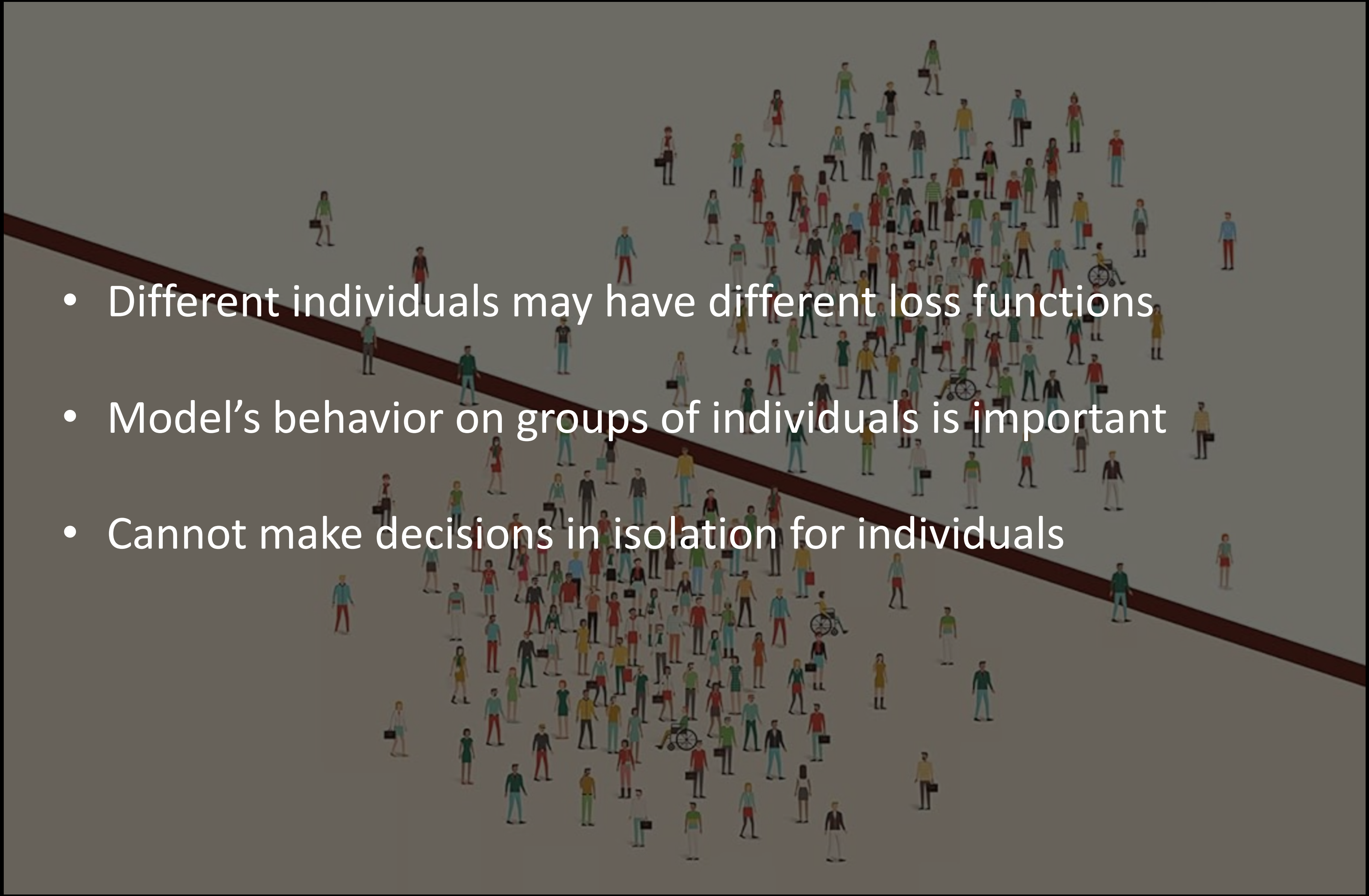


Reality: Predictions affect individuals



Pic: Patterns, Predictions, and Actions. Moritz Hardt, Ben Recht

Reality: Predictions affect individuals



- Different individuals may have different loss functions
- Model’s behavior on groups of individuals is important
- Cannot make decisions in isolation for individuals

Pic: Patterns, Predictions, and Actions. Moritz Hardt, Ben Recht

Loss minimization

Distribution D on $X \times \{0,1\}$

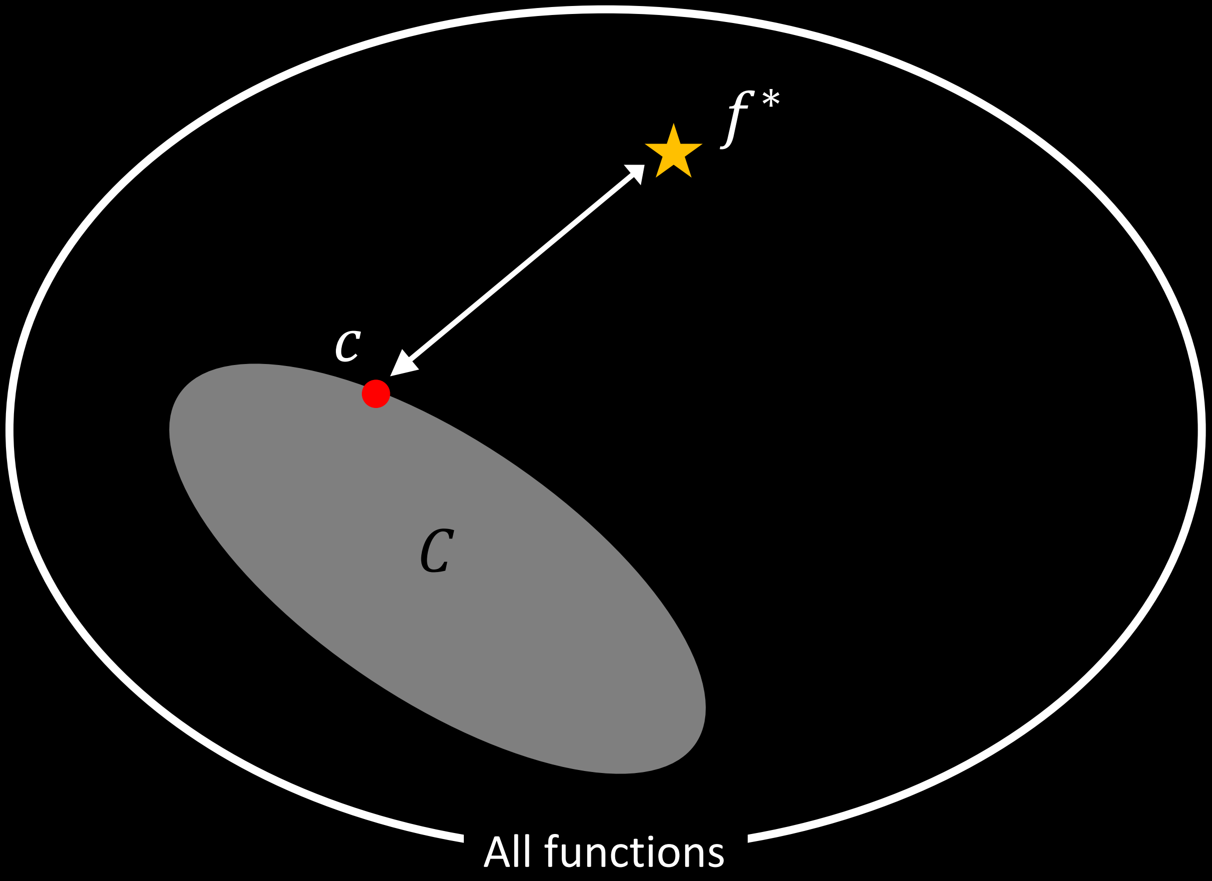
- Bayes optimal predictor: $f^*(x) = \Pr[y = 1|x]$.

A class C of hypotheses

- Hypothesis class $C = \{c: X \ \rightarrow \ R\}$.

A loss function ℓ:

- Given true label y, predict $p \in R$, suffer a loss $\ell(y, p)$.



Loss Minimization:

Find $c \in C$ minimizing $E[\ell(y, c(x))]$.

Which loss function to use?

Loss Minimization:
Find $c \in C$ minimizing $E[\ell(y, c(x))]$.

