Realizing the Promises of Algorithmic Recourse through Reliability, Accessibility, and Fairness Principles

Kate S. Boxer
New York University
New York, NY, USA
kb145@nyu.edu

Daniel B. Neill
New York University
New York, NY, USA
daniel.neill@nyu.edu

Abstract

Algorithmic recourse aims to provide individuals with actionable recommendations to reverse unfavorable outcomes from algorithmic decision-makers. For these systems to foster agency and trust, they must adhere to three principles: (1) recommendations, when acted upon, reliably lead to favorable outcomes, (2) realistically implementable recommendations are accessible at a high rate, and (3) fairness considerations must be upheld. We propose a novel training framework for algorithmic decision-makers that jointly optimizes accessibility to recommendations, predictive fairness, and fair algorithmic recourse, including equalized access to recommendations and equalized cost of recommendations across sensitive subpopulations, by using a burden-based multi-objective loss function. Evaluations across three data settings demonstrate significant improvements in availability of recommendations, reduced recommendation costs, and improved individual and group fairness properties compared to benchmarks. By imposing various constraints for generating recommendations, our approach ensures that recommendations reliably lead to favorable outcomes. This framework sets a new standard for algorithmic recourse by ensuring that systems that provide recourse uphold reliability, accessibility, and fairness standards, which is essential for materially increasing agency for those subject to algorithmic decisions and growing trust in algorithmic decision-making systems more broadly.

CCS Concepts

• **Social and professional topics** → **Socio-technical systems**; • **Mathematics of computing** → Mathematical optimization.

Keywords

Algorithmic recourse, Fairness and explainability in machine learning, Algorithmic decision-making, Socio-technical system analysis

ACM Reference Format:
Kate S. Boxer and Daniel B. Neill. 2025. Realizing the Promises of Algorithmic Recourse through Reliability, Accessibility, and Fairness Principles. In *Equity and Access in Algorithms, Mechanisms, and Optimization (EAAMO '25), November 05–07, 2025, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 23 pages. https://doi.org/10.1145/3757887.3763008

EAAMO '25, Pittsburgh, PA, USA

ACM ISBN 979-8-4007-2140-3/25/11
https://doi.org/10.1145/3757887.3763008

1 Introduction

The field of *algorithmic recourse* within explainable machine learning aims to equip individuals who have received unfavorable outcomes from an algorithmic decision-maker with the necessary information to achieve favorable outcomes in future interactions with the same algorithmic decision-maker [54]. Systems that provide recourse often consist of two components: an algorithmic decision-maker (typically a fixed binary classifier) that decides which individuals are given (un)favorable outcomes, and a recommendation generator that provides *only* those individuals who are given the unfavorable outcome with a recommendation (in the form of an action set) to perform and subsequently receive the favorable outcome [20]. To illustrate this concretely, consider an individual who applies for a mortgage with the following information: (Male, 55 years old, 57.5% of his cumulative credit line is still available, 586 credit score), and an algorithm determines that he is too high-risk to be granted the mortgage. Along with the rejection, he is provided with a recommendation to perform the following action: reduce his credit utilization so that 79% of his cumulative credit line is available. If that action is performed, when he reapplies for a mortgage, it will be approved.

Algorithmic recourse has been touted to support various principles that are necessary for individuals' well-being, including enabling individuals to assert agency through planning (i.e., *temporally extended agency*) and building societal *trust* in algorithmic decision-makers at large [51]. However, for these promises to fully materialize in practice, we argue that systems providing recourse must adhere to the following three principles:

- Principle R They must produce **reliable** recommendations, meaning that when individuals perform the recommendations provided by the system, and return to the same algorithmic decision-maker, they should consistently receive favorable outcomes.
- Principle A The systems must provide **access** to realistically implementable recommendations at a high rate for those given unfavorable outcomes.
- Principle F Systems that provide recourse must be **fair** with regards to consistent treatment regardless of race, gender, and other sensitive attributes.

Therefore, for systems that provide recourse to genuinely empower individuals to achieve favorable outcomes, fulfilling any single criterion – whether fair algorithmic decision-making, or reliability, accessibility, or fairness of recommendations – is insufficient because this is a multifaceted task and therefore requires a multi-objective approach.

Much of algorithmic recourse research has focused primarily on recommendation formulations and accompanying recommendation

generator algorithms [3, 21, 27, 33, 39, 50, 54]. While we motivate our choice of recommendation formulation, specifically regarding Principle R, our research goal is to **train algorithmic decision-makers, in the form of gradient-based classification models, that facilitate algorithmic recourse aligned with the above principles of reliability, accessibility, and fairness**. This is because, importantly, regardless of which recommendation definition and accompanying generator one uses, algorithmic decision-makers trained with recourse-agnostic objectives will often provide subpar recourse when deployed in real-world settings. We demonstrate this empirically by evaluating algorithmic decision-makers trained with recourse-agnostic objectives, as shown in Section 4. Intuitively, if a model heavily relies on features that individuals cannot modify (such as zip code at birth or race), this model will most likely provide very little recourse to most individuals regardless of the recommendation generator it is paired with.

We present a *burden*-based approach for reasoning about systems that provide recourse that considers *all* individuals in a target population, not only those given the unfavorable outcome and provided with recommendations. We formulate a multi-objective loss function that integrates various metrics, formed from burden measurements, to train algorithmic decision-makers, as outlined in Section 3. Our technique addresses various issues affecting systems that provide recourse, including non-reliable recommendations, lack of access to recommendations, imbalanced erroneous access to or denial of favorable outcomes, and imbalanced access to and cost of recommendations across sensitive subpopulations.

Our novel contributions include:

- Providing a guiding set of principles pertaining to reliability of recommendations, access to recommendations, and fairness for systems that provide recourse (Section 1).
- Illuminating a comprehensive set of fairness issues that affect systems that provide recourse (Section 2.2).
- Introducing a burden measurement that encapsulates predictive outcomes, access to recommendations, and cost of recommendations for the full target population, as well as an individual fairness measurement ("excess burden") that forms the foundation for a multi-objective loss function for training algorithmic decision-makers (Section 3).
- An empirical evaluation of our method for three data settings that highlights the statistically significant improvements for relevant metrics of our method compared to the benchmark methods (Section 4), including an ablation study (Section 4.1) that empirically supports our claim that to materialize algorithmic recourse's full potential, systems must simultaneously optimize multiple objectives.

By offering recommendations, systems that provide recourse often subtly imply that the responsibility for achieving favorable outcomes rests solely on individuals, deflecting the responsibility away from the system itself [47]. Our research offers a principled method to improve the algorithmic decision-maker itself (i.e., the underlying decision model), emphasizing that designers of these systems share responsibility, through the algorithmic decision-maker they provide, in enabling individuals to achieve favorable outcomes.

2 Motivation

2.1 Formalizations of Recommendations with Respect to Principle R and Principle A

As stated above, systems that provide recourse consist of an algorithmic decision-maker, $h_\theta(\cdot)$, which takes as input an individual i's feature values, $x_i = \{x_{ij}\}$. The algorithmic decision-maker either assigns individual i a favorable outcome, $h_\theta(x_i) = 1$, or unfavorable outcome, $h_\theta(x_i) = 0$. If individual i is given the unfavorable outcome, $h_\theta(x_i) = 0$, a recommendation generator algorithm, $A(\cdot)$, produces a recommendation for individual i. These recommendation algorithms are often, at their core, optimization solvers for minimization problems. For example, an early formulation of a recommendation as a constrained minimization problem adapted from [54] is displayed below:

$$x_i^{rec-Lp} = \underset{x' \in x}{\arg\min}\{\text{dist}(x_i, x') \mid h_\theta(x') = 1\} \quad \text{where dist}(\cdot,\cdot) \text{ is a Lp distance function.} \tag{1}$$

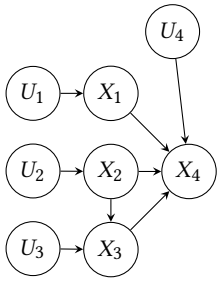
Therefore, a recommendation generator algorithm, $A(\cdot)$, that employs the recommendation definition in Equation 1 takes as input a point, x_i, that falls on the negative side of $h_\theta(\cdot)$'s decision boundary, and finds the nearest point to x_i in the feature space, as measured by a Lp distance function, that falls on the positive side of $h_\theta(\cdot)$'s decision boundary. x_i^{rec-Lp} can be decomposed as $x_i^{rec-Lp} = x_i + \delta_i^*$, where δ_i^* is treated as the *action set* an individual i needs to perform to get a favorable outcome from $h_\theta(\cdot)$ [20].

Ustun et al. [50] pointed out that when δ^* is unrestricted, an individual could be recommended to change their race, decrease their age, increase their age to 250, etc. Therefore, they proposed a taxonomy for restricting action set spaces through additional constraints, encoded as $\delta_i \in F(x_i)$, to ensure that recommendations are actionable and plausible. Additionally, they highlighted that $\text{dist}(\cdot,\cdot)$ as an Lp distance function might be an ill-suited measurement to minimize in the context of algorithmic recourse because the distance within a feature space might not be analogous to the effort it requires for an individual to shift their features. Therefore, they present various options for $\text{cost}(\cdot,\cdot)$ as a parametric component of the optimization problem. A formal definition, which is adapted from [50], is presented below:

$$x_i^{rec-cost} = x_i + \delta_i^* \quad \text{where } \delta_i^* = \underset{\delta_i \in F(x_i)}{\arg\min}\{\text{cost}(x_i, \delta_i) \mid h_\theta(x_i + \delta_i) = 1\} \tag{2}$$

The final formulation we explore was introduced by Karimi et al. [21], who critique the assumption that features can change independently of each other. They show that acting on a recommendation to change one feature may unintentionally alter additional features, and therefore cannot guarantee a favorable result in future interactions with the algorithmic decision-maker unless these causal relationships are taken into account.

