Algorithm Design: Fairness and Accuracy

Annie Liang, Jay Lu, and Xiaosheng Mu (2021)

Presenter: Kyohei Okumura (Northwestern Econ)

Motivation

- algorithms are used to guide many high-stakes decisions
  - medical treatment/loan/bail/employment

- recent empirical evidence: algorithms often have errors that vary systematically across subgroups of the population

Example: algorithms in healthcare (Obeymeyer et al., 2019)

patients assigned to the same risk score have substantially different actual health risks depending on race

Fairness vs. Accuracy

- algorithms are increasingly optimized not only for accuracy but also “fairness”
- what is the tradeoff between fairness and accuracy?
- this paper characterizes a “fairness-accuracy frontier” that gives an insight into algorithm design for designers with different fairness concerns
  (not only for designer with a very specific optimization criterion)

Sample Question

University of California will no longer consider SAT and ACT scores as it may discriminate against applicants on the basis of their race, wealth, and disability.

When is it reasonable to ban some characteristics due to fairness concerns?

Framework

Setup

- single designer; population of (non-strategic) subjects
- each subject i is described by three variables:
  - type $Y_i \in \mathcal{Y}$
    (e.g., need for a medical treatment)
  - group $G_i \in \{r, b\}$
    (e.g., race, wealth, socioeconomic status)
  - covariate $X_i \in \mathcal{X}$
    (e.g., image scans, # of hospital visits, blood tests)
- X_i is observed by the designer; Assume $|\mathcal{X}| < \infty$
- Y_i is not directly observed; G_i may not be included in X_i.
- $(Y_i, G_i, X_i) \overset{\text{i.i.d.}}{\sim} P$ for some distribution P

Algorithm

- each subject i receives a decision $d_i \in \{0, 1\}$
  (e.g., whether the treatment is provided)

- the designer chooses an algorithm

$$a \colon \mathcal{X} \to \Delta(\{0, 1\})$$

  - $a(X_i)$ is the probability of i's getting the treatment

Group Errors

- fix a loss function $\ell(d, y, g)$ (real-valued)

Example: misclassification

$\mathcal{Y} := \{0, 1\}$. $\ell(d, y, g) := \mathbb{1}\{d \neq y\}$.

Def. (group error)

the error for group $g \in \{r, b\}$ given algorithm a is

$$e_g(a) := \mathbb{E}[\ell(a(X), Y, g) \mid G = g].$$

i.e., the average loss for subjects in group g.

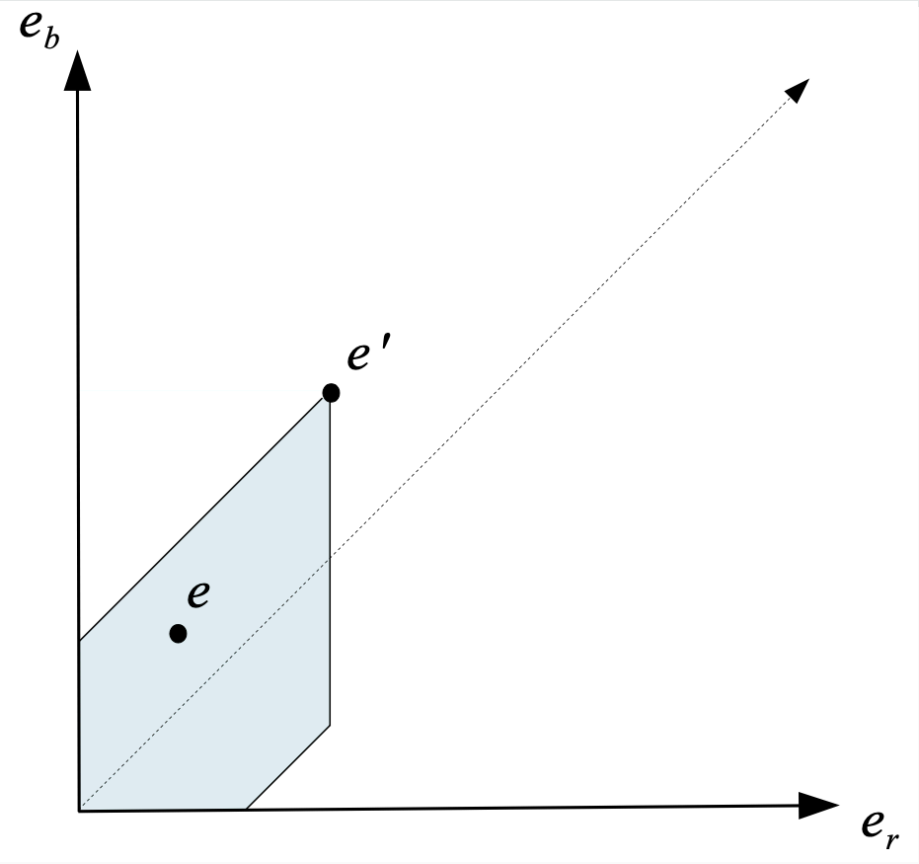
- Once we decide on algorithm a, a pair of group errors $(e_r(a), e_b(a))$ is determined.
- designer has a preference over error pairs (and algorithms).