Proper losses:

- Squared loss $\ell_2(y,p) = (y-p)^2$
- Cross entropy loss $\ell_{ce}(y,p) = y \log p + (1-y)\log(1-p)$

If $y \sim Ber(p)$, best action is p

Improper losses:

- ℓ_1 loss $\ell_1(y,p) = |\, y\ - p\,|$
- Different false positive/negative costs

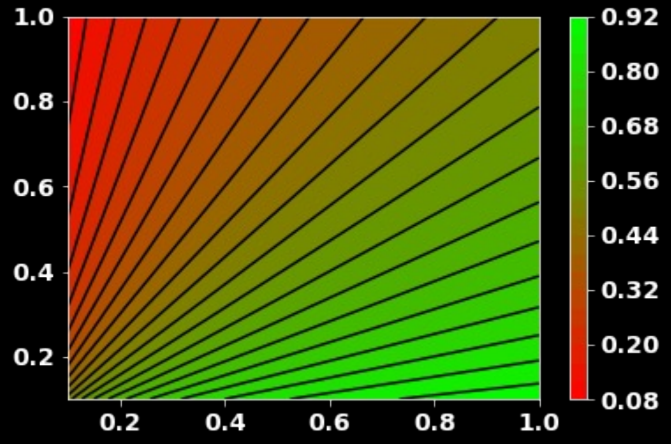
If $y \sim Ber(p)$, best action is $k_\ell(p) \neq id\ (p)$

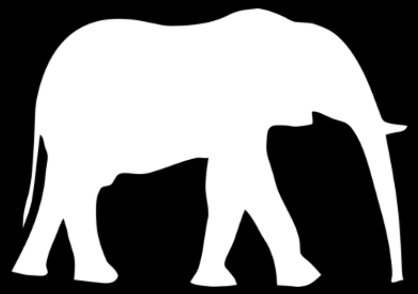
Different loss functions can lead to different models

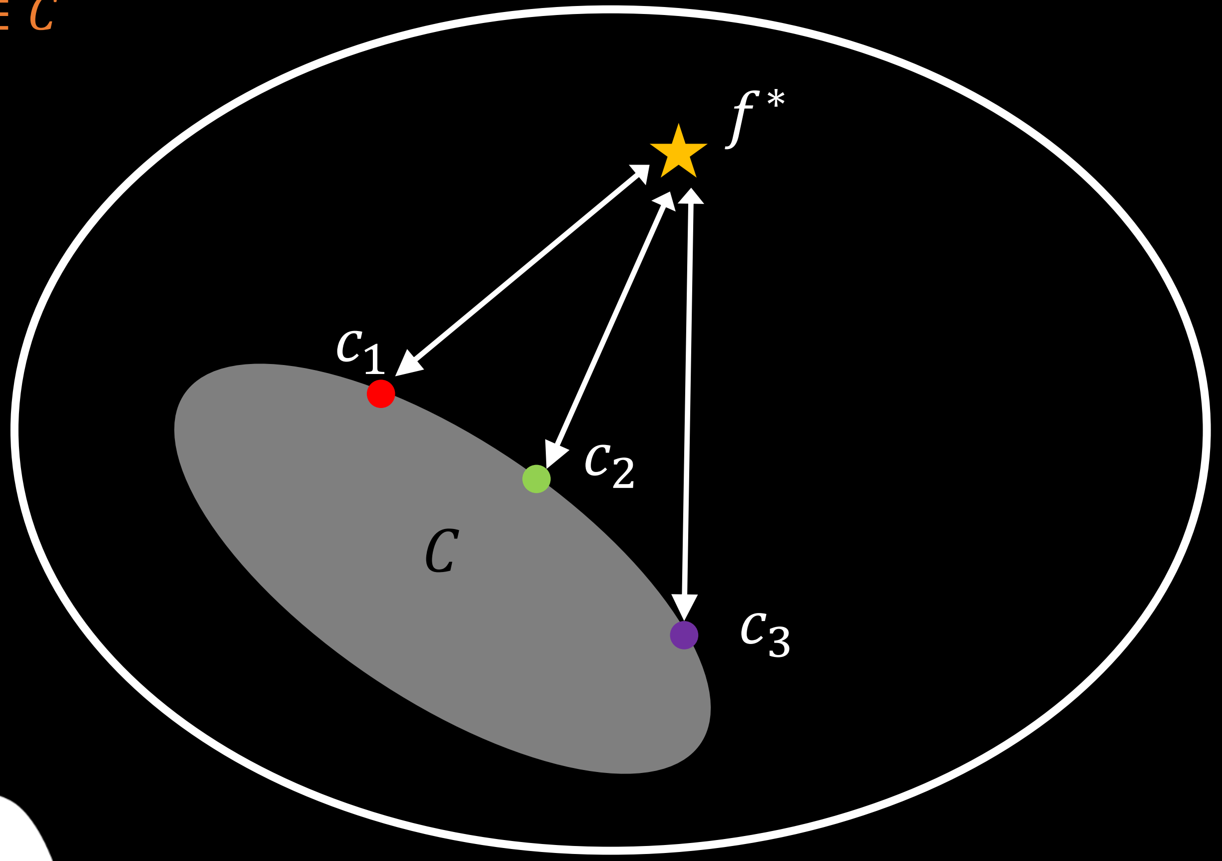
Different loss functions can produce different models $c \in C$

Models obtained for some loss could lose relevant information for minimizing another loss

E.g. binary classification with different false positive/negative costs







There may not be one relevant loss function

- May not know the 'correct' loss function at time of learning (what medical interventions will be used?)
- May want to learn for multiple, varied loss functions (aspirin vs surgery?)
- May want to learn now for future, yet unknown loss functions (future medical intervention?)

ML models are increasingly used to compute individual risk scores (e.g. heart disease, recidivism, dropping out of school etc.) which could be used for multiple interventions

If we had true probabilities from f^*, could post-process for any loss/downstream decision.

Can we get similar guarantees, without having to learn f^*?

OMNIPREDICTORS

Parikshit Gopalan
Apple

Adam Tauman Kalai
OpenAI

Omer Reingold
Stanford

Udi Wieder
Apple

OMNIPREDICTORS

L: family of loss functions C: hypothesis class

Def: An (L, C)-**omnipredictor** is $f: X \to [0,1]$ such that for every $\ell \in L$,

$$E[\ell(y, k_\ell(f(x)))] \leq \min_{c \in C} E[\ell(y, c(x))]$$

Learn once for all L, post-process later for any $\ell \in L$ using k_ℓ

Bayes-opt f^* is an omnipredictor for all (L, C)

Can we efficiently learn omnipredictors for rich (L, C)?

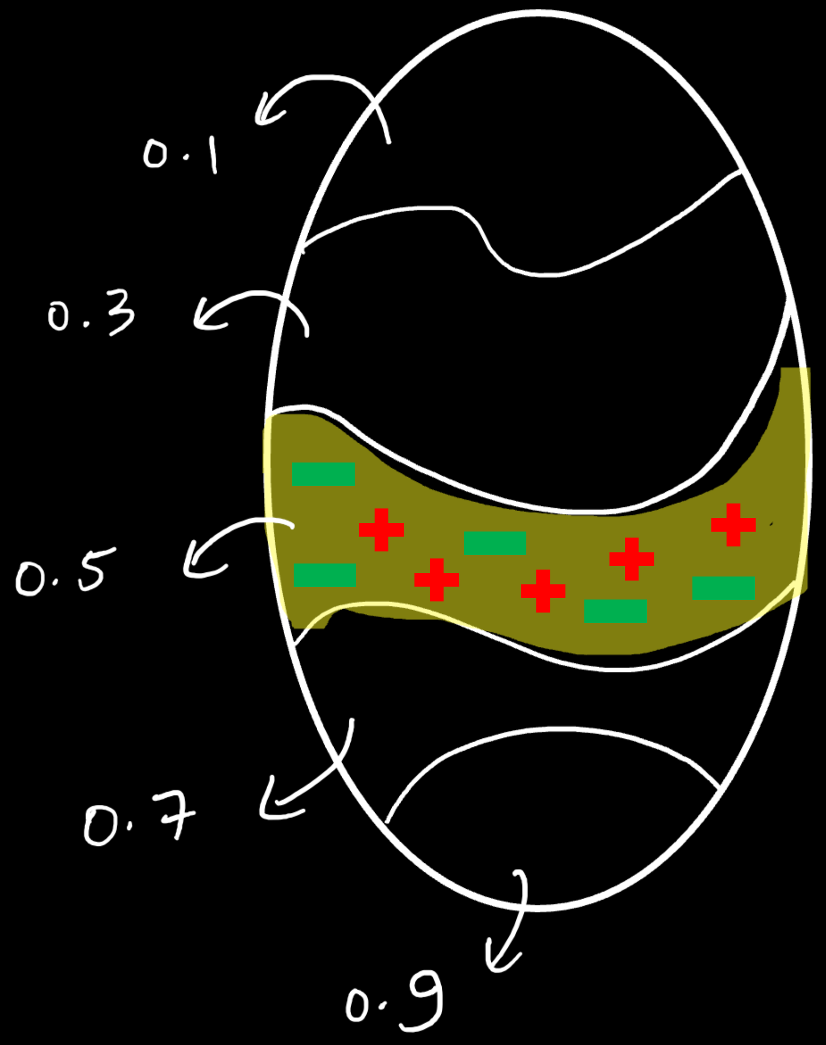
Multicalibration

[HebertJohnson-Kim-Reingold-Rothblum, ICML'18]

A notion of multigroup fairness.