For concreteness, we introduce a **simulated mortgage lending setting**, which we use throughout the paper. Consider a dataset for a mortgage lending scenario where the features for each individual are gender (X_1), age (X_2), proportion of cumulative credit line available (X_3), and credit score (X_4). These features relate to each other



$$X_1 := U_1$$
$$X_2 := U_2$$
$$X_3 := f_3(X_2) + U_3$$
$$X_4 := f_4(X_1, X_2, X_3) + U_4$$

Figure 1: The structural causal model (SCM) $\mathcal{M}$ for a simple simulated mortgage lending setting. Endogenous variables: X_1 is gender. X_2 is age. X_3 is proportion of cumulative credit line available. X_4 is credit score. Exogenous variable distributions: $U_1 \sim$ Bernoulli(0.50). $U_2 \sim \mathcal{N}(38, 22)$ truncated to $[18, 75] \cap \mathcal{Z}$. $U_3 \sim$ Uniform$(0.40, 1.09)$. $U_4 \sim$ Uniform$(200, 400)$. Structural functions: $f_3(X_2) = -0.005 * X_2$. $f_4(X_1, X_2, X_3) = X_1 + 50 * X_2 + 400 * X_3$. Outcome variable (not shown): whether an individual is trustworthy for a mortgage. Note, this is a simplified, simulated scenario rather than a complete and accurate model of mortgage lending. For more information about this data setting, including simulation algorithm, reference Appendix D.1.1. For more information on SCMs, reference Appendix B.

through the causal relationships shown in a structural causal model (SCM), $\mathcal{M} = (\mathcal{U}, \mathcal{X}, \mathcal{F})$, in Figure 1 [21, 36], which takes the form of an *additive noise model* [35]. Here $\mathcal{U}$ are exogenous variables, $\mathcal{X}$ are endogenous variables, and $\mathcal{F}$ are functional relationships between variables. If an individual who was denied a mortgage was given a recommendation that involved solely modifying variable X_2 to x' (equivalently, performing $do(X_2 = x')$ using the *do*-operator), this would result in variables X_3 and X_4 being modified as well because X_2 is a parent of both variables in $\mathcal{M}$, as shown in Figure 1. (For a primer on SCMs and the abduction-action-prediction framework using the *do*-operator, please reference Appendix B.) Therefore, the following definition, adapted from [21], accounts for the causal effects of recommendations when determining if a recommendation results in a favorable outcome:

$$x_i^{rec-SCM} = x_i \mid do(x_{ij} + \delta_{ij}^*)_{\forall j \in \delta_i^*}$$
$$\text{where } \delta_i^* = \underset{\delta_i \in F(x_i)}{\arg\min} \{\text{cost}(x_i, \delta_i) \mid h_\theta(x_i \mid do(x_{ij} + \delta_{ij})_{\forall j \in \delta_i}) = 1\} \tag{3}$$

For Equation 3, we assume δ_i only contains the recommended actions, indexed by j, for individual i, and the *do*-operator uses the specific SCM, $\mathcal{M}$, for the given data setting. A recommendation generator algorithm that assumes the recommendation formulation in Equation 3 could be represented as $\delta_i^* = A(x_i, h_\theta(\cdot), \text{cost}(\cdot, \cdot), F(\cdot), \mathcal{M})$ where $x_i^{rec-SCM} = x_i \mid do(x_{ij} + \delta_{ij}^*)_{\forall j \in \delta_i^*}$. Unlike Equation 2, $x_i^{rec-SCM} \neq x_i + \delta_i^*$, since modifying the variables in δ_i^* also impacts their descendants in the SCM.

We present the evolutionary nature of these recommendation formulations (Equations 1-3) to highlight the following phenomenon in algorithmic recourse research, which is supported by proofs in [21, 50]: **as the definition of a valid recommendation becomes more parametric and the corresponding optimization problem becomes more constrained, the recommendations become more reliable in terms of their realistic mapping to low-effort actions for individuals and their execution resulting in favorable outcomes in the future**. Therefore, to ensure that recommendations result in favorable outcomes, which fully addresses Principle R, and are executable for individuals, which partially addresses Principle A, we adopt Equation 3 as the recommendation definition we utilize in this research, and we use the accompanying recommendation generator algorithm, MINT [19]. For conciseness, for the remainder of the paper, we use $A_\theta(x_i) = A(x_i, h_\theta(\cdot), \text{cost}(\cdot, \cdot), F(\cdot), \mathcal{M})$, where $A_\theta(\cdot)$ assumes the recommendation definition in Equation 3 and takes the classifier's output $h_\theta(\cdot)$ as its input.

Additionally, we use the following cost function which is commonly used in algorithmic recourse research [19–21]:

$$\text{cost}(x_i, \delta_i) = \frac{1}{m} \sum_{\forall j \in \delta_i} w_j |\delta_{ij}|$$
$$\text{where } w_j = \frac{1}{\max_{\forall i}(x_{ij}) - \min_{\forall i}(x_{ij})} \tag{4}$$

In Equation 4, as described in [19–21], m is the number of actionable features in the dataset. *Actionable* refers to features that can be used in action sets, excluding unmodifiable features such as race and zip code at birth [20, 50]. This cost function normalizes the cost of an action independently for each feature based on its observed data distribution, allowing for diverse data types and settings. For the feasibility and plausibility constraints we used, $F(\cdot)$, for each data setting, see Appendix D.1.

As the recommendation definitions become more complex and the corresponding optimization problems become more constrained, two non-ideal scenarios can occur that are antithetical to Principle A: (1) individuals given unfavorable outcomes have no access to recommendations, and will continue to receive the unfavorable outcome regardless of what future actions they take; or (2) individuals are provided with very high-cost recommendations.

When scenario (1) occurs for individual i, we say that there is *no coverage* for individual i, and denote this as $A_\theta(x_i) = \varnothing$, as defined in [20, 25]. An inadequate recommendation generator could result in low coverage of recommendations. However, if the algorithmic decision-maker relies solely on non-actionable features, such as race, zip code at birth, or number of past late payments, then regardless of the recommendation generator, all individuals with

unfavorable outcomes will have no recommendations available to them. For scenario (2), as noted in [47], providing high-cost recommendations could effectively prevent many individuals from being able to act upon them, therefore, there are ethical motivations for designing systems that provide *low-cost* recommendations.

2.2 Fair Algorithmic Recourse in Relation to Principle F

As mentioned in Principle F, systems that provide recourse should conform to some notation of fairness. Gupta et al. [15] proposed a group parity measurement for fair algorithmic recourse:

$$\alpha^{\text{eq-cost}} = \left| \frac{1}{|\hat{S}_a^-|} \sum_{x_i \in \hat{S}_a^-} \text{cost}(x_i, \delta_i^*) - \frac{1}{|\hat{S}_{a'}^-|} \sum_{x_i \in \hat{S}_{a'}^-} \text{cost}(x_i, \delta_i^*) \right|, \quad (5)$$

where a binary sensitive attribute j is used to partition the set of individuals given unfavorable outcomes into two groups, $\hat{S}_a^- = \{x_i \in x : h_\theta(x_i) = 0, x_{ij} = a\}$ and $\hat{S}_{a'}^- = \{x_i \in x : h_\theta(x_i) = 0, x_{ij} = a'\}$. Fairness is measured by the absolute difference between subpopulations a and a' of the average cost of recommendations for individuals given unfavorable outcomes. Higher $\alpha^{\text{eq-cost}}$ represents a larger disparity in costs of recommendations.

Example 1: Let's say we have an algorithmic decision-maker, $h_\theta^{\text{eq-cost}}(\cdot)$, that is trained to minimize the balanced 0/1 loss and $\alpha^{\text{eq-cost}}$ (defined in Equation 5) for the simulated mortgage data setting introduced in Section 2.1 and shown in Figure 1. In Figure 2 we show a plot of the misclassification rates and recourse for $h_\theta^{\text{eq-cost}}(\cdot)$ for a balanced sample of the simulated mortgage data. Below are some observations for Figure 2:

- For those individuals with negative ground truth labels (who should be given the **unfavorable** outcome), men are erroneously and advantageously being granted the favorable outcome at a higher rate (0.18 FPR for men vs. 0 FPR for women). Women have a lower rate of coverage by 30%: they have *no access to the favorable outcome regardless of their future actions*.
- For those individuals with positive ground truth labels (who should be given the **favorable** outcome), women are erroneously (and to their disadvantage) being granted the unfavorable outcome at a higher rate (0.90 FNR for women vs. 0.70 FNR for men). Women have a lower rate of coverage by 24%: even though *they should have been given the favorable outcome*, they have *no access to the favorable outcome regardless of their future actions* .

These observations about Figure 2 reveal three fairness issues overlooked by $\alpha^{\text{eq-cost}}$: (1) imbalanced misclassification rates across sensitive subpopulations, creating disparities in erroneous access to or denial of favorable outcomes; (2) disparities in recommendation coverage rates across sensitive subpopulations; and (3) the lack of stratification by the true label (positive and negative class) when equalizing recommendation costs across sensitive subpopulations. This allows for expected costs of recommendations to be equalized in suboptimal ways, such as by equalizing the cost of recommendations for individuals who are false negatives in sensitive subpopulation A with individuals who are true negatives in sensitive subpopulation B.

An additional fairness metric was proposed in [53] that defines individual fair algorithmic recourse as an equal cost of recommendations for an individual and a counterfactual estimation of that individual if they were a member of the complement sensitive subpopulation (formally defined in Equation 6). This definition assumes that there is an underlying SCM, $\mathcal{M}$, and the counterfactual for x_i is $x_i^{CF} = x_i \mid do(1 - x_{ij})$, as proposed in [26], and $h_\theta(\cdot)$ is fair if $\alpha^{\text{ind-fair}} = 0$. Here $\delta_i^{*CF} = A_\theta(x_i^{CF})$. (To clarify, while the same SCM for a given setting is used to generate counterfactual estimates, x_i^{CF}, and recommendations, $x_i^{rec-SCM}$ (Equation 3), x_i^{CF} is *not* a recommendation.)

$$\alpha^{\text{ind-fair}} = \max_{x_i \in x : h_\theta(x_i) = 0} \left| \text{cost}(x_i, \delta_i^*) - \text{cost}(x_i^{CF}, \delta_i^{*CF}) \right| \quad (6)$$

As explained in [53], to ensure $\alpha^{\text{ind-fair}} = 0$, the sensitive attribute x_{ij} and all descendants of x_{ij} in $\mathcal{M}$ must be ignored when training $h_\theta(\cdot)$. In many contexts, this results in very few or no features available to train $h_\theta(\cdot)$, such as in the First-Year Law School Success setting in Figure 5b if one were considering gender and race as the axes for forming sensitive subpopulations. We record this metric in our evaluations but we do not directly use it as a benchmark method.

Example 2: In this example, we train an algorithmic decision-maker, $h_\theta^{0/1}(\cdot)$, to minimize the balanced 0/1 loss for the simulated mortgage dataset in Figure 1. In Figure 3, we see a sample of men and women and their counterfactuals when a *hard intervention* is performed using the SCM on their gender. (See Appendix B for information on calculating counterfactuals for SCMs.) There are men and women with the same outcome as their counterfactuals (◐ or ◐). For all men where this is not the case, the real-world man is at an advantage compared to his counterfactual, either by having lower recommendation cost than his counterfactual (◓) or erroneously being granted the favorable outcome when his counterfactual was not (◑). Conversely, for *all* women where their counterfactual has a different outcome, they are at a disadvantage compared to their counterfactual, for example:

- ◐ represents a true negative woman whose counterfactual, which estimates what would have happened to her if she was a man, would have been given the favorable outcome
- ◐ represents a true negative woman with no coverage whose counterfactual would have had access to recourse
- ◐ represents a false negative woman with no coverage whose counterfactual would have been given the favorable outcome
- ◐ represents a false negative woman with no coverage whose counterfactual would have access to recourse

This highlights a key point: **directionality matters when evaluating the difference between the costs of recourse for an individual and their counterfactual**. The absolute value in $\alpha^{\text{ind-fair}}$ (Equation 6) erases that directionality, which represents advantage (negative direction) and disadvantage (positive direction) for real-world individuals compared to their counterfactual, and could be correlated with protected class membership, as shown in Figure 3.