Fairness-accuracy Dominance

Def. (fairness-accuracy (FA) dominance)

$$(e_r, e_b) >_{\text{FA}} (e'_r, e'_b) \overset{\Delta}{\iff} \begin{cases} e_r \leq e'_r, e_b \leq e'_b & \text{(higher accuracy)} \\ |e_r - e_b| \leq |e'_r - e'_b| & \text{(higher fairness)} \end{cases}$$

with at least one of these inequalities strict.



Preferences

- a preference $\succsim$ over $\mathbb{R}^2$ is consistent with FA-dominance if $e \succ e'$ whenever $e >_{\text{FA}} e'$.
- we assume that designer's preference is consistent with FA-dominance

Examples of FA-dominance consistent preferences

1. utilitarian (bayes-optimal)

$$\min_{e_r, e_b} -(p_r e_r + p_b e_b),$$

where p_g is the proportion of group g

2. egalitarian

$$\min_{e_r, e_b} |e_r - e_b|$$

3. rawlsian

$$\min_{e_r, e_b} \max\{e_r, e_b\}$$

4. constrained optimization

$$\min_{a} p_r e_r(a) + p_b e_b(a) \text{ s.t. } |e_r(a) - e_b(a)| \leq \varepsilon$$

Results:
Fairness-accuracy Frontier

Fairness-accuracy Frontier

Def. (feasible set)

the feasible set given X is

$$\mathcal{E}(X) := \{(e_r(a), e_b(a)) : a \in \Delta(\{0,1\})^{\mathcal{X}}\}.$$

i.e., the set of error pairs that is achieved by some algorithm.

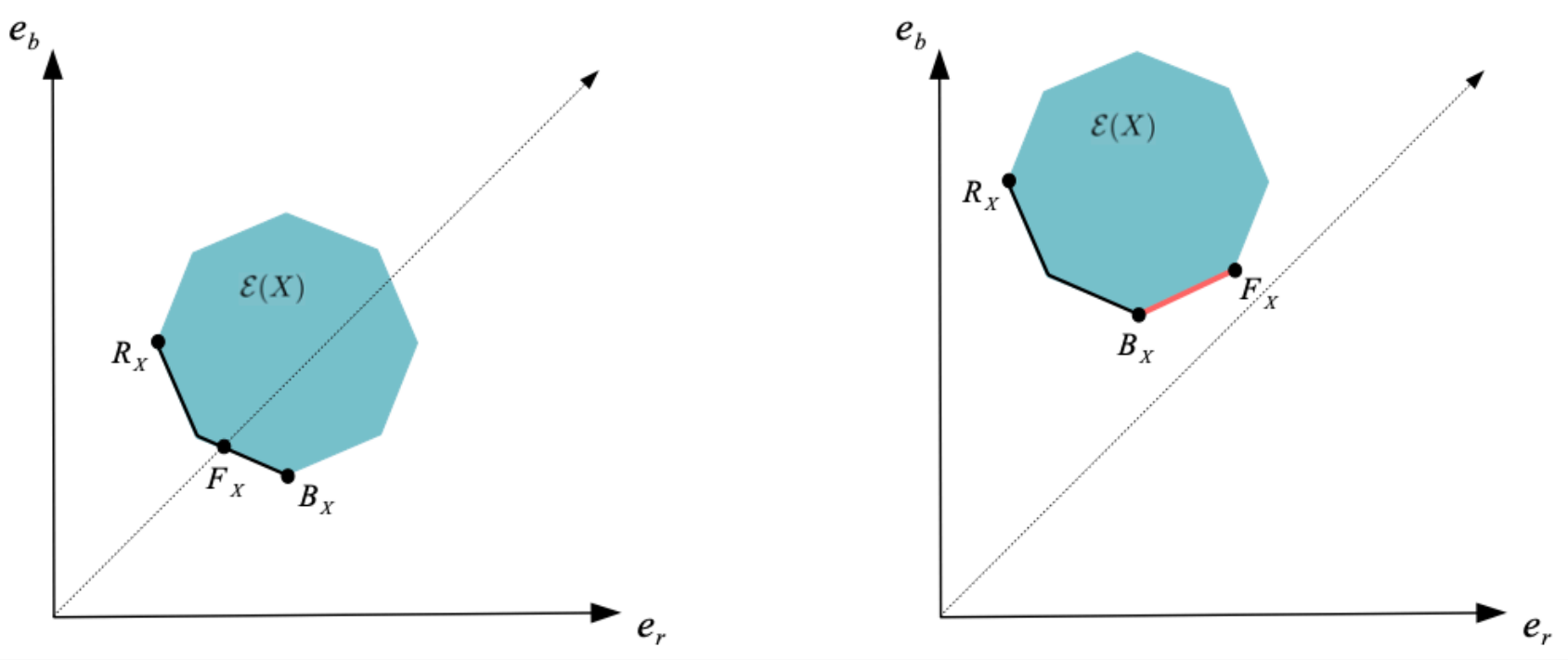
Def. (fairness-accuracy frontier)

the fairness-accuracy frontier given X is

$$\{e \in \mathcal{E}(X) : \nexists e' \in \mathcal{E}(X),\ e' >_{\text{FA}} e\},$$

i.e., the set of feasible points that are not FA-dominated by another feasible point.