Calibration [Dawid, AoS'85] The predictor f is calibrated if $E_D[y \mid f(x) = v] = v$.

"Predictions mean what they say"



Multicalibration

[HebertJohnson-Kim-Reingold-Rothblum, ICML'18]

A notion of multigroup fairness.

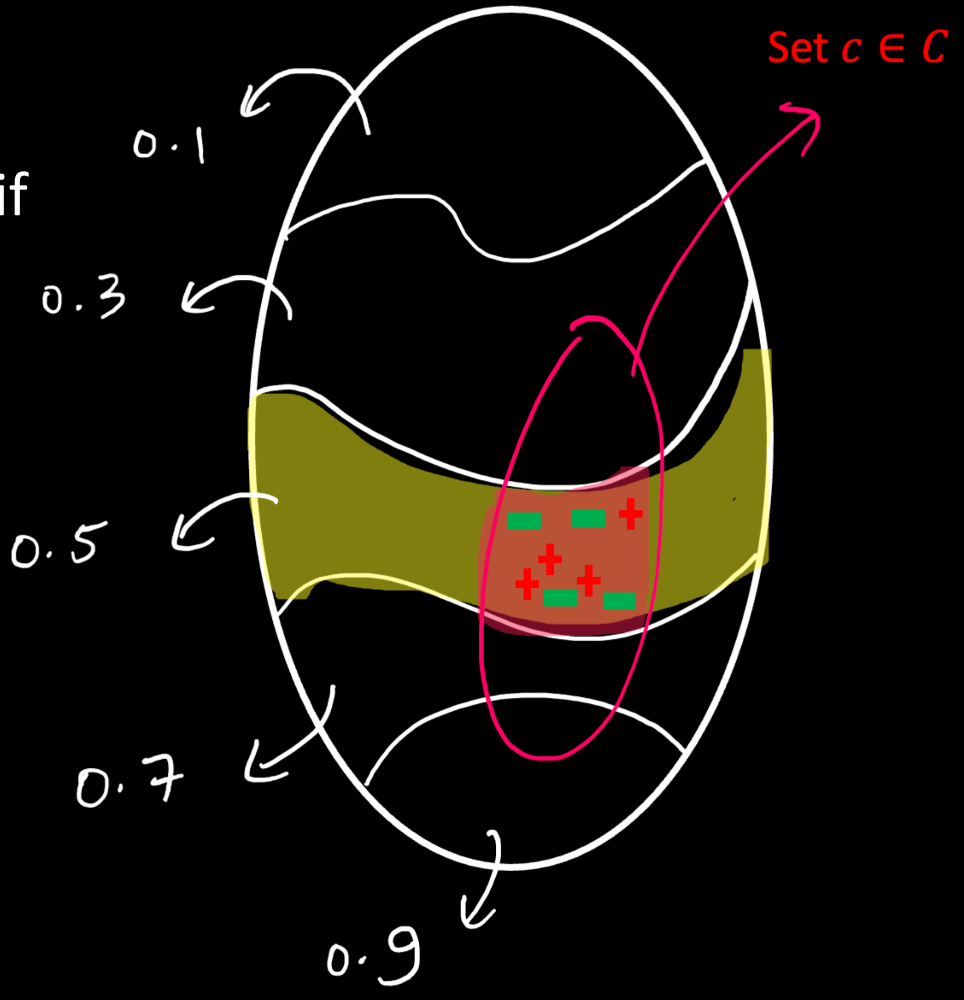
Calibration [Dawid, AoS'85] The predictor f is calibrated if $E_D[y| f(x) = v] = v$.

"Predictions mean what they say"

Multicalibration [HKRR'18]: Consider a class of Boolean valued functions C. f is multicalibrated for C if it is calibrated conditioned on every $c \in C$,

$$E_D[y| f(x) = v, c(x) = 1] = v$$
$$\Leftrightarrow E_D\,[c(x)(y - v)| f(x) = v] = 0$$

- C captures sub-populations we wish to protect.



Multicalibration

[HebertJohnson-Kim-Reingold-Rothblum, ICML'18]

A notion of multigroup fairness.

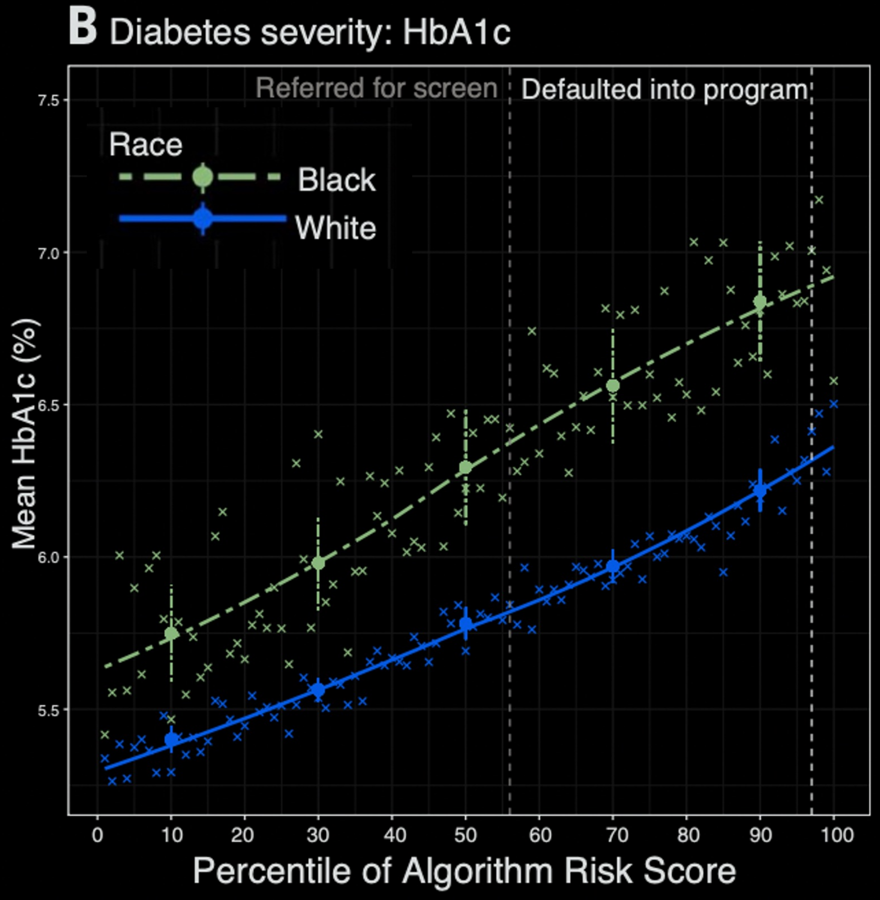
Calibration [Dawid, AoS'85] The predictor f is calibrated if $E_D[y|\, f(x) = v] = v$.

"Predictions mean what they say"

Multicalibration [HKRR'18]: Consider a class of Boolean valued functions C. f is multicalibrated for C if it is calibrated conditioned on every $c \in C$,

$$E_D[y|\, f(x) = v, c(x) = 1] = v$$
$$\Leftrightarrow E_D\,[c(x)(y - v)|\, f(x) = v] = 0$$

- C captures sub-populations we wish to protect.