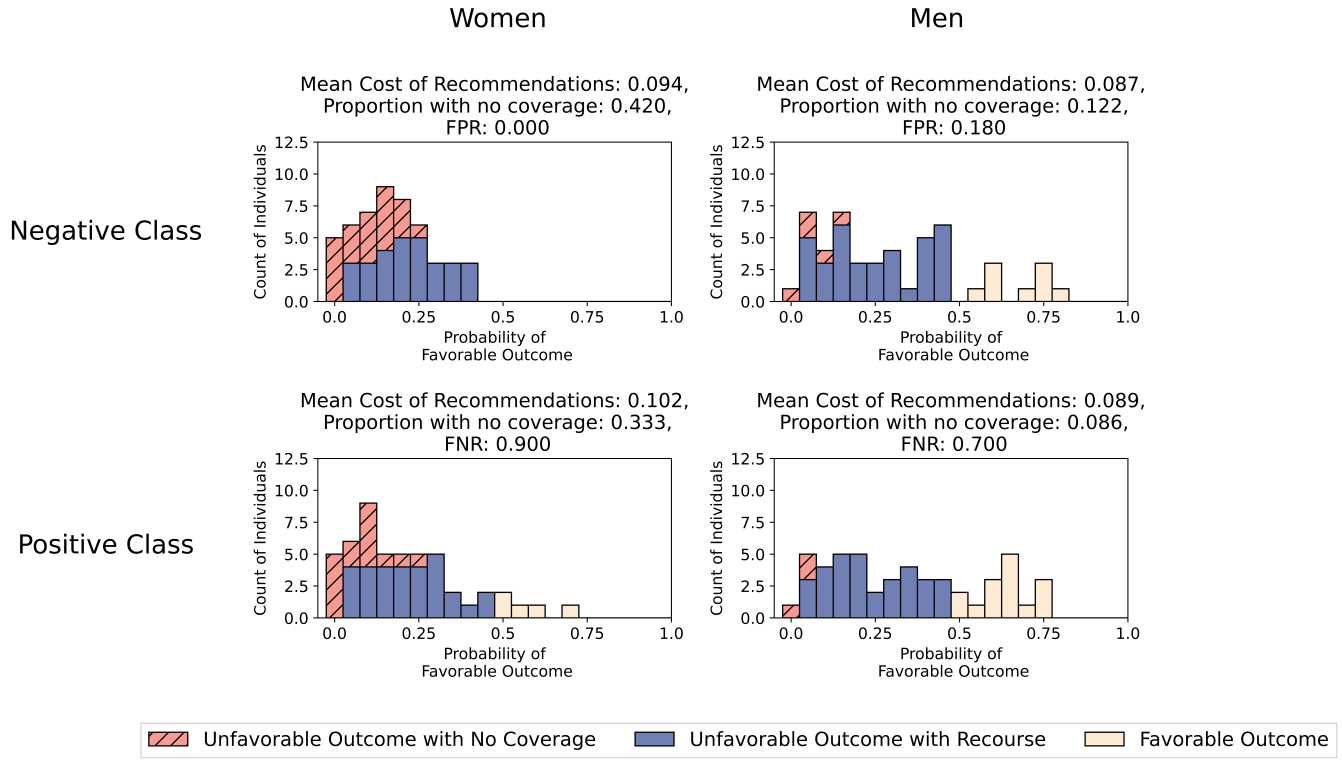


Figure 2: Plots of classification and recourse for 200 individuals from test data of the Simulated Mortgage Data (shown in Figure 1) in total (50 sampled from each subplot) for algorithmic decision-maker, $h_\theta^{\text{eq-cost}}(\cdot)$, where $\alpha^{\text{eq-cost}} = 0.011$ for test data. Proportion with no coverage is calculated for all individuals in the subsample given the unfavorable outcome and mean cost is calculated for all individuals in the subsample given the unfavorable outcome with recommendations (i.e., with coverage).

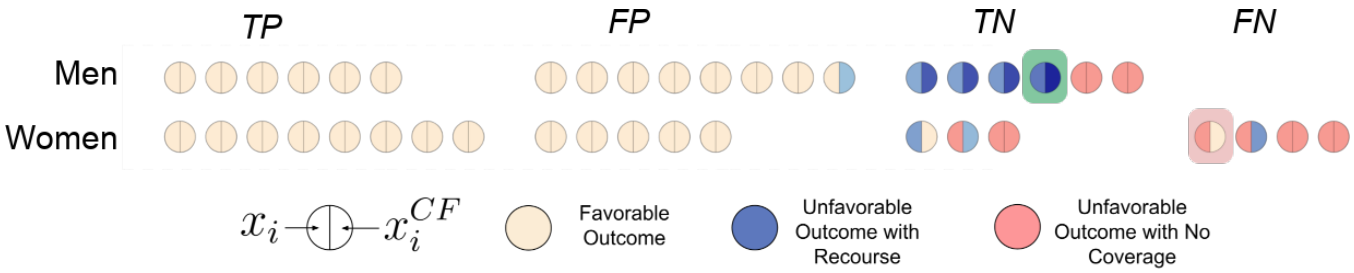


Figure 3: Sample of 40 individuals (upper row: 20 men, lower row: 20 women) for simulated mortgage setting in Figure 1 and their outcomes and recourse for $h_\theta^{0/1}(\cdot)$. The left-side shading of a circle represents individual x_i, the right-side shading represents their counterfactual $x_i^{CF} = x_i \mid \boldsymbol{do}(1 - x_{ij})$, and x_{ij} is the indicator variable representing membership in a sensitive subpopulation. Lighter blue represents lower cost recommendations. For this sample, $\alpha^{\text{ind-fair}} = 0.094$ and the dot in the green highlighted background represents that value (◓) - a man is at an *advantage* compared to his counterfactual because he has lower cost recommendations. The dot enclosed in the pink background represents the individual at the most *disadvantage* in this sample: a false negative woman with no coverage whose counterfactual based on an hard intervention on gender would have been given the favorable outcome (◒). $\alpha^{\text{ind-fair}}$ does not account for this individual because it only measures differences for those who received unfavorable outcomes with coverage, whose counterfactual also received an unfavorable outcome with coverage.

3 Our Approach

In Section 3.1, we introduce an individual-level measure of *burden*, which encapsulates predictive outcome (access to favorable outcomes) and access to and cost of recommendations. We also introduce an individual fairness measure ("excess burden"). We then introduce a multi-objective loss function that utilizes, at its core, the burden and excess burden measurements to train fair algorithmic decision-makers with high access to recommendations in Section 3.2.

3.1 Burden-Based Measurements of Access, Cost of Algorithmic Recourse and Individual Fairness

As shown in Section 2.2, systems that provide recourse, when examining misclassification rates and access to recommendations (coverage rates), can have disparate impacts across sensitive subpopulations. This is primarily because many frameworks for measuring fair algorithmic recourse only examine the subset of data that were given an unfavorable outcome and have a recommendation available. As shown in Figure 4, this might only cover a small subset of the data. We propose a function $b(x_i)$ that produces a *burden* measurement that encapsulates all the possible scenarios that could occur for an individual i who interacts with a system that provides recourse.

$$b(x_i) = \begin{cases} 0 & \text{if } h_\theta(x_i) = 1, \\ \frac{e^{\frac{c_i}{\lambda}} - 1}{e^{\frac{c_i}{\lambda}} + 1} & \text{if } h_\theta(x_i) = 0 \text{ and } A_\theta(x_i) \neq \varnothing, \\ 1 & \text{if } h_\theta(x_i) = 0 \text{ and } A_\theta(x_i) = \varnothing, \end{cases} \tag{7}$$

where $c_i = \text{cost}(x_i, \delta_i^*)$ and $\delta_i^* = A_\theta(x_i)$.

As shown in Equation 7, if individual i is given the favorable outcome, $h_\theta(x_i) = 1$, they have no burden to get the favorable outcome and $b(x_i) = 0$. If individual i is given the unfavorable outcome, and they have a recommendation provided by the recommendation generator, $A_\theta(x_i)$, the cost of their recommendation, $c_i = \text{cost}(x_i, \delta_i^*)$, is mapped between 0 and 1 using a hyperbolic tangent function scaled by λ (where $\lambda > 0$). If individual i is given the unfavorable outcome with **no coverage**, they are given the maximum burden of 1. Note, λ calibrates the burden measurement given the cost function and data setting. For the cost function defined in Equation 4 and our data settings, we use $\lambda = 0.10$.

Next, we define *excess burden* as the positive difference between the burdens for individual x_i and individual x_i's counterfactual, x_i^{CF}:

$$e(x_i) = \max(b(x_i) - b(x_i^{CF}), 0) \text{ where } x_i^{CF} = x_i \mid do(1 - x_{ij}). \tag{8}$$

This is similar to $\alpha^{\text{ind-fair}}$ in that it is an individual fairness measurement, but it only captures *disadvantage* for the real-world individual compared to their counterfactual, as discussed in Example 2 of Section 2.2. Additionally, $\alpha^{\text{ind-fair}}$ is only defined if an individual and their counterfactual get the unfavorable outcome with a recommendation (have coverage), while our burden measurement $b(\cdot)$ accounts for *all* scenarios for an individual x_i and their counterfactual x_i^{CF}, such as being granted the favorable outcome or having no coverage (Figure 4).

3.2 Training Fair Algorithmic Decision-Makers that Provide High Access to Low-Cost Recommendations

We use the individual-level burden and excess burden measurements in Equations 7 and 8 to construct a multi-objective loss function, incorporating loss terms for accuracy, accessibility and low cost recommendations (minimizing overall burden), individual fairness (minimizing excess burden), and group fairness (balancing excess burden). We then use this loss function to train algorithmic decision-makers that provide high access to realistically implementable recommendations (Principle A) and uphold various fairness criteria (Principle F). Fulfilling Principle R is a matter of adopting a recommendation definition that reliably results in a favorable outcome, and is discussed in Section 2.1.