Fairness-accuracy frontier

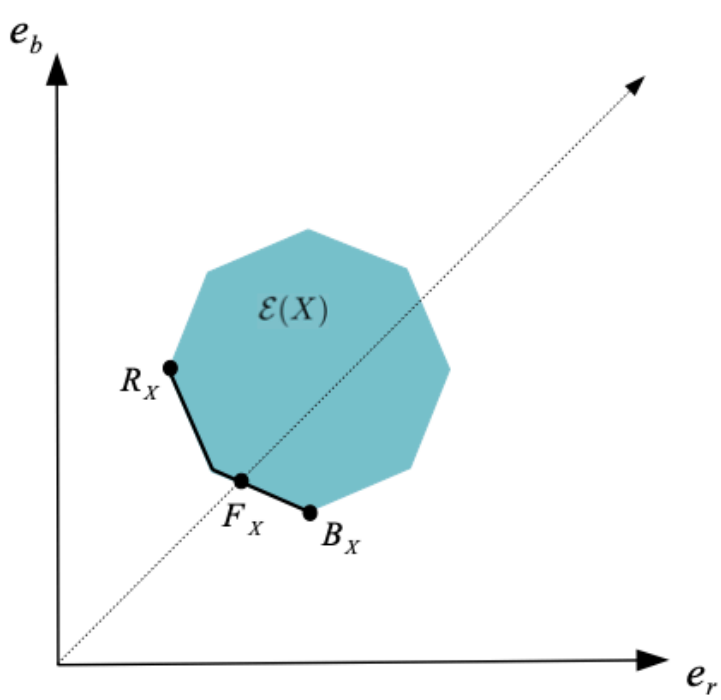


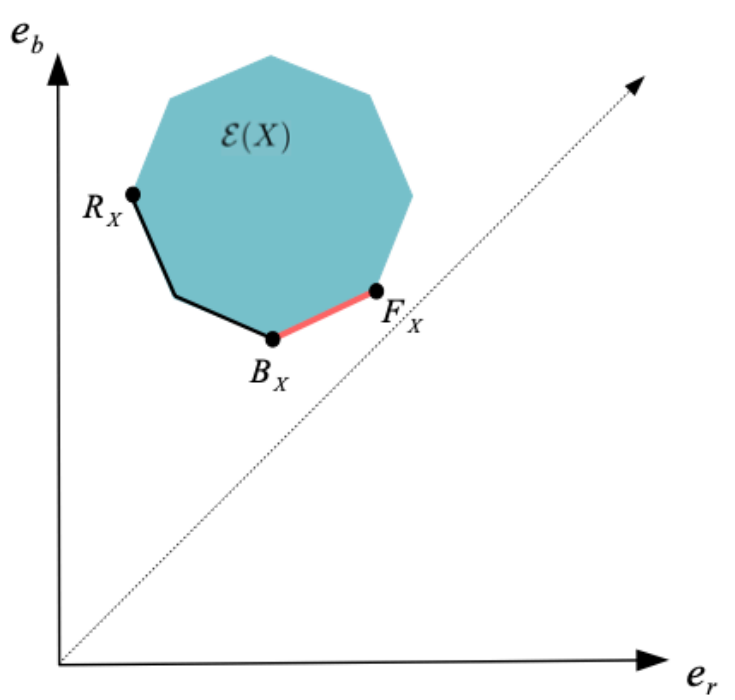
- **Fact:** the feasible set $\mathcal{E}(X)$ is closed and convex.

Def. (three important points)

- R_X: the feasible point that minimizes group r's error e_r
- B_X: the feasible point that minimizes group b's error e_b
- F_X: the feasible point that minimizes $|e_r - e_b|$