Dissecting racial bias in an algorithm used to manage the health of populations, Obermeyer et al., Science 2019

Omnipredictors from Multicalibration

Def: An (L, C)-**omnipredictor** is $f: X \to [0,1]$ such that for every $\ell \in L$,

$$E[\ell(y, k_\ell(f(x)))] \leq \min_{c \in C} E[\ell(y, c(x))]$$

Thm: If f is multicalibrated for C, then it is an $(L_{cvx}, Lin(C))$-omnipredictor where

- L_{cvx} is all convex, Lipschitz loss functions
- $Lin(C) = \{\sum_i \lambda_i c_i\}$

Post-processing function k_ℓ same as for Bayes-opt f^*

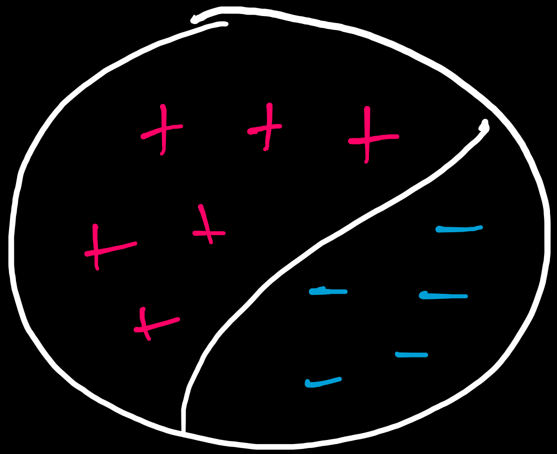
Using omnipredictor, can get:

- ℓ_2 loss: Linear regression:
- ℓ_1 loss: Linear programming [Kalai-Klivans-Mansour-Servedio'05]
- Cross-entropy loss: Logistic regression
- Exponential loss: Adaboost [Freund-Shapire'98]
- ..

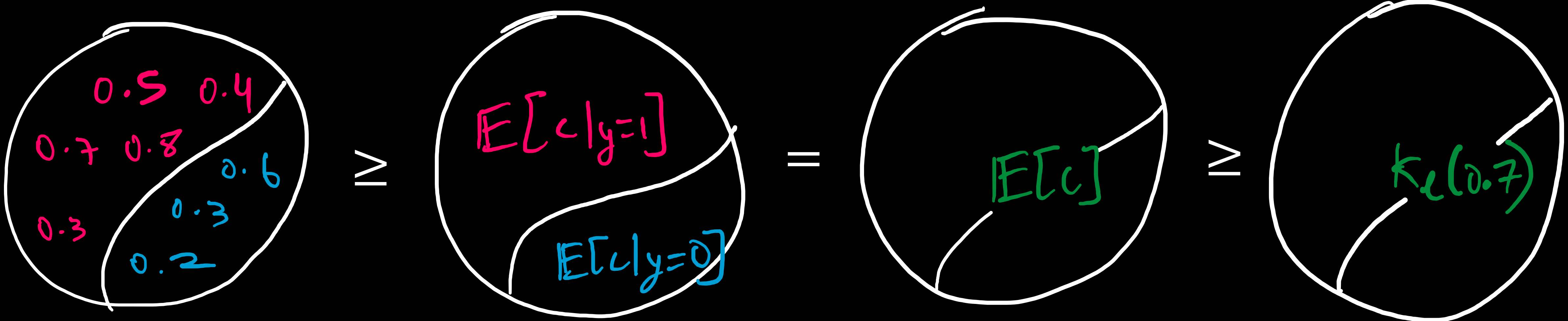
Proof sketch

Simplifying assumptions:

- f^* is Boolean
- Perfect multicalibration
  $E_D[c(x)(y-v) \mid f(x)=v] = 0$



$y \mid f(x) = 0.7$



$c \mid f(x) = 0.7$

Jensen's, convexity of loss

Multicalibration ⇒ no correlation with y

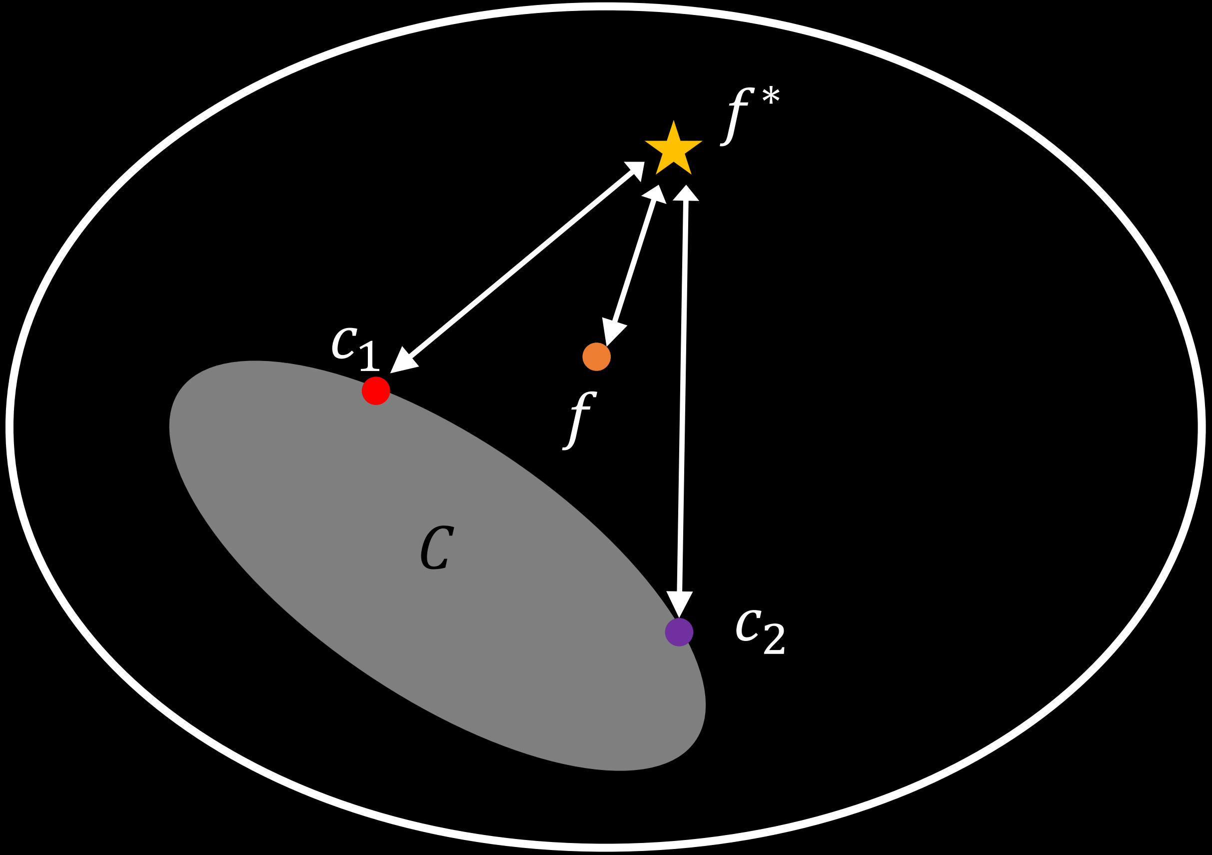
$y \sim Ber(0.7)$, definition of k_ℓ

Omnipredictors from Multicalibration

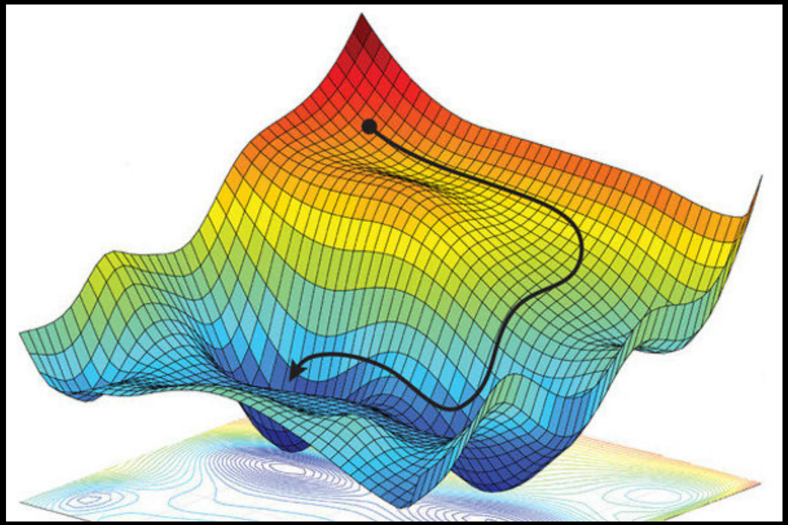
Def: An (L, C)-**omnipredictor** is $f: X \to [0,1]$ such that for every $\ell \in L$,