We propose the following loss function to minimize while training algorithmic decision-makers:

$$\mathcal{L}_\theta^*(\cdot) = \mathcal{L}_\theta^{\text{acc}}(\cdot) + \mathcal{L}_\theta^{\text{burd}}(\cdot) + \mathcal{L}_\theta^{\text{exc-burd}} + \mathcal{L}_\theta^{\text{bal-exc-burd}}(\cdot) \tag{9}$$

We will walk through each component of Equation 9. For $\mathcal{L}_\theta^{\text{acc}}(\cdot)$, for our research, we use the balanced 0/1 loss:

$$\mathcal{L}_\theta^{\text{acc}}(\cdot) = (\beta_{FP} + \epsilon_{\text{corr}}) * FPR + \beta_{FN} * FNR \tag{10}$$

Therefore, by minimizing Equation 10, the misclassification rates (FPR and FNR) decrease. Other functions for predictive performance could be substituted in $\mathcal{L}_\theta^{\text{acc}}(\cdot)$, if one were concerned with calibration, etc. We will discuss the correction parameter, ϵ_{corr}, below. For $\mathcal{L}_\theta^{\text{burd}}(\cdot)$, we use the following:

$$\begin{aligned} \mathcal{L}_\theta^{\text{burd}}(\cdot) = {} & \beta_{FN}^{\text{burd}}(\mathbb{E}_{x_i \sim FN}[b(x_i)] * FNR) \\ & + \beta_{TN}^{\text{burd}}(\mathbb{E}_{x_j \sim TN}[b(x_j)] * TNR) \end{aligned} \tag{11}$$

Therefore, for Equation 11, for instances that are classified as negative (false negatives and true negatives), we minimize their burden, which increases the expected coverage for predicted negative instances, and also minimizes the average cost of recommendations for individuals given the unfavorable outcome with coverage. This aligns with Principle A which states that recommendations need to be available at a high rate and at a low enough cost that they can be realistically implemented. Note that, by multiplying by FNR and TNR respectively, the terms in Equation 11 represent the average burdens for *all* individuals with $y_i = 1$ and $y_i = 0$ respectively, not only the individuals given the unfavorable decision, since individuals who are classified positive have zero burden. We provide the functionality for practitioners to set different weights for false negatives and true negatives through the parameters β_{FN}^{burd} and β_{TN}^{burd}, since one may be more concerned with low-cost recourse for someone given the unfavorable outcome in error.

Next, we define the loss component for minimizing the individual-fairness measure of *excess burden*:

$$\begin{aligned} \mathcal{L}_\theta^{\text{exc-burd}}(\cdot) = {} & \beta_{FN}^{\text{exc-burd}}(\mathbb{E}_{x_i \sim FN}[e(x_i)] * FNR) \\ & + \beta_{TN}^{\text{exc-burd}}(\mathbb{E}_{x_j \sim TN}[e(x_j)] * TNR) \end{aligned} \tag{12}$$

Our $\mathcal{L}_\theta^{exc-burd}(\cdot)$ in Equation 12 takes a similar form in terms of the parameterization by false negatives and true negatives using $\beta_{FN}^{\text{exc-burd}}$ and $\beta_{TN}^{\text{exc-burd}}$, but is aimed at minimizing *excess burden*

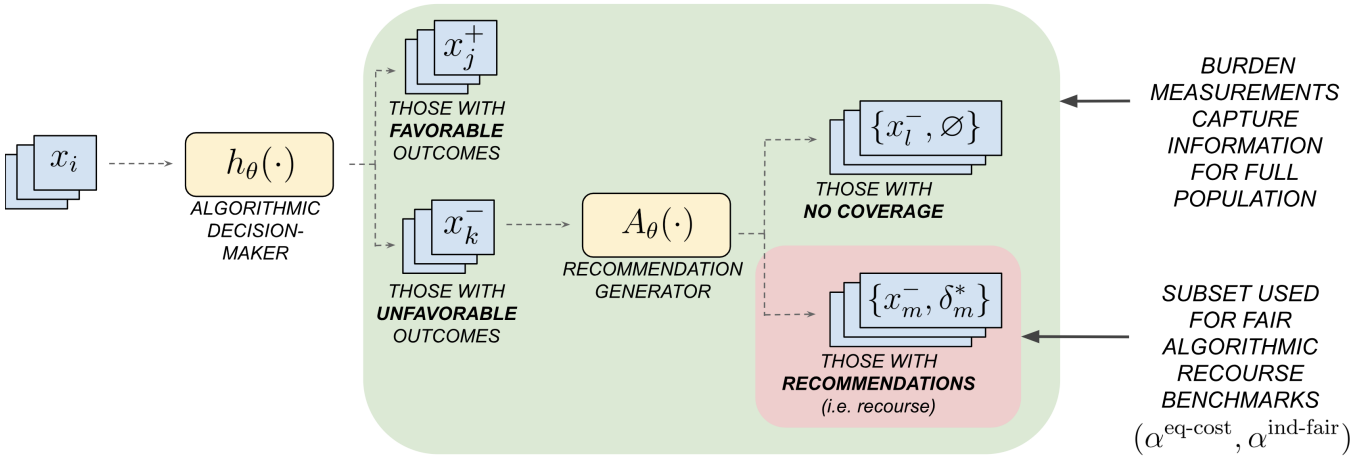


Figure 4: Diagram showing workflow of systems that provide algorithmic recourse. The pink box encapsulates the subset of data that is examined by the fairness measures $\alpha^{\text{eq-cost}}$ (Equation 5) and $\alpha^{\text{ind-fair}}$ (Equation 6). The green box shows all the pieces of information (predictive outcome, coverage, and cost of recommendation) encapsulated in our burden measurements and covers all the individuals in a population.

rather than burden. This term mitigates fairness issues related to imbalances in coverage and cost of recommendations that *disadvantage* an individual x_i compared to their counterfactual x_i^{CF}, which addresses Principle F. Note that we minimize the population average of the excess burden, in contrast to constraining $\alpha^{\text{ind-fair}} = 0$ as in [53], since the latter could greatly limit the features usable for training. Next, we define the group-fairness measurement of *balanced excess burden*:

$$\mathcal{L}_\theta^{\text{bal-exc-burd}}(\cdot) = \beta_-^{\text{bal-exc-burd}} \left| \mathbb{E}_{x_i \sim S_a^-}[e(x_i)] - \mathbb{E}_{x_j \sim S_{a'}^-}[e(x_j)] \right| + \beta_+^{\text{bal-exc-burd}} \left| \mathbb{E}_{x_k \sim S_a^+}[e(x_k)] - \mathbb{E}_{x_l \sim S_{a'}^+}[e(x_l)] \right|, \tag{13}$$

where $S_a^- = \{x_i : y_i = 0, x_{ij} = a\}$, $S_{a'}^- = \{x_i : y_i = 0, x_{ij} = a'\}$, $S_a^+ = \{x_i : y_i = 1, x_{ij} = a\}$ and $S_{a'}^+ = \{x_i : y_i = 1, x_{ij} = a'\}$. Therefore, $L_\theta^{\text{bal-exc-burd}}(\cdot)$ in Equation 13 serves to measure the imbalance in excess burden across sensitive subpopulations a and a' stratified by true label (positive and negative class), correcting the issue pertaining to $\alpha^{\text{eq-cost}}$ of comparing true negatives to false negatives observed in Example 1 of Section 2.2. Given that excess burden represents the disadvantage an individual i faces compared to their counterfactual, this term maintains that the excess burden should be distributed equally across sensitive subpopulations conditional on their true label. Importantly, since expectations are taken over all individuals (not just those who receive negative outcomes with coverage), minimizing $\mathcal{L}_\theta^{\text{bal-exc-burd}}$ also minimizes imbalances in misclassification rates, which addresses issues of predictive fairness, in relation to Principle F.

Lastly, ϵ_{corr} in Equation 10 is added to the false positive rate penalty to mitigate the risk of true negatives being flipped to false positives to minimize $\mathcal{L}_\theta^*(\cdot)$, given that true negatives often have higher cost recommendations or no coverage, while the burden and excess burden loss terms are 0 for a false positive. Therefore, we set $\epsilon_{\text{corr}} \approx \beta_{TN}^{\text{burd}} \mathbb{E}_{x_i \sim TN}[b(x_i)]$ where $\mathbb{E}_{x_i \sim TN}[b(x_i)]$ is estimated for a data setting using a baseline classifier (logistic regression). We note all values of ϵ_{corr} used for each data setting, as well as β_{FP} and β_{FN}, in Appendix D.3, Table 6.

Therefore, our overall objective is to learn the classifier parameters θ for $h_\theta^*(\cdot)$ that minimize $\mathcal{L}_\theta^*(\cdot)$:

$$\theta^* = \arg\min_\theta (\mathcal{L}_\theta^*(\cdot)) \text{ where } h_\theta^*(\cdot) \text{ and } A_\theta(\cdot) \text{ are inputs for } \mathcal{L}_\theta^*(\cdot) \tag{14}$$

$\mathcal{L}_\theta^*(\cdot)$ is calculated based on the predictions of $h_\theta^*(\cdot)$ and the algorithm that generates recommendations, $A_\theta(\cdot)$, which takes $h_\theta^*(\cdot)$ as an input. The recommendation generator, and its corresponding optimization problem (defined in Equation 3), forms the constraints based on various pieces of information including x_i and $h_\theta(\cdot)$. The specific algorithm we use, MINT [19], forms and solves the optimization problem independently for each recommendation generated, and consequentially, as $h_\theta(\cdot)$ is modified the optimization constraints change. Therefore, Equation 14 is a classic bi-level optimization problem, where the outer optimization problem solves Equation 14 and the inner problem finds the minimal cost recommendation using $A_\theta(\cdot)$ conditional on $h_\theta(\cdot)$. As a result, $\mathcal{L}_\theta^*(\cdot)$ is not easily differentiable [13]. To make our approach recommendation generator-agnostic, applicable for all differentiable models, and flexible for variations of our loss function, we use a method related to finite differences method called Simultaneous Perturbation Stochastic Approximation (SPSA) [45, 46]. This method is similar to stochastic finite differences approaches in that it produces $\frac{\partial \hat{\mathcal{L}}_\theta^*(\cdot)}{\partial \theta}$, which can be used in a gradient descent algorithm, but it provides certain advantages, such as fewer gradient estimates, through simultaneous perturbations of multiple θ parameters at once. This is especially important for computationally expensive algorithms such as most recommendation generator algorithms. For more information about SPSA and the parametric settings we used

(which are the default values recommended for practical effectiveness, theoretical soundness and convergence guarantees in [46]) see Appendix C.1. For a given model, we perform gradient descent, using $\frac{\partial \hat{\mathcal{L}}^*_\theta(\cdot)}{\partial \theta}$ from SPSA to update $h^*_\theta(\cdot)$ until convergence of $\mathcal{L}^*_\theta(\cdot)$, and then perform a random reset of the parameters θ. We repeat this process of training until convergence and random resets iteratively for 12 hours for a given model, and then take the parameters that produce the lowest training loss for $\mathcal{L}^*_\theta(\cdot)$ as our final model in terms of the θ^* parameters. We use θ^* to produce evaluation metrics for our test data. For the exact parameters of our gradient descent algorithm with random resets (convergence rule, batch size, initialization scheme for θ, pseudocode (Algorithm 1), and expected number of random resets), see Appendix C.1.