Group-skewed vs. Group-balanced



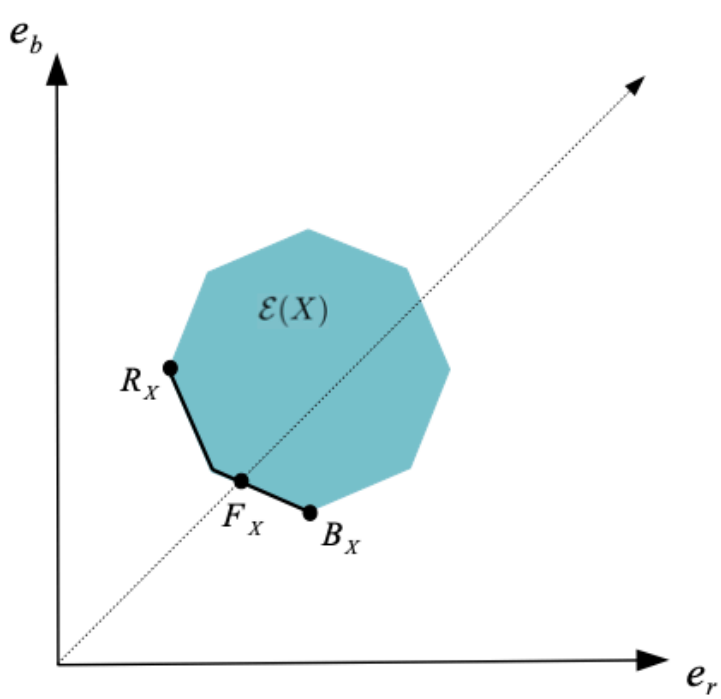


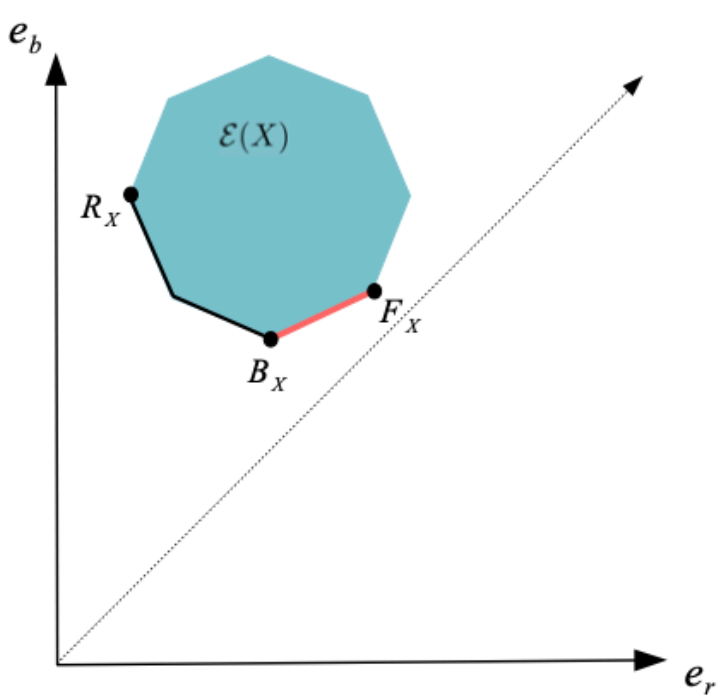
Def. (group-skewed/balanced)

covariate X is

- r-skewed if $e_r < e_b$ at R_X and $e_r \leq e_b$ at B_X.
  "group r's error is lower both at R_X and B_X"
- b-skewed if $e_b < e_r$ at B_X and $e_b \leq e_r$ at R_X.
- group-balanced otherwise