$$E[\ell(y, k_\ell(f(x)))] \le \min_{c \in C} E[\ell(y, c(x))]$$

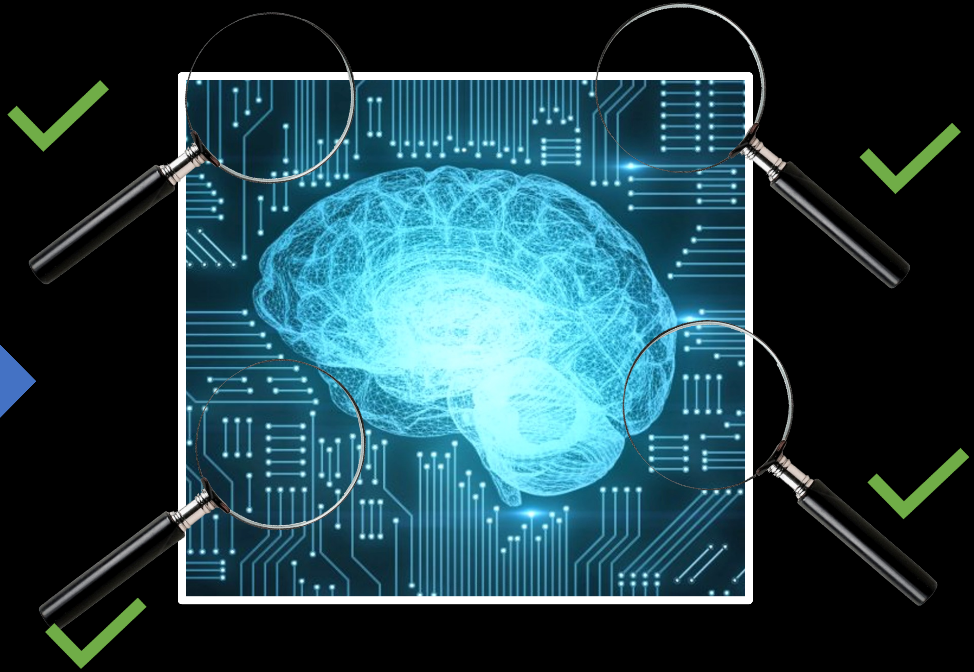
Thm: If f is multicalibrated for C, then it is an $(L_{cvx}, Lin(C))$-omnipredictor.



Indistinguishability from nature as a learning paradigm

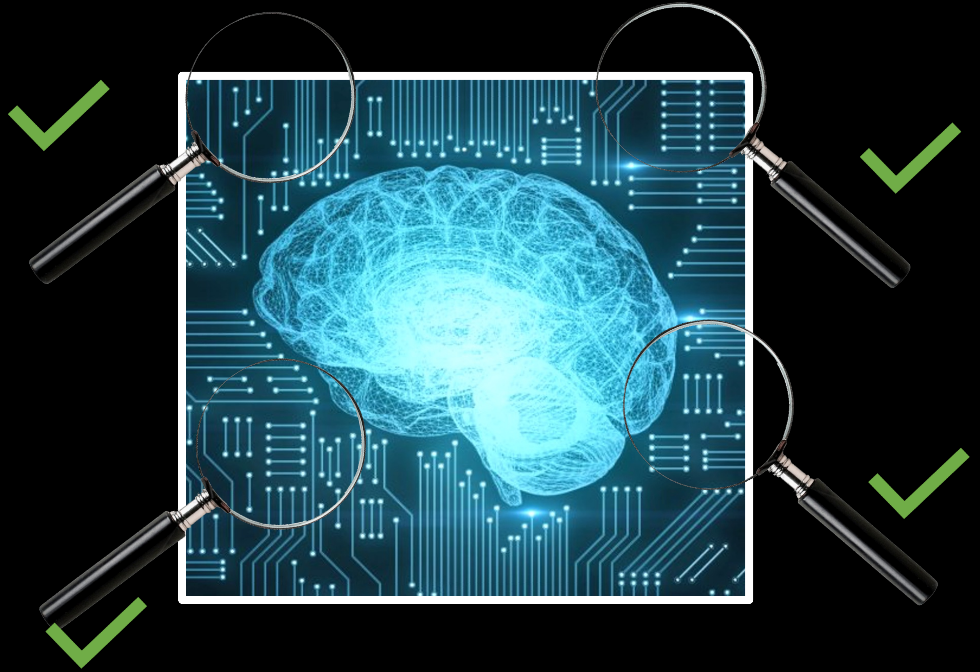


Optimize a single objective



Learn to fool class of tests
Optimize later (for some objective)

Indistinguishability from nature as a learning paradigm



Formalized in work of Dwork-Kim-Reingold-Rothblum-Yona, STOC’21 “Outcome indistinguishability”

Gopalan-Hu-Kim-Reingold-Wieder, ITCS’23 formally relate this to omniprediction

Fair rankings under composition, using indistinguishability

Joint work with:

Siddartha Devic

USC

David Kempe

USC

Aleksandra Korolova

Princeton

Stability and Multigroup Fairness in Ranking with Uncertain Predictions, ICML'24

Fair rankings under composition, using indistinguishability

Rank n candidates for a job:

x_1

x_2

…

x_n

$f(x)$ probability of x being qualified for job based on some model:

$f(x_1) = 0.3$ $\quad$ $f(x_2) = 0.6$ $\quad$ $f(x_n) = 0.6$

A ranking mechanism M takes as input $\{f(x_i): i \in [n]\}$, and produces a (randomized) ranking of $\{x_i: i \in [n]\}$

- Which ranking mechanisms M are fair?
- Which predictors f lead to fair rankings?
- Can the ranking inherit fairness of f?

Fair ranking mechanisms

Consider two universes.

Candidates: …

$f^*(x)$: 0.8 0.3 0.4 $\xrightarrow{M}$ Dist. over rankings

Candidates: …

Labels sampled from f^*:

✓ ✗ … ✗ $\xrightarrow{M}$ Dist. over rankings

✓ ✓ … ✗ $\xrightarrow{M}$ Dist. over rankings

⋮

✓ ✗ … ✓ $\xrightarrow{M}$ Dist. over rankings

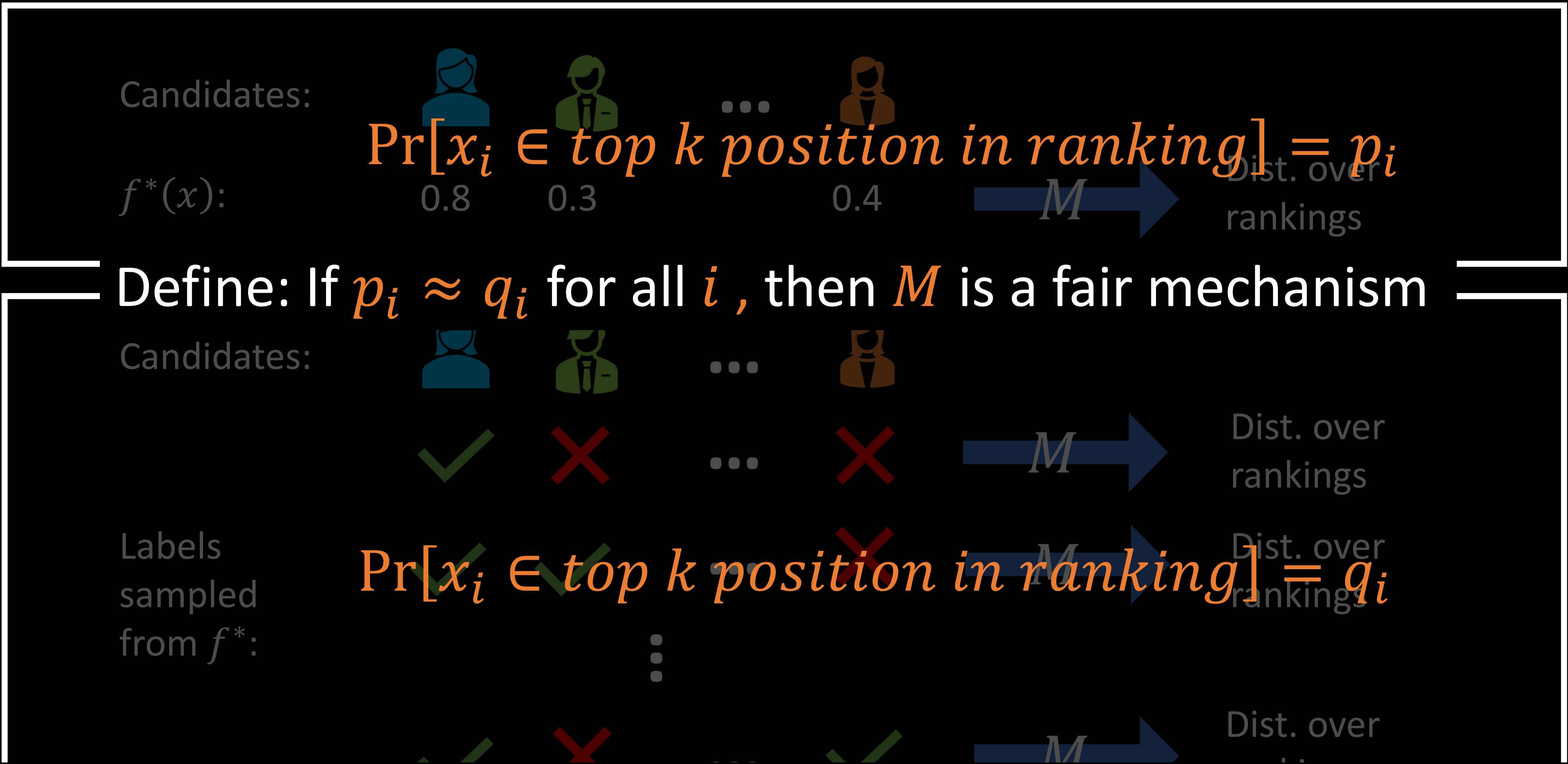
Fair ranking mechanisms

Consider two universes. For any k,

$$\Pr[x_i \in top\ k\ position\ in\ ranking] = p_i$$

Define: If $p_i \approx q_i$ for all i , then M is a fair mechanism

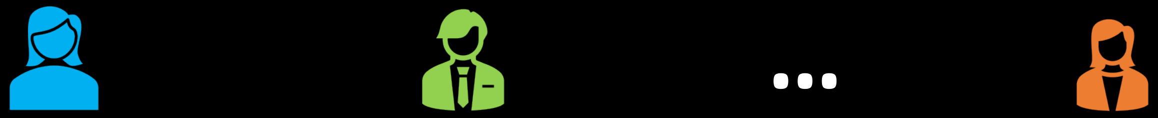
$$\Pr[x_i \in top\ k\ position\ in\ ranking] = q_i$$



Singh-Kempe-Joachims Neurips’21 introduced this, and a mechanism which satisfies this

When do fair predictors yield fair rankings?

Rank n candidates for a job:



Fair ranking mechanism M takes as input $\{f(x_i): i \in [n]\}$, and produces a (randomized) ranking of $\{x_i: i \in [n]\}$

Consider two universes.

$f(x)$ probability of x being qualified for job based on some model:	$f(x_1) = 0.3$	$f(x_2) = 0.6$	$f(x_n) = 0.6$

$f^*(x)$ is ground truth probability of x being qualified for job:	$f^*(x_1) = 0.3$	$f^*(x_2) = 0.6$	$f^*(x_n) = 0.6$

When do fair predictors yield fair rankings?

Rank n candidates for a job: …

Fair ranking mechanism M takes as input $\{f(x_i): i \in [n]\}$, and produces a (randomized) ranking of $\{x_i: i \in [n]\}$

Consider two universes. Set of groups C.

$f(x)$ probability of x being qualified for job based on some model: $f(x_1) = 0.3$ $f(x_2) = 0.6$ $f(x_n) = 0.6$

$\mathrm{E}[\#$ individuals from group $c \in C$ in top k position in ranking under $f]$

Definition: Rankings from f are multigroup fair w.r.t. set of groups C, if these expectations are ≈ same for any group $c \in C$

$f^*(x)$ is ground truth probability of x being qualified for job: $f^*(x_1) = 0.3$ $f^*(x_2) = 0.6$ $f^*(x_n) = 0.6$

$\mathrm{E}[\#$ individuals from group $c \in C$ in top k position in ranking under $f^*]$

When do fair predictors yield fair rankings?

Definition: Rankings from f are multigroup fair w.r.t. set of groups C, if for any group $c \in C$ and k,

$$\mathrm{E}[\#\text{ individuals from group } c \text{ in top } k \text{ position in ranking under } f\,] \approx \mathrm{E}[\#\text{ individuals from group } c \text{ in top } k \text{ position in ranking under } f^*\,]$$

Thm (informal): If f is multicalibrated for C, then rankings produced by f are multigroup fair w.r.t. to C.

Composition of fairness properties: ranking inherits fairness of predictors

Similar indistinguishability framework also yields fairness notions for *matching* problems



Fairness in Matching under Uncertainty ’23

Multicalibration in practice

Joint work with:

Dutch Hansen

USC

Siddartha Devic

USC

Preetum Nakkiran

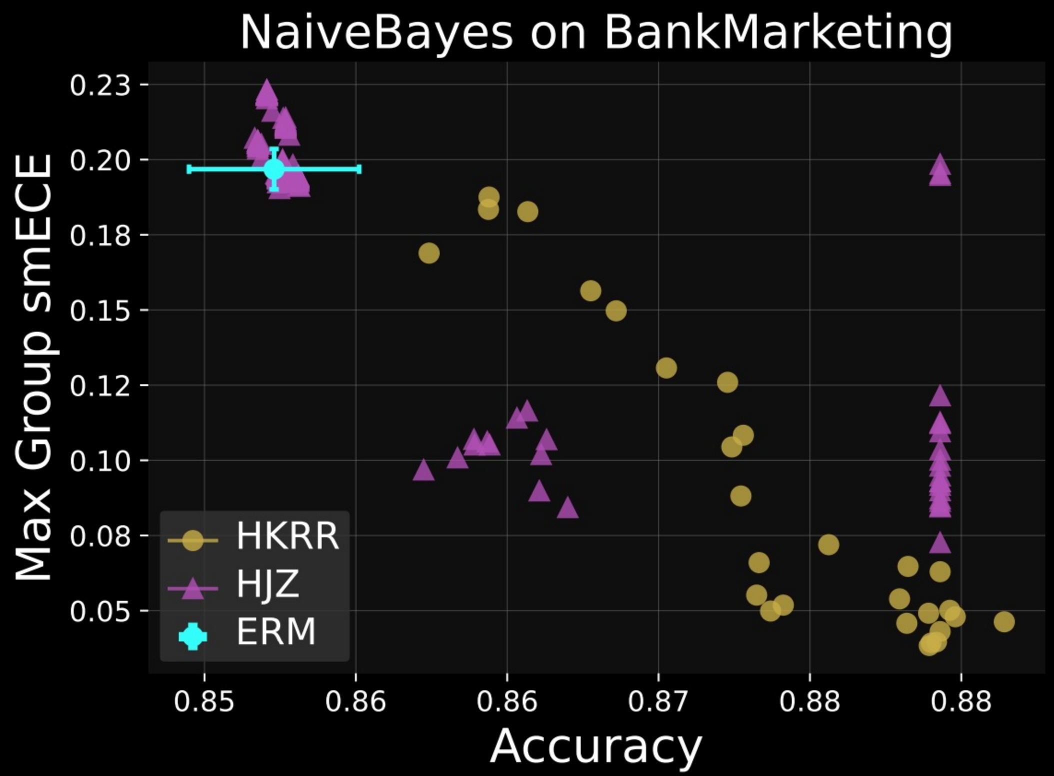
Apple

When is Multicalibration Post-Processing Necessary? ‘24

Large-scale evaluation of multicalibration

How multicalibrated are current models?