4 Evaluation

In Section 1, we outline a set of principles and illustrate in Section 2 various suboptimal scenarios that occur, even when utilizing existing fair algorithmic recourse methods. Our main argument is that, for recourse to achieve its positive potential, algorithmic decision-makers must be trained to satisfy multiple objectives. Therefore, we propose a multi-objective loss, $\mathcal{L}^*_\theta(\cdot)$, to minimize during training of algorithmic decision-makers in Section 3.

Therefore, we will show that the algorithmic decision-makers, $h^*_\theta(\cdot)$, trained with our loss function, $\mathcal{L}^*_\theta(\cdot)$, outperform other algorithmic decision-makers. We use the following algorithmic decision-makers as benchmarks:

- $h^{0/1}_\theta(\cdot)$ - Algorithmic decision-makers trained to minimize balanced 0/1 loss.
- $h^{\text{eq-cost}}_\theta(\cdot)$ - Algorithmic decision-makers trained to minimize balanced 0/1 loss and equalize the cost of recommendations across sensitive subpopulations, as defined by $\alpha^{\text{eq-cost}}$ in Equation 5 and used in Example 1 (Section 2.2).
- $h^{\text{bal-err}}_\theta(\cdot)$ - Algorithmic decision-makers trained to minimize balanced 0/1 loss and equalize the error rates (FPR and FNR) across sensitive subpopulations. We use this benchmark to address the issue of differential erroneous access to or denial of the favorable outcome across sensitive subpopulations, as motivated in Example 1.
- $h^{\text{bal+eq}}_\theta(\cdot)$ - Algorithmic decision-makers trained to minimize balanced 0/1 loss and equalize the error rates *and* recommendation costs across sensitive subpopulations.

We note that $h^{0/1}_\theta(\cdot)$ and $h^{\text{bal-err}}_\theta(\cdot)$ are trained with *recourse-agnostic* objectives, as we discuss further below. We train the benchmark algorithmic decision-makers using SPSA. For exact loss functions and more details about these benchmark methods, please reference Appendix D.2.

We train these algorithmic decision makers in three data settings: (1) the simulated mortgage lending setting introduced in Section 1 (Figure 1) where the sensitive feature is gender; (2) the German Credit Data [16] setting where the sensitive feature is gender, the other features are age, credit amount and duration of months for repayment, and the outcome variable for training is whether an individual is creditworthy (Appendix D.1, Figure 5a); and (3) the First-Year Law School Success [55] setting where the sensitive feature is race (Black or white), the other features are LSAT score, gender, and undergraduate GPA, and the outcome variable for training is whether the individual performed above average in their first-year of law school (Appendix D.1, Figure 5b). One can imagine algorithmic decision-makers being used in these settings for making mortgage lending, credit lending, and law school acceptance decisions, respectively.

Lastly, we test three differentiable models as algorithmic decision-makers: logistic regression (lr); a multi-layer perceptron classifier with one layer and two hidden units (MLP(1x2)); and a multi-layer perceptron classifier with one layer and four hidden units (MLP(1x4)).

Table 1 displays the results for the MLP(1x2) classifiers for all the data settings for the benchmarks and for our method (using default coefficient values $\beta = 1$ for all burden, excess burden, and balanced excess burden terms). This table contains established metrics related to algorithmic recourse and predictive accuracy such as expected cost of recommendations ($\mathbb{E}(\text{Cost})$), coverage, $\alpha^{\text{eq-cost}}$ (Equation 5), etc. Critically, we note that these metrics are distinct from the components of our loss function or burden measurements, which we directly use for optimization. For full results for all the model classes, which are comparable in performance to the MLP(1x2) classifiers, as well as results for models that place more emphasis on false negatives rather than true negatives, see Appendix D.4, Table 9. We review the results in Table 1 in relation to Principle A for *accessibility* and Principle F for *fairness*:

In regards to Principle A. Our method, $h^*_\theta(\cdot)$, statistically significantly increases access to recommendations for those with the unfavorable outcomes (Coverage) compared to the benchmark methods for the Simulated Mortgage Lending and First-Year Law School Success settings. It also increases coverage compared to the benchmarks for the German Credit Data setting, but this dataset achieves higher coverage at baseline and therefore the increase is not statistically significant. Furthermore, for all settings, our method statistically significantly decreases the expected cost of recommendations for those with unfavorable outcomes compared to the benchmarks.

In regards to Principle F. Our method, $h^*_\theta(\cdot)$, statistically significantly decreases the absolute difference for mean cost across sensitive subpopulations ($|\Delta_a|$ for $\mathbb{E}(\text{cost})$) compared to the benchmarks for all data settings. Importantly, our method, $h^*_\theta(\cdot)$, statistically significantly outperforms $h^{\text{eq-cost}}_\theta(\cdot)$ in minimizing $|\Delta_a|$ for $\mathbb{E}(\text{cost})$ for all data settings, even though $h^{\text{eq-cost}}_\theta(\cdot)$ is directly trained to minimize this value. As motivated by Example 1 in Section 2.2, different rates of coverage across sensitive subpopulations create a fairness issue. Our method statistically significantly decreases the absolute difference in rate of coverage compared to the benchmarks for all data settings ($|\Delta_a|$ for Coverage).

Additionally, as mentioned in Example 1, comparing costs across subpopulations agnostic of their observed y_i value (i.e., positive class or negative class) could result in the issue of equalizing the cost for false negative individuals in one subpopulation compared to true negative individuals in the other subpopulation. Our method lowers $|\Delta_a|$ for $\mathbb{E}(\text{cost})$ when stratified by positive and negative class ($y_i = 1$ and $y_i = 0$) compared to all the benchmarks for all data settings, with statistically significant reductions except in the Simulated Mortgage Lending setting for the positive class. While our method does not explicitly aim to achieve $\alpha^{\text{ind-fair}} = 0$ for the

Data Setting	Algorithmic Decision-Maker	Balanced 0/1 Loss	Coverage	$\lvert\Delta_a\rvert$ for Coverage	$\mathbb{E}$(Cost)	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) ($\alpha^{\text{eq-cost}}$)	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) $\forall\, y_i = 1$	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) $\forall\, y_i = 0$	$\alpha^{\text{ind-fair}}$	$\lvert\Delta_a\rvert$ for FPR	$\lvert\Delta_a\rvert$ for FNR
Simulated Mortgage Lending	$h^{0/1}$	0.514	0.653	0.203†	0.089†	0.026†	0.027†	0.029†	0.088‡	0.294	0.311
	$h^{\text{eq-cost}}$	0.511	0.771	0.117†	0.089	0.013†	0.018†	0.018†	0.051‡	0.268	0.271
	$h^{\text{bal-err}}$	0.505	0.763	0.031	0.093	0.018†	0.014†	0.019†	0.014‡	0.056	0.055
	$h^{\text{bal+eq}}$	0.504	0.812	0.035	0.088	0.011	0.012	0.017	0.012	0.053	0.053
	h^{*}	0.511	**0.999**	**0.001**	**0.058**	**0.007**	0.010	**0.011**	**0.004***	0.054	0.047
German Credit Data	$h^{0/1}$	0.408	0.975	0.014	0.083	0.018	0.017	0.025	0.135	0.411	0.383
	$h^{\text{eq-cost}}$	0.414	0.957	0.031	0.078	0.012	0.012	0.017	0.110‡	0.337	0.306
	$h^{\text{bal-err}}$	0.424	0.892	0.043	0.083	0.015	0.015	0.024	0.069	0.077	0.068
	$h^{\text{bal+eq}}$	0.431	0.891	0.037	0.078	0.014	0.016	0.023	0.054	0.089	0.078
	h^{*}	0.429	0.993	**0.005**	**0.055**	**0.007**	**0.007**	**0.013**	**0.034***	0.143	0.113
First-Year Law School Success	$h^{0/1}$	0.414	0.766	0.507	0.194	0.298†	0.239†	0.301†	0.600‡	0.656	0.707
	$h^{\text{eq-cost}}$	0.419	0.681	0.931	0.149	0.343†	0.120†	0.318†	———‡	0.664	0.741
	$h^{\text{bal-err}}$	0.454	0.661	0.097	0.215†	0.071†	0.112†	0.059†	0.133‡	0.070	0.099
	$h^{\text{bal+eq}}$	0.447	0.848†	0.062†	0.180†	0.051†	0.089†	0.042†	0.145‡	0.080	0.130
	h^{*}	0.450	**1.000**	**0.000**	**0.119**	**0.018**	**0.033***†	**0.017**	0.140‡	0.124	0.137

Table 1: Results are shown for MLP(1x2) classifiers, $h^{*}_{\theta}(\cdot)$, trained by minimizing $\mathcal{L}^{*}_{\theta}(\cdot)$, where $\beta^{\text{burd}}_{FN} = \beta^{\text{burd}}_{TN} = \beta^{\text{exc-burd}}_{FN} = \beta^{\text{exc-burd}}_{TN} = \beta^{\text{bal-exc-burd}}_{-} = \beta^{\text{bal-exc-burd}}_{+} = 1$. For readability, we omit the θ subscript and $(\cdot)$ notation as well as standard deviations of all metrics. The settings for β_{FP}, β_{FN}, and ϵ_{corr} differ per data setting and appear in Appendix D.3, Table 6. Each value is averaged over 50 runs, each with a unique seed for a 70-30 train-test split. Classifiers are fit on training data, and results in the table reflect test data metrics. Benchmarks ($h^{0/1}_{\theta}(\cdot)$, $h^{\text{eq-cost}}_{\theta}(\cdot)$, $h^{\text{bal-err}}_{\theta}(\cdot)$, and $h^{\text{bal+eq}}_{\theta}(\cdot)$) are described in Section 4. Balanced 0/1 is $\frac{1}{2}FPR + \frac{1}{2}FNR$. Coverage is the proportion of individuals with unfavorable outcomes who received a recommendation. $\mathbb{E}$(Cost) is average cost for those with unfavorable outcomes and coverage. $\lvert\Delta_a\rvert$ is the absolute difference in the relevant metric across sensitive subpopulations, stratified by positive class ($y_i = 1$) and negative class ($y_i = 0$), where applicable. $\alpha^{\text{ind-fair}}$ is defined in Equation 6. The bold values indicate statistically significant improvements ($p < 0.05$, one-tailed t-test) over *all* the benchmark methods. No statistically significant reductions in performance were observed in this subset of results. † represents missingness (e.g. no individuals with recommendations for some runs) . ‡ represents no values found for $\alpha^{\text{ind-fair}}$ for some runs. * represents statistical significance was evaluated against all benchmarks that were not missing but not the full set of benchmarks. – represents for all runs $\alpha^{\text{ind-fair}}$ could not be calculated (insufficient coverage for x_i or x^{CF}_i).

reasons noted in Section 2.2, our method statistically significantly lowers $\alpha^{\text{ind-fair}}$ for the German Credit Data and Simulated Mortgage Lending settings compared to the benchmark methods. Lastly, while our method does not balance error rates as well as the benchmarks that explicitly perform this task, $h^{\text{bal-err}}_{\theta}(\cdot)$ and $h^{\text{bal+eq}}_{\theta}(\cdot)$, our method *drastically* decreases the absolute difference in error rates across sensitive subpopulations for all data settings compared to the benchmark trained for predictive accuracy, $h^{0/1}_{\theta}(\cdot)$, and the benchmark trained to equalize the costs of recommendations across sensitive subpopulations, $h^{\text{eq-cost}}_{\theta}(\cdot)$.