Characterization of FA frontier



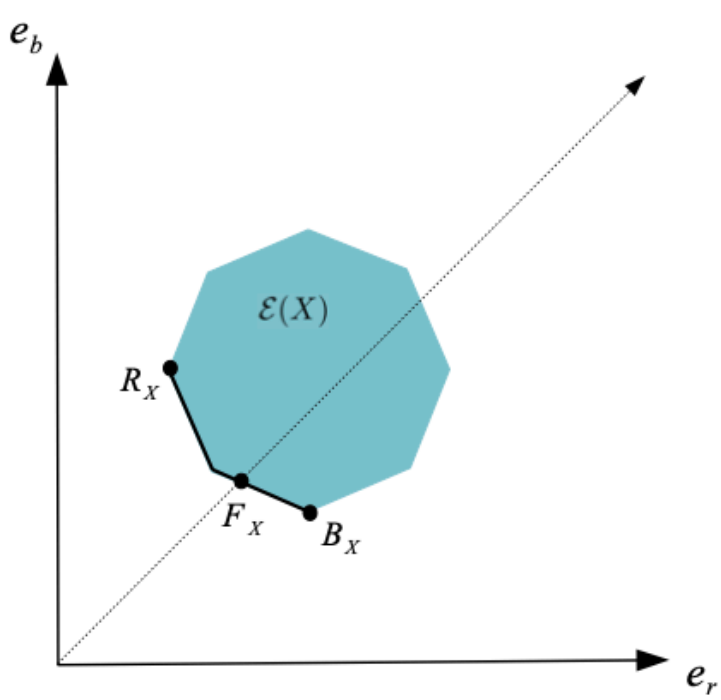


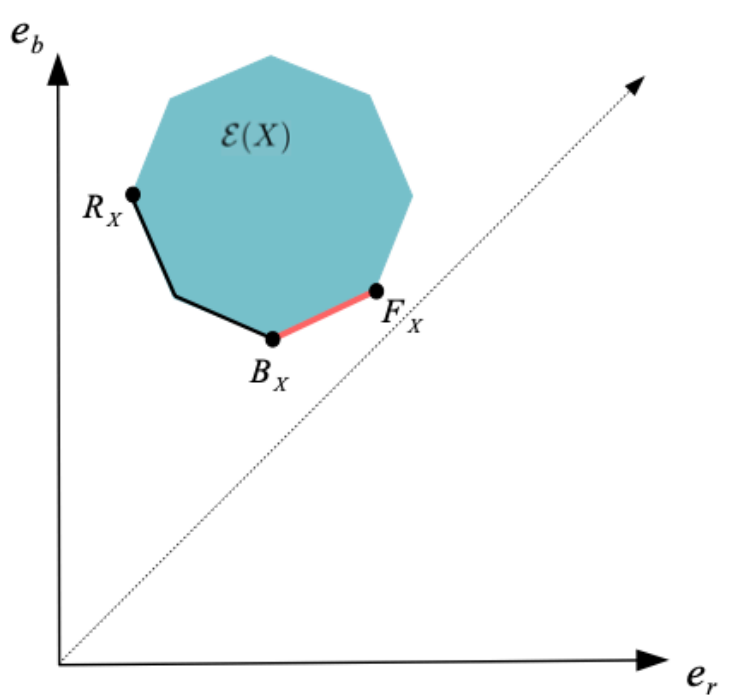
Lem. (characterization of FA frontier)

FA frontier is lower boundary of the feasible set $\mathcal{E}(X)$ between

- R_X and B_X if X is group-balanced (usual Pareto frontier)
- R_X and F_X if X is r-skewed (usual Pareto frontier + more)

Characterization of FA frontier





Thm. (fairness-accuracy conflict)

It can be optimal to decrease accuracy due to fairness concerns only when X is group-skewed.

- designer (policymaker) should NOT sacrifice accuracy due to fairness concerns if the input is group-balanced.

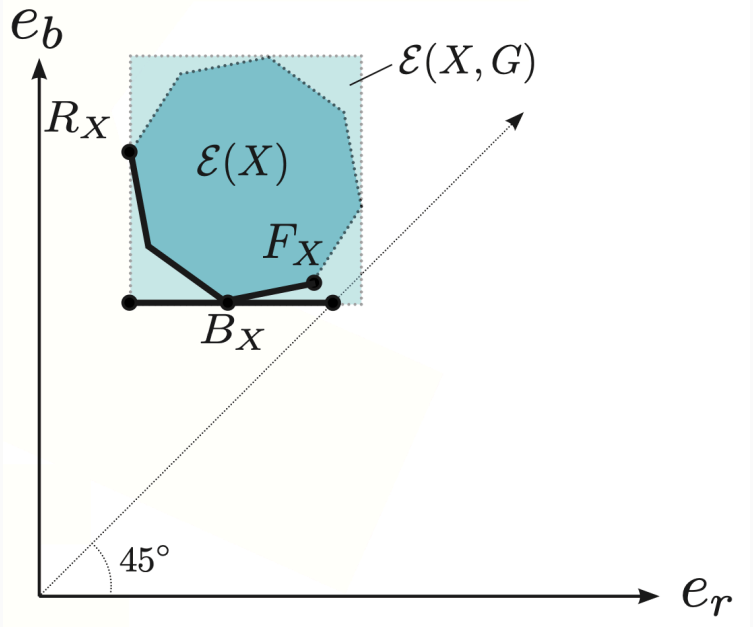
Which of group balance and group skew is more common?

- difficult to anticipate without an empirical analysis
- See Appendix for examples in which X is group-skewed/balanced examples

What happens when G is added as an input?

Result [informal] (adding group variables)

access to G reduces the error for the "worse-off group".