1. Multicalibration post-processing can help inherently uncalibrated models like SVMs, decision trees etc.

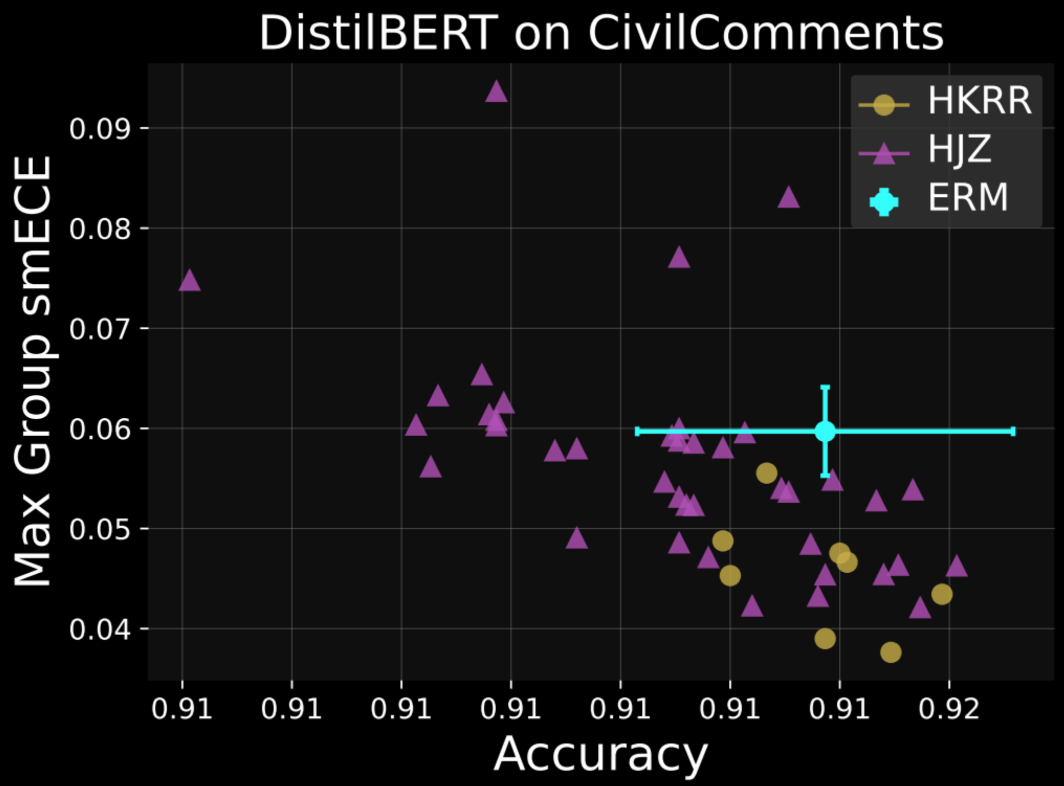


Large-scale evaluation of multicalibration

How multicalibrated are current models?

1. Multicalibration post-processing can help inherently uncalibrated models like SVMs, decision trees etc.

2. Deep neural networks tend to be relatively multicalibrated without additional post-processing



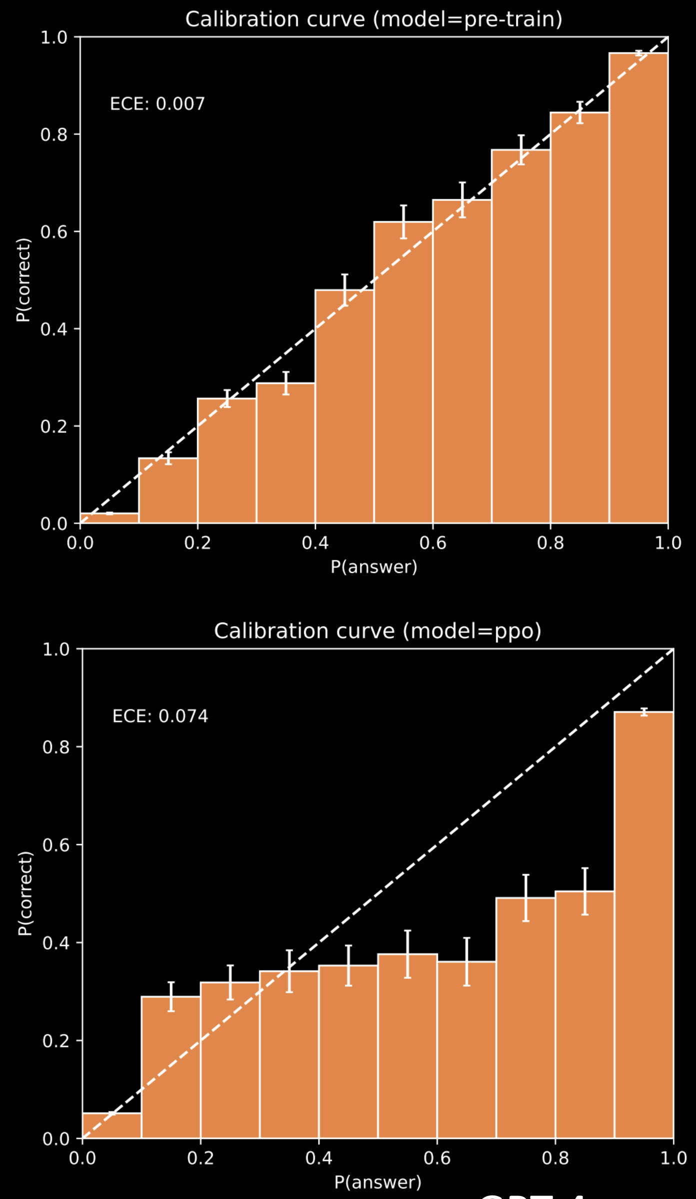
Large-scale evaluation of multicalibration

How multicalibrated are current models?

1. Multicalibration post-processing can help inherently uncalibrated models like SVMs, decision trees etc.

2. Deep neural networks tend to be relatively multicalibrated without additional post-processing

3. More scope of improving worst-case calibration error in settings where large models are fine-tuned



GPT-4 paper

Large-scale evaluation of multicalibration

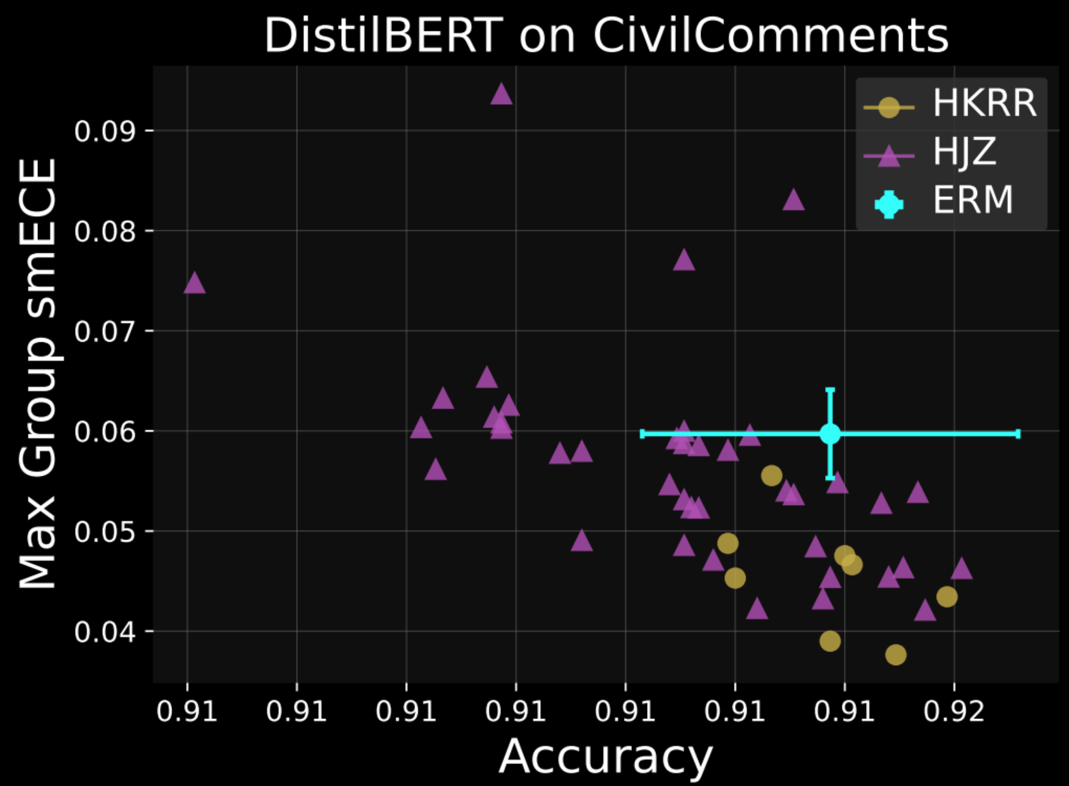
How multicalibrated are current models?