$h^{0/1}_{\theta}(\cdot)$ and $h^{\text{bal-err}}_{\theta}(\cdot)$ are not trained with any information provided by the recommendation generator, including the cost or availability of recommendations. Therefore, they are models that are trained with *recourse-agnostic objectives*. We observe that these models provide less coverage and higher expected cost recommendations, demonstrating that these models are less ideal for providing recourse compared to the other models trained with objectives that incorporate recommendation information.

Lastly, $h^{\text{bal+eq}}_{\theta}(\cdot)$—which aims to equalize both error rates and expected costs of recommendations across subpopulations—does not outperform our method, indicating that fair prediction objectives combined with equalized recourse costs are insufficient to capture all the scenarios addressed by our multi-objective approach.

4.1 Ablation Evaluation for Multi-Objective Loss Function

We provide an ablation study which demonstrates that our loss function, $\mathcal{L}^{*}_{\theta}(\cdot)$, a multi-objective function with terms $\mathcal{L}^{\text{acc}}_{\theta}(\cdot)$, $\mathcal{L}^{\text{burd}}_{\theta}(\cdot)$, $\mathcal{L}^{\text{exc-burd}}_{\theta}(\cdot)$, and $\mathcal{L}^{\text{bal-exc-burd}}_{\theta}(\cdot)$, outperforms these terms individually. These results support the claim that a single criterion, whether it be optimizing for a singular metric of fairness, $\mathcal{L}^{\text{exc-burd}}_{\theta}(\cdot)$ or $\mathcal{L}^{\text{bal-exc-burd}}_{\theta}(\cdot)$, or accessibility, $\mathcal{L}^{\text{burd}}_{\theta}(\cdot)$, is insufficient, and our multi-objective approach provides substantial performance improvements. Thus, we train the following models:

- $h^{\text{burd}}_{\theta}(\cdot)$ - Algorithmic decision-makers trained to minimize balanced 0/1 loss with $\mathcal{L}^{\text{acc}}_{\theta}(\cdot)$ and minimize burden with $\mathcal{L}^{\text{burd}}_{\theta}(\cdot)$, where $\beta^{\text{burd}}_{FN} = \beta^{\text{burd}}_{TN} = 3$.
- $h^{\text{exc-burd}}_{\theta}(\cdot)$ - Algorithmic decision-makers trained to minimize balanced 0/1 loss with $\mathcal{L}^{\text{acc}}_{\theta}(\cdot)$ and minimize excess burden with $\mathcal{L}^{\text{exc-burd}}_{\theta}(\cdot)$, where $\beta^{\text{exc-burd}}_{FN} = \beta^{\text{exc-burd}}_{TN} = 3$.
- $h^{\text{bal-exc-burden}}_{\theta}(\cdot)$ - Algorithmic decision-makers trained to minimize balanced 0/1 loss with $\mathcal{L}^{\text{acc}}_{\theta}(\cdot)$ and balance excess burden with $\mathcal{L}^{\text{bal-exc-burd}}_{\theta}(\cdot)$, where $\beta^{\text{bal-exc-burd}}_{-} = \beta^{\text{bal-exc-burd}}_{+} = 3$.

In reviewing the results of the ablation study in Table 2, while the algorithmic decision-makers trained to minimize burden and excess burden, $h^{\text{burd}}_{\theta}(\cdot)$ and $h^{\text{exc-burd}}_{\theta}(\cdot)$, statistically significantly increase coverage for the First-Year Law School Success setting, and all the models show improvements in terms of coverage and minimizing

Data Setting	Algorithmic Decision-Maker	Balanced 0/1 Loss	Coverage	$\lvert\Delta_a\rvert$ for Coverage	$\mathbb{E}(\text{Cost})$	$\lvert\Delta_a\rvert$ for $\mathbb{E}(\text{Cost})$ ($\alpha^{\text{eq-cost}}$)	$\lvert\Delta_a\rvert$ for $\mathbb{E}(\text{Cost})$ $\forall\, y_i = 1$	$\lvert\Delta_a\rvert$ for $\mathbb{E}(\text{Cost})$ $\forall\, y_i = 0$	$\alpha^{\text{ind-fair}}$	$\lvert\Delta_a\rvert$ for FPR	$\lvert\Delta_a\rvert$ for FNR
First-Year Law School Success	$h^{0/1}$	0.414	0.766	0.507	0.194	$0.298^{\dagger}$	$0.239^{\dagger}$	$0.301^{\dagger}$	$0.600^{\ddagger}$	0.656	0.707
	$h^{\text{eq-cost}}$	0.419	0.681	0.931	0.149	$0.343^{\dagger}$	$0.120^{\dagger}$	$0.318^{\dagger}$	$---^{\ddagger}$	0.664	0.741
	$h^{\text{bal-err}}$	0.454	0.661	0.097	$0.215^{\dagger}$	$0.071^{\dagger}$	$0.112^{\dagger}$	$0.059^{\dagger}$	$0.133^{\ddagger}$	0.070	0.099
	$h^{\text{bal+eq}}$	0.447	$0.848^{\dagger}$	$0.062^{\dagger}$	$0.180^{\dagger}$	$0.051^{\dagger}$	$0.089^{\dagger}$	$0.042^{\dagger}$	$0.145^{\ddagger}$	0.080	0.130
	h^{burd}	0.417	**0.985**	0.042	0.166	$0.215^{\dagger}$	$0.218^{\dagger}$	$0.206^{\dagger}$	$0.369^{\ddagger}$	0.654	0.552
	$h^{\text{exc-burd}}$	0.423	**0.971**	0.112	0.151	$0.180^{\dagger}$	$0.166^{\dagger}$	$0.177^{\dagger}$	$0.368^{\ddagger}$	0.618	0.519
	$h^{\text{bal-exc-burd}}$	0.449	0.783	0.080	0.192	$0.065^{\dagger}$	$0.092^{\dagger}$	$0.057^{\dagger}$	$0.173^{\ddagger}$	0.059	0.121
	h^{*}	0.450	**1.000**	**0.000**	**0.119**	**0.018**	$\mathbf{0.033}^{*\dagger}$	**0.017**	$0.140^{\ddagger}$	0.124	0.137

Table 2: Results for ablation evaluation in Section 4.1 for the MLP(1x2) classifiers, $h^{*}_{\theta}(\cdot)$, and ablated classifiers, $h^{\text{burd}}_{\theta}(\cdot)$, $h^{\text{exc-burd}}_{\theta}(\cdot)$, and $h^{\text{bal-exc-burd}}_{\theta}(\cdot)$ for the First-Year Law School Success setting (Appendix D.1, Figure 5b). The benchmarks ($h^{0/1}_{\theta}(\cdot)$, $h^{\text{eq-cost}}_{\theta}(\cdot)$, $h^{\text{bal-err}}_{\theta}(\cdot)$, and $h^{\text{bal+eq}}_{\theta}(\cdot)$) are described in Section 4 with technical details in Appendix D.2. Ablated classifiers are described in Section 4.1. Columns, formatting in relation to statistical testing and symbols are identical in description to those provided in Table 1.

cost of recommendations, and equalizing coverage, cost, and access compared to the other benchmarks, none of them provide the level of statistically significant improvements that our method, $h^{*}_{\theta}(\cdot)$, trained with $\mathcal{L}^{*}_{\theta}(\cdot)$, provides.

5 Limitations

Both the definition we adopt for recommendations in Equation 3, and our excess burden measure in Equation 8, utilize structural causal models. As in other research using SCMs [21, 26, 30, 53], we adopt standard causal assumptions including *causal sufficiency*, *positivity*, etc. The SCM we use for our simulated mortgage lending setting fully satisfies the necessary assumptions for SCMs. The SCM we use for the German Credit Data setting has been used in prior research including [21]. The SCM we use for the First-Year Law School Success setting adopts a graph from prior research [53], and we derive the structural equations. Our work demonstrates that our method outperforms benchmarks that do not use SCMs in a variety of settings, with different assurances around these assumptions and proper model specificity. With that said, more research about how robust SCMs are to these assumptions and model specifications, similar to [4] and [22], would greatly enhance all fairness research that uses SCMs, including ours.

We assume binary sensitive subpopulation membership. Many real-world datasets, including the First-Year Law School Success setting, have multiple sensitive attributes with more than two categories. Extending our approach to multi-categorical or intersectional subpopulations would be a valuable direction for future work. This could be done through incorporating pattern detection methods to find multidimensional and intersectional disparities in burden measurements to dynamically define sensitive subpopulations during training.

The SPSA method, a stochastic generalization of the finite differences method, was used to train our algorithmic decision-makers with gradient descent. SPSA assumes a smooth loss function, which requires use of a sufficiently smooth cost function for recommendations. Substituting a different fitting method, such as a genetic algorithm, would allow our method to apply to discrete cost functions as well. Similarly, our proposed loss function for minimization, $\mathcal{L}^{*}_{\theta}(\cdot)$, is only applicable for differentiable models. An interesting extension of our research could be to use the components of our loss function, $\mathcal{L}^{\text{burd}}_{\theta}(\cdot)$, $\mathcal{L}^{\text{exc-burd}}_{\theta}(\cdot)$ and $\mathcal{L}^{\text{bal-exc-burd}}_{\theta}(\cdot)$, to develop a model-agnostic method for fitting algorithmic decision-makers, using iterative training with instance weighting and/or data augmentation.