Caveat

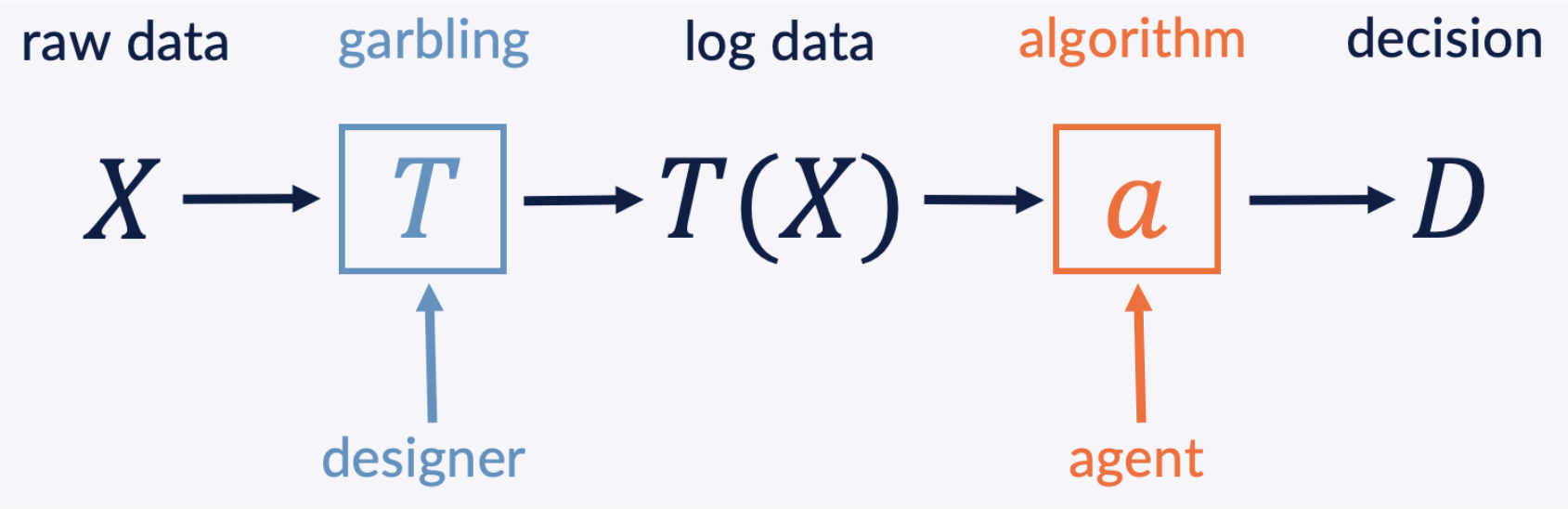
- the result is not true for the other group
- not about adding G or not for some fixed algorithm

Input Design: when the designer only controls the inputs

Input Design

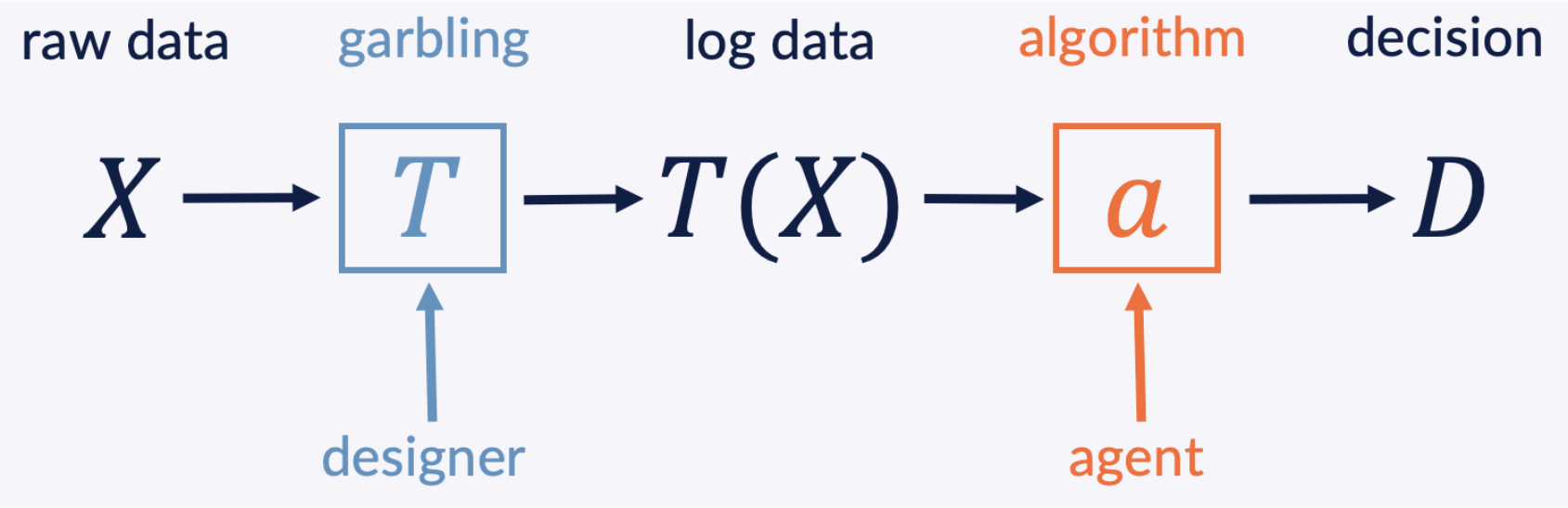
- so far, we assume that the one who designs the algorithms also has control over the inputs.
- from now on, we consider a different scenario:
  - the designer (e.g., government) only controls the inputs;
  - the agent (e.g., firm, university) designs the algorithm for his own sake using the inputs provided by the designer.