1. Multicalibration post-processing can help inherently uncalibrated models like SVMs, decision trees etc.

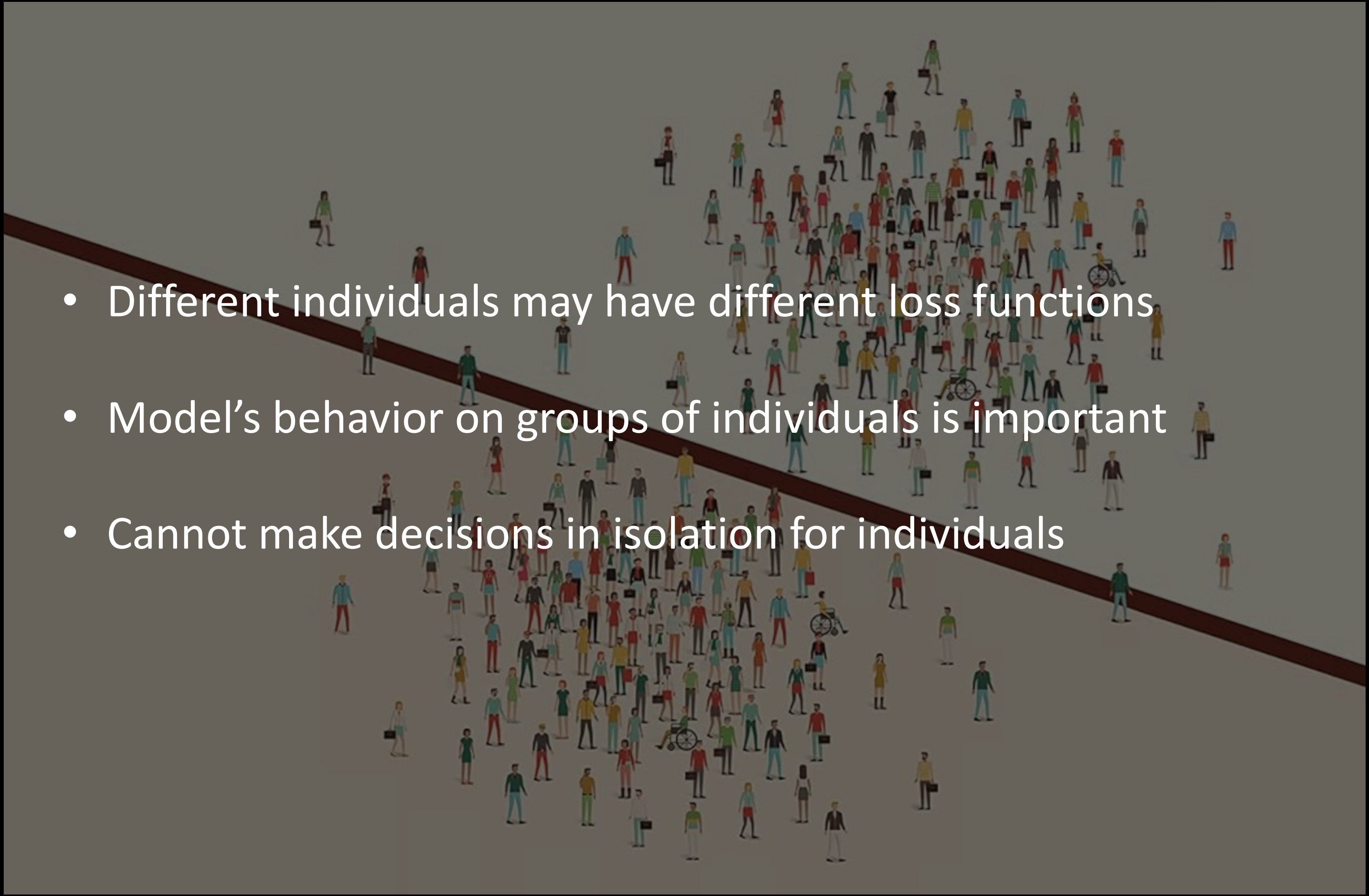
2. Deep neural networks tend to be relatively multicalibrated without additional post-processing

3. More scope of improving worst-case calibration error in settings where large models are fine-tuned

4. Plug-and-play and sample efficient post-processing techniques could help



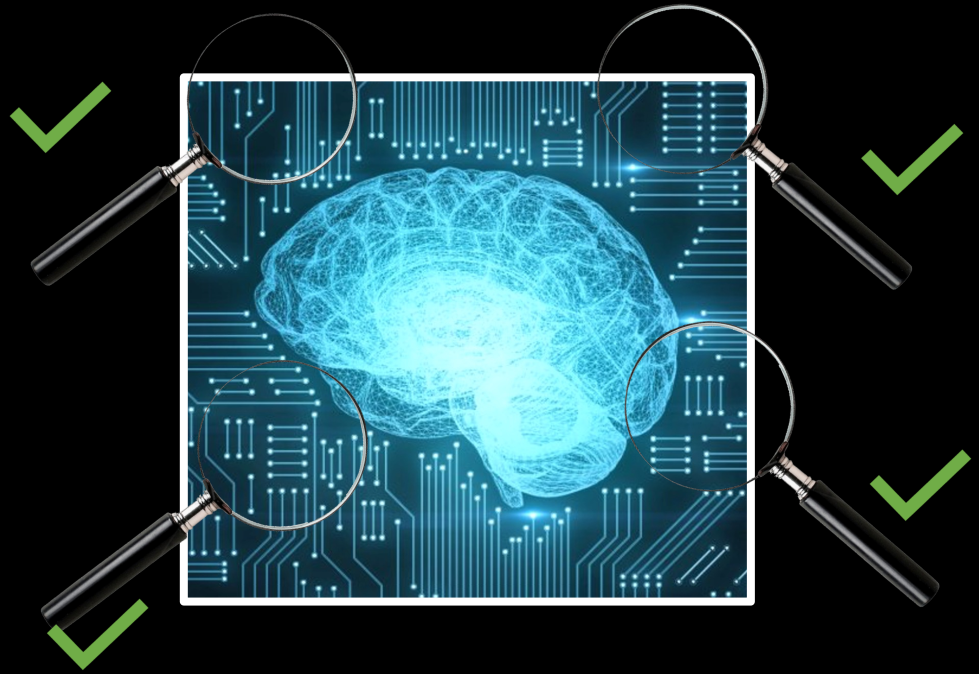
Reality: Predictions affect individuals



- Different individuals may have different loss functions
- Model's behavior on groups of individuals is important
- Cannot make decisions in isolation for individuals

Pic: Patterns, Predictions, and Actions. Moritz Hardt, Ben Recht

Indistinguishability from nature as a learning paradigm



- Can learn once for a large class of loss functions
- Can get fairness guarantees which compose nicely in settings with multiple individuals

Parikshit Gopalan

Apple

Adam Kalai

OpenAI

Omer Reingold

Stanford

Udi Wieder

Apple

Siddartha Devic

USC

David Kempe

USC

Aleksandra Korolova

Princeton

Dutch Hansen

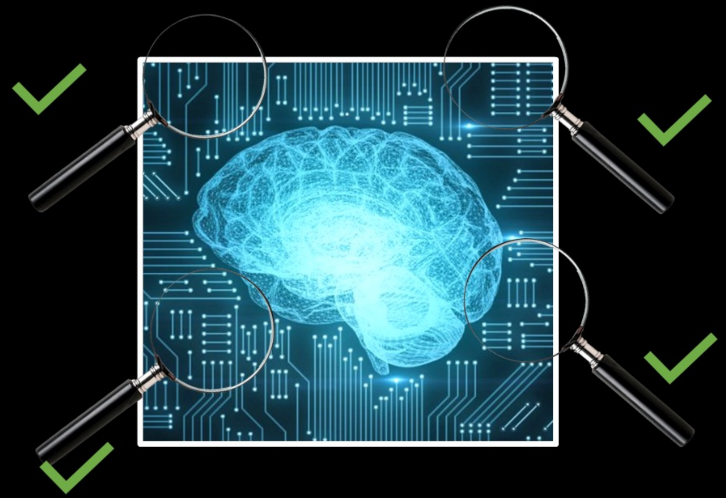
USC

Preetum Nakkiran

Apple

Thanks!

Indistinguishability from nature as a learning paradigm



- Can learn once for a large class of loss functions
- Can get fairness guarantees which compose nicely in settings with multiple individuals