We utilize one recommendation generator algorithm in our research. As argued in Section 2.1, we do not consider all recommendation definitions, including [54], [50], and [42], inline with Principle R's goal of reliable recommendations, and therefore testing other recommendation generator algorithms seems tangential to our overall research goal of training algorithmic decision-makers that satisfy our three principles. Additionally, while we use multiple benchmark methods, we use only one fairness definition [15] from the fair algorithmic recourse literature. While others exist [1, 42], they do not address issues of lack of coverage or encapsulate all scenarios that could occur for a full population and rather focus on the subset of data with unfavorable outcomes as shown in Figure 4. Therefore, we anticipate they would perform similarly to [15] on the relevant metrics of interest. Finally, we note that the fairness principles and metrics proposed in [3] are reliant on opportunity sets rather than intervention-based counterfactuals, making it challenging to compare them to our method without extensive modifications to their formulation of fairness.

For limitations that pertain to the framing of algorithmic recourse at large, please reference Appendix A. The topics discussed in Appendix A include issues of model drift in relation to reliable recommendations, social and epistemic norms including deflection of responsibility inherent to the framing of algorithmic recourse, limitations of fixed cost functions, issues of shifting of population distributions in unintended, and potentially harmful, ways, and privacy concerns.

6 Related Work

Algorithmic recourse is an emerging field within explainable machine learning. Previous research has developed methods for generating recommendations, including gradient optimization techniques [18, 28, 29, 54], integer programming [50], graph-based methods [3, 39], genetic algorithms [24, 42], autoencoder-based methods [5, 7, 33], and SAT solvers [19]. For surveys of these methods,

see [20, 52]. Additionally, [32] provides a tool for benchmarking recommendation generators, and [25] introduces a verification process for determining if a model has no coverage for a given individual.

Some research has proposed desiderata for recourse, including actionability of recommendations [50], robustness to model drift [10, 14, 23, 31, 49], robustness to small input changes [6, 9, 43], and diversity of recommendations [29, 48]. We specifically reference the fairness criteria of Gupta et al. [15] and von Kügelgen et al. [53] above; other research on fair recourse includes [1, 3, 9, 42]. While previous research notes the connection between predictive fairness and fair algorithmic recourse, to the best of our knowledge, our research is the first to frame fairness of algorithmic recourse as a multi-objective issue that must consider all scenarios (imbalanced error rates, rates of coverage, imbalanced cost of recommendations, etc.) for a full target population.

Research focused on training algorithmic decision-makers for recourse with similar goals to our research include Gupta et al. [15], which we use as a benchmark, and Ross et al. [41], who specifically focus on training models for high coverage. Our research focuses on issues of coverage, as well as fairness and reliability of recommendations.

Venkatasubramanian and Alfano [51] present a philosophical basis for algorithmic recourse and Karimi et al. [21] argue for using SCMs in generating recommendations. Both papers were influential in forming our research goals. Lastly, we reference additional research pertaining to the ethics of algorithmic recourse in Appendix A.

7 Conclusion

Systems that provide recourse have the potential to improve the lives of individuals who interact (sometimes in compulsory settings) with algorithmic decision-makers when pursuing important life goals, such as attempting to obtain credit and education. While technically any model paired with a recommendation generator could be considered a system that provides recourse, for these systems to realize the potential of algorithmic recourse, the *right* model needs to be used in this system. We provide an approach, motivated by a set of principles, for training algorithmic decision-makers, given a recommendation generator, that results in fair systems that provide access to reliable recommendations.

Acknowledgments

This work was partially supported by the National Science Foundation's Program on Fairness in Artificial Intelligence in Collaboration with Amazon under Grant No. IIS-2040898.

We thank Professor Edward McFowland III for his contributions in helping to formulate and scope this research.

References

[1] Andrew Bell, Joao Fonseca, Carlo Abrate, Francesco Bonchi, and Julia Stoyanovich. 2024. Fairness in Algorithmic Recourse Through the Lens of Substantive Equality of Opportunity. *arXiv preprint arXiv:2401.16088* (2024).

[2] Emily Black, Samuel Yeom, and Matt Fredrikson. 2020. Fliptest: fairness testing via optimal transport. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 111–121.

[3] Lucius EJ Bynum, Joshua R Loftus, and Julia Stoyanovich. 2024. A New Paradigm for Counterfactual Reasoning in Fairness and Recourse. *arXiv preprint arXiv:2401.13935* (2024).

[4] Carlos Cinelli, Daniel Kumor, Bryant Chen, Judea Pearl, and Elias Bareinboim. 2019. Sensitivity analysis of linear structural causal models. In *International conference on machine learning*. PMLR, 1252–1261.

[5] Amit Dhurandhar, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Paishun Ting, Karthikeyan Shanmugam, and Payel Das. 2018. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. *Advances in neural information processing systems* 31 (2018).

[6] Ricardo Dominguez-Olmedo, Amir H Karimi, and Bernhard Schölkopf. 2022. On the adversarial robustness of causal algorithmic recourse. In *International Conference on Machine Learning*. PMLR, 5324–5342.

[7] Michael Downs, Jonathan L Chu, Yaniv Yacoby, Finale Doshi-Velez, and Weiwei Pan. 2020. Cruds: Counterfactual recourse using disentangled subspaces. *ICML WHI* 2020 (2020), 1–23.

[8] Frederick Eberhardt and Richard Scheines. 2007. Interventions and causal inference. *Philosophy of science* 74, 5 (2007), 981–995.

[9] Ahmad-Reza Ehyaei, Amir-Hossein Karimi, Bernhard Schölkopf, and Setareh Maghsudi. 2023. Robustness implies fairness in causal algorithmic recourse. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*. 984–1001.

[10] João Fonseca, Andrew Bell, Carlo Abrate, Francesco Bonchi, and Julia Stoyanovich. 2023. Setting the right expectations: Algorithmic recourse over time. In *Proceedings of the 3rd ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization*. 1–11.

[11] Ruijiang Gao and Himabindu Lakkaraju. 2023. On the impact of algorithmic recourse on social segregation. In *International Conference on Machine Learning*. PMLR, 10727–10743.

[12] Sofie Goethals, Kenneth Sörensen, and David Martens. 2023. The privacy issue of counterfactual explanations: explanation linkage attacks. *ACM Transactions on Intelligent Systems and Technology* 14, 5 (2023), 1–24.

[13] Stephen Gould, Basura Fernando, Anoop Cherian, Peter Anderson, Rodrigo Santa Cruz, and Edison Guo. 2016. On differentiating parameterized argmin and argmax problems with application to bi-level optimization. *arXiv preprint arXiv:1607.05447* (2016).

[14] Hangzhi Guo, Feiran Jia, Jinghui Chen, Anna Squicciarini, and Amulya Yadav. 2022. Rocoursenet: Distributionally robust training of a prediction aware recourse model. *arXiv preprint arXiv:2206.00700* (2022).

[15] Vivek Gupta, Pegah Nokhiz, Chitradeep Dutta Roy, and Suresh Venkatasubramanian. 2019. Equalizing recourse across groups. *arXiv preprint arXiv:1909.03166* (2019).

[16] Hans Hofmann. 1994. Statlog (German Credit Data). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5NC77.

[17] Catherine Huang, Chelse Swoopes, Christina Xiao, Jiaqi Ma, and Himabindu Lakkaraju. 2023. Accurate, Explainable, and Private Models: Providing Recourse While Minimizing Training Data Leakage. *arXiv preprint arXiv:2308.04341* (2023).

[18] Shalmali Joshi, Oluwasanmi Koyejo, Warut Vijitbenjaronk, Been Kim, and Joydeep Ghosh. 2019. Towards realistic individual recourse and actionable explanations in black-box decision making systems. *arXiv preprint arXiv:1907.09615* (2019).

[19] Amir-Hossein Karimi, Gilles Barthe, Borja Balle, and Isabel Valera. 2020. Model-agnostic counterfactual explanations for consequential decisions. In *International conference on artificial intelligence and statistics*. PMLR, 895–905.

[20] Amir-Hossein Karimi, Gilles Barthe, Bernhard Schölkopf, and Isabel Valera. 2022. A survey of algorithmic recourse: contrastive explanations and consequential recommendations. *Comput. Surveys* 55, 5 (2022), 1–29.

[21] Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. 2021. Algorithmic recourse: from counterfactual explanations to interventions. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*. 353–362.

[22] Amir-Hossein Karimi, Julius Von Kügelgen, Bernhard Schölkopf, and Isabel Valera. 2020. Algorithmic recourse under imperfect causal knowledge: a probabilistic approach. *Advances in neural information processing systems* 33 (2020), 265–277.

[23] Kshitij Kayastha, Vasilis Gkatzelis, and Shahin Jabbari. 2024. Learning-Augmented Robust Algorithmic Recourse. arXiv:2410.01580 [cs.LG] https://arxiv.org/abs/2410.01580

[24] Gunnar König, Timo Freiesleben, and Moritz Grosse-Wentrup. 2023. Improvement-focused causal recourse (ICR). In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 11847–11855.

[25] Avni Kothari, Bogdan Kulynych, Tsui-Wei Weng, and Berk Ustun. 2023. Prediction without preclusion: Recourse verification with reachable sets. *arXiv preprint arXiv:2308.12820* (2023).

[26] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. *Advances in neural information processing systems* 30 (2017).

[27] Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Xavier Renard, and Marcin Detyniecki. 2017. Inverse classification for comparison-based interpretability in machine learning. *arXiv preprint arXiv:1712.08443* (2017).

[28] Ana Lucic, Harrie Oosterhuis, Hinda Haned, and Maarten de Rijke. 2022. FOCUS: Flexible optimizable counterfactual explanations for tree ensembles. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 5313–5322.

[29] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. 2020. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 607–617.

[30] Ece Çiğdem Mutlu, Niloofar Yousefi, and Ozlem Ozmen Garibay. 2022. Contrastive counterfactual fairness in algorithmic decision-making. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*. 499–507.

[31] Duy Nguyen, Ngoc Bui, and Viet Anh Nguyen. 2023. Distributionally robust recourse action. *arXiv preprint arXiv:2302.11211* (2023).

[32] Martin Pawelczyk, Sascha Bielawski, Johannes van den Heuvel, Tobias Richter, and Gjergji Kasneci. 2021. Carla: a python library to benchmark algorithmic recourse and counterfactual explanation algorithms. *arXiv preprint arXiv:2108.00783* (2021).

[33] Martin Pawelczyk, Klaus Broelemann, and Gjergji Kasneci. 2020. Learning model-agnostic counterfactual explanations for tabular data. In *Proceedings of the web conference 2020*. 3126–3132.

[34] Martin Pawelczyk, Himabindu Lakkaraju, and Seth Neel. 2023. On the privacy risks of algorithmic recourse. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 9680–9696.

[35] Judea Pearl. 2009. *Causality*. Cambridge university press.

[36] Judea Pearl. 2010. Causal inference. *Causality: objectives and assessment* (2010), 39–58.

[37] Judea Pearl. 2013. Structural counterfactuals: A brief introduction. *Cognitive science* 37, 6 (2013), 977–985.

[38] Sikha Pentyala, Shubham Sharma, Sanjay Kariyappa, Freddy Lecue, and Daniele Magazzeni. 2023. Privacy-Preserving Algorithmic Recourse. *arXiv preprint arXiv:2311.14137* (2023).

[39] Rafael Poyiadzi, Kacper Sokol, Raul Santos-Rodriguez, Tijl De Bie, and Peter Flach. 2020. FACE: feasible and actionable counterfactual explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 344–350.

[40] Kaivalya Rawal, Ece Kamar, and Himabindu Lakkaraju. 2020. Algorithmic recourse in the wild: Understanding the impact of data and model shifts. *arXiv preprint arXiv:2012.11788* (2020).

[41] Alexis Ross, Himabindu Lakkaraju, and Osbert Bastani. 2021. Learning models for actionable recourse. *Advances in Neural Information Processing Systems* 34 (2021), 18734–18746.

[42] Shubham Sharma, Jette Henderson, and Joydeep Ghosh. 2019. Certifai: Counterfactual explanations for robustness, transparency, interpretability, and fairness of artificial intelligence models. *arXiv preprint arXiv:1905.07857* (2019).

[43] Dylan Slack, Anna Hilgard, Himabindu Lakkaraju, and Sameer Singh. 2021. Counterfactual explanations can be manipulated. *Advances in neural information processing systems* 34 (2021), 62–75.

[44] Kacper Sokol and Peter Flach. 2019. Counterfactual explanations of machine learning predictions: opportunities and challenges for AI safety. In *2019 AAAI Workshop on Artificial Intelligence Safety, SafeAI 2019*. CEUR Workshop Proceedings.

[45] James C Spall. 1992. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control* 37, 3 (1992), 332–341.

[46] James C Spall. 1998. Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on aerospace and electronic systems* 34, 3 (1998), 817–823.

[47] Emily Sullivan and Atoosa Kasirzadeh. 2024. Explanation Hacking: The perils of algorithmic recourse. *arXiv preprint arXiv:2406.11843* (2024).

[48] Emily Sullivan and Philippe Verreault-Julien. 2022. From explanation to recommendation: Ethical standards for algorithmic recourse. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*. 712–722.

[49] Sohini Upadhyay, Shalmali Joshi, and Himabindu Lakkaraju. 2021. Towards robust and reliable algorithmic recourse. *Advances in Neural Information Processing Systems* 34 (2021), 16926–16937.

[50] Berk Ustun, Alexander Spangher, and Yang Liu. 2019. Actionable recourse in linear classification. In *Proceedings of the conference on fairness, accountability, and transparency*. 10–19.

[51] Suresh Venkatasubramanian and Mark Alfano. 2020. The philosophical basis of algorithmic recourse. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 284–293.

[52] Sahil Verma, Varich Boonsanong, Minh Hoang, Keegan Hines, John Dickerson, and Chirag Shah. 2024. Counterfactual explanations and algorithmic recourses for machine learning: A review. *Comput. Surveys* 56, 12 (2024), 1–42.

[53] Julius von Kügelgen, Amir-Hossein Karimi, Umang Bhatt, Isabel Valera, Adrian Weller, and Bernhard Schölkopf. 2022. On the fairness of causal algorithmic recourse. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 36. 9584–9594.

[54] Sandra Wachter, Brent Mittelstadt, and Chris Russell. 2017. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harv. JL & Tech.* 31 (2017), 841.

[55] Linda F Wightman. 1998. LSAC National Longitudinal Bar Passage Study. LSAC Research Report Series. (1998).

A Challenges in Algorithmic Recourse

Our research, and much of the algorithmic recourse literature, relies on the assumption that an individual will revisit the *identical* algorithmic decision-maker. While research, including [1, 10, 40], explores challenges related to stability across different algorithmic decision-makers (often referred to as *model drift*), we strongly advocate for policy and legislative measures to ensure model consistency or to mandate the endorsement of recommendations. This is essential because even the most promising research cannot fully eliminate this issue, and the reliability of recommendations is essential for any system providing recourse. Having these kinds of assurances changes the framing of algorithmic recourse, both in regards to Principle R and in relation to issues of model drift. For example, a recommendation provided by a system that provides recourse, if legislatively endorsed, does not need to necessarily result in a favorable decision from the original algorithmic decision-maker *or* future algorithmic decision-makers. Rather, the endorsement is sufficient to guarantee that if an individual follows the recommendation then they will receive a favorable outcome. This points to an emerging but promising area of algorithmic recourse research that deconstructs the assumption that the algorithmic decision-maker is the final arbitrator of favorable outcomes [1].

As noted in [47], there are various assumptions and potential pitfalls that are inherent to the framing of algorithmic recourse. They notably discuss that counterfactuals in their early conception in explainable AI were supposed to help individuals understand how a decision was reached, offer grounds for contesting the decision, and understand how to reverse an unfavorable decision [54]. Sullivan and Kasirzadeh [47] explain that the "recourse-first" norm that focuses on understanding how to reverse an unfavorable decision neglects an important aspect of understanding how the algorithmic decision-maker makes decisions in the first place, and this is a harmful oversight. They cite various reasons for this, and we will briefly review the ones that are most notable in relation to our research. They assert that by providing only actionable recommendations for individuals, individuals are not provided with essential information like, for example, the most heavily weighted feature in this model is race. Other research, including [2], would provide explanations like this. They argue for the importance of the norm of the epistemology of understanding, which they believe is overlooked when recommendations are the main focus. While it is challenging to endorse an epistemological norm agnostic of context and population input, it is clear that only focusing on actionable recommendations at an individual-level for systems that provide recourse runs the risk of placing all the burden of unfavorable outcomes on individuals rather than the underlying systems that are distributing unfavorable outcomes or socio-structural issues that are preventing favorable outcomes. Sullivan and Kasirzadeh [47] refer to this as deflecting responsibility. In regard to our research, on an aesthetic level, we refer to recourse as providing *recommendations* not *explanations*. This is to ensure that we are not presenting the illusion that these recommendations serve as explanations as well. Additionally, our research focuses on how to find optimal algorithmic decision-makers, while holding the recommendation generator mechanism fixed, which has the built-in assertion that the designers of a system that provides recourse are responsible for examining how their system is affecting a population in regards to fairness, and access and reliability of recommendations. Lastly, we are extremely careful to be selective about the contexts we use as examples in this paper. This is *not* to say that education and financial customer lending contexts do not have socio-structural issues, but more so to stress that some contexts, such as judicial and correctional settings, present such a high level of socio-structural issues that it is implausible to entertain that systems that provide algorithmic recourse might expand a sense of agency or trust to those subject to them.

Sullivan and Kasirzadeh [47] also discuss that using a singular cost function imposes a social norm as to what is considered more challenging to act upon. They mention that, for example, for a recommendation that asks an individual to attain more education, let us say to move from having a high school degree to an associate's degree, this might be more challenging to perform for some individuals than others. To the best of our knowledge, all algorithmic recourse research, including ours, use some kind of fixed mechanism for assigning cost of recommendations. This suggests that it might be useful to incorporate participatory mechanisms into systems that provide algorithmic recourse. One form of that could be to provide a set of recommendations, which could allow individuals to pick what they consider to be low-cost, similar to that presented in [29]. Another could be, through incorporating survey feedback, to learn user-calibrated cost functions. The point is, having one cost function for a full population presents not only limitations to algorithmic recourse but, given its probable miscalibration across different sensitive subpopulations, could advantage some subpopulations and disadvantage others.

Systems that provide recourse, by providing a recommendation, present the risk of shifting population distributions in unintended and potentially harmful ways. We note, however, that all algorithmic systems that individuals interact with and have impacts on individuals' lives shift population distributions. Therefore, it is not a question of *if* the population distribution is shifting because of the system but *how* the population distribution is shifting. We stress this to ground the issue within a larger phenomenon that should be a consideration for all algorithmic decision-makers in sociotechnical settings, not just systems that provide recourse. In the context of algorithmic recourse, the issue of exacerbation of social segregation has been discussed in [11, 53]. This specifically pertains to the idea that recommendations differ substantially enough across sensitive subpopulations that the result of individuals acting upon them would be subpopulations differing even more substantially over time. This points to larger questions which are at the crux of most fair machine learning research, such as what the expectations should be for algorithmic decision-makers when subpopulations, defined by protected class membership, have differing data distributions? Should their data distributions, post-interaction with a system (traditional algorithmic decision-makers, an algorithm with human-in-the-loop, or a system that provides recourse), be shifted closer to each other, remain the same distance, or be further apart? In the case of algorithmic recourse, there are recommendation generator algorithms, including those utilizing methods for robust recommendations as formulated by [9], which minimize this issue by making the recommendations robust to protected class membership – meaning similar recommendations, not just recommendation

costs, for similar individuals across sensitive subpopulations. This requires a much deeper, most likely context-specific, evaluation by domain experts to determine when these kind of recommendation generators should be enforced. For example, in some contexts, access to the favorable outcome might be so essential to the well-being of individuals' lives that the tradeoff of higher-cost recourse for all individuals versus low-cost differing recommendation sets across subpopulations presents a challenging decision. Regardless, these kinds of robust recommendation generator algorithms could be paired with our method because these recommendation generators do not address the underlying issue of whether the algorithmic decision-maker is the right model to achieve reliability, accessibility, and fairness for a system that provides recourse.

Lastly, there are various privacy issues concerning algorithmic recourse, including revealing the underlying model [44]. There are also concerns about being able to infer which individuals were in a training data set by the recommendation costs provided by the system that provides recourse [12, 34, 44]. Privacy-preserving solutions, including [17, 38], for algorithmic recourse represent a newer line of research that is critical for addressing these issues and for the safe deployment of systems that provide recourse to be adopted in real-world settings.

B Select Primer on Structural Causal Models

Structural causal models (SCMs) are a subclass of structural equation models that are used to model causal relations. A structural causal model (SCM) consists of two components: (1) a directed acyclic graph (DAG) with directed edges and nodes; and (2) a set of structural equations which dictate how nodes interact through edges causally.

Nodes consist of two types, exogenous variables ($\mathcal{U}$) and endogenous variables ($\mathcal{X}$). Exogenous variables have no parent nodes, whereas endogenous variables have parents and/or a causal mechanism within the graph. For example, in the SCM for the German Credit Data setting (