Input Design



- T: "data logging policy" to which the designer commits
  - formally, $T\colon \mathcal{X} \to \Delta(\mathcal{T})$ is a garbling for some set $\mathcal{T}$
- agent designs an algorithm to maximize his own payoff given the logging policy T
- the designer influences the final decision by choosing T, taking the agent's response into account
- Assume the utilitarian agent

Input Design



Examples of garblings

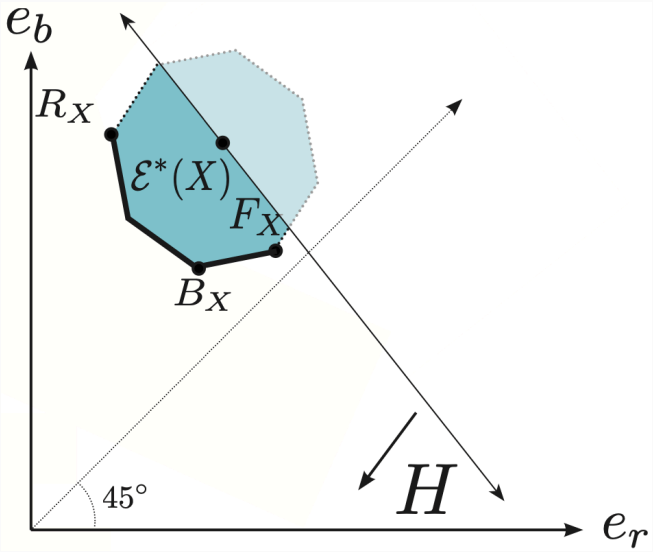
- test scores partitioned into coarse grades
- drop an input (e.g., “Ban the Box” – no use of criminal record for job application)

How powerful is input design?

- we can show that (under weak conditions) the designer can implement the favorite outcome by designing the inputs details

Prop. [informal] (power of input design)

Under weak conditions, any point that is feasible given X can be implemented using some garbling of X



Add/Ban Covariates?

- regulatory question: should certain inputs be banned?
  - some group identities are already banned (e.g., race for healthcare)
  - other covariates are also banned due to fairness concerns (e.g., test scores for college entrance)

- we can study this using the proposed framework

Question

how does FA frontier changes when excluding covariates?

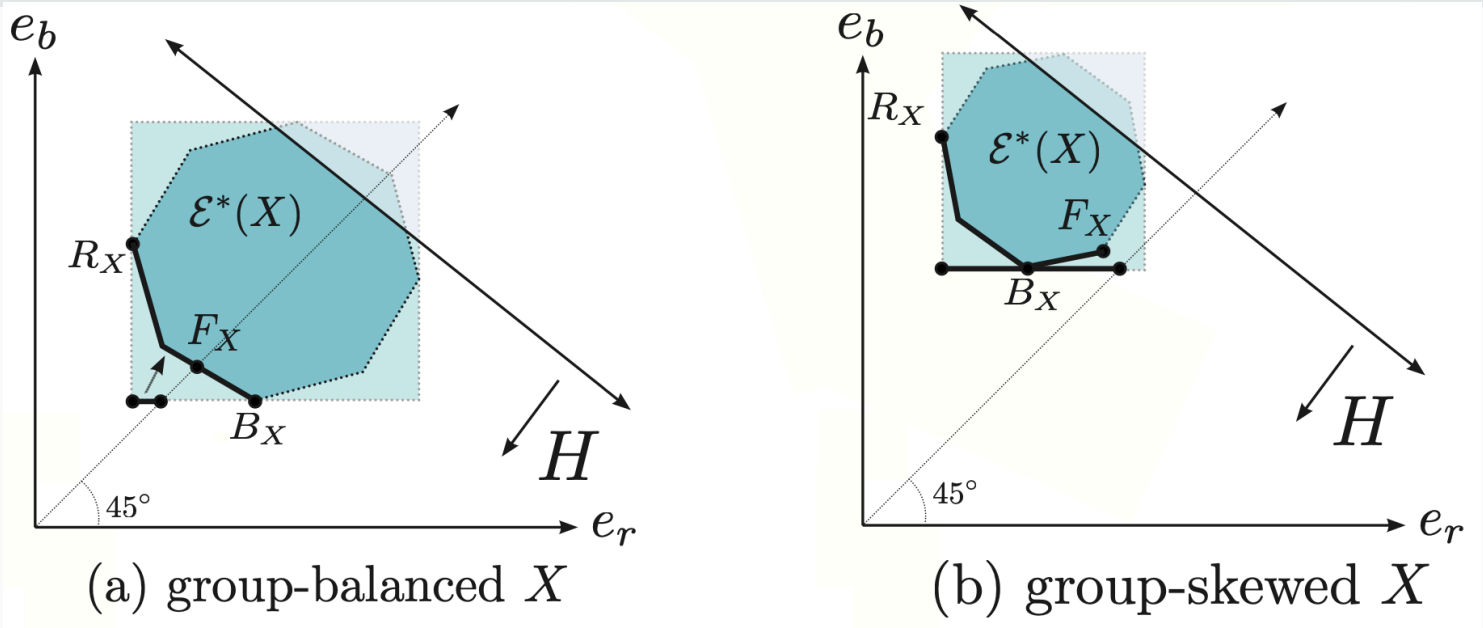
- We will see the following two scenarios:
  1. Excluding group identity: X vs. (X, G)
  2. Excluding a covariate X' when G is known: (X, G) vs. (X, G, X').

1. Excluding Group Identity

- compare X to (X, G)

Result [informal] (excluding group identity)

(under weak conditions) excluding G "uniformly worsens the frontier" iff X is group-balanced.



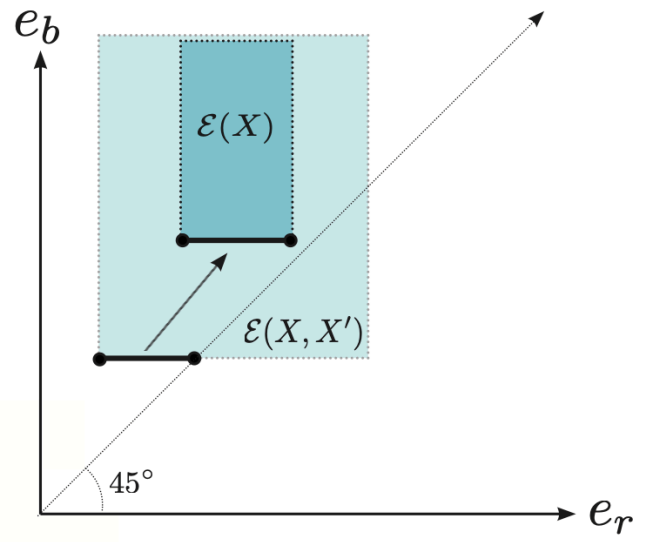
- NB: in this framework, it is almost always better to include G even when X is group-skewed.

2. Excluding a covariate when G is known

- compare (X, G) to (X, G, X').

Result [informal]

(under weak conditions) if (X, G) is group-skewed, then the designer is never better-off by excluding X' given (X, G).



(a) X' reduces group b's error

- designer can be better-off by excluding covariates from raw data only when G is already banned.
- NB: it is almost always the case that (X, G) is group-skewed.

Nuances/qualifications

- the results depend critically on the assumption that the designer has access to a fully flexible garbling of the inputs X (i.e., no restriction on the logging policy)
- do not imply a ranking between sending X' (un-garbled) versus excluding it

Summary

- framework for evaluating the accuracy/fairness tradeoffs of algorithms
- characterized the fairness-accuracy frontier over different designer preferences for how to trade off these criteria
- explained how certain statistical properties of the algorithm's inputs (group-balancedness) impact the shape of this frontier
- in some cases (e.g., when the inputs are group-balanced), there are conclusions/policy recommendations that hold for all designer preferences in a broad class

Appendix

Which of group balance and group skew is more common?

- difficult to anticipate without an empirical analysis

Example: X might be group-balanced

- this may happen when X has a group-dependent meaning
- e.g., X: frequency of moves
  - moving frequently signals high creditworthiness for high-income group (r); low creditworthiness for low-income group (b)
- suppose the algorithm observes a high frequency of moving;
- the decision made by best algorithms for group r (lend), does NOT coincide with the one for group b (not lend);
- decreasing error for group r implies increasing error for group b.

Which of group balance and group skew is more common?

- difficult to anticipate without an empirical analysis

Example: X might be group-skewed

- suppose X is asymmetrically informative
- e.g., X: frequency of hospital visits
  - low-income patients cannot visit hospitals even when they need treatments;
  - infrequent visit does not necessarily mean that patients don't need treatment for low-income patients.
- the decision made by best algorithms coincides for both groups given x (give treatment iff frequency is high)
- this implies a lower error for high-income patients

back

Feasible and Pareto Sets

- we assume that the agent is utilitarian
- let f_T denotes the utilitarian-optimal algorithm given T

Def. (feasible set and FA frontier under input design)

the feasible set under input design given X is

$$\mathcal{E}^*(X) := \{e(f_T) : T \text{ is a garbling of } X\}.$$

the fairness-accuracy frontier under input design given X is

$$\mathcal{F}^*(X) := \{e \in \mathcal{E}^*(X) : \nexists e' \in \mathcal{E}^*(X),\ e' >_{\text{FA}} e\}.$$

How powerful is input design?

- we can show that (under weak conditions) the designer can implement the favorite outcome by designing the inputs

- let e_0 be the agent's best payoff given no info
- let $H := \{(e_r, e_b) : p_r e_r + p_b e_b \leq e_0\}$

Prop. (power of input design) back

$\mathcal{E}^*(X) = \mathcal{E}(X) \cap H$,
i.e., any point that is feasible given X and in the half space H can be implemented using some garbling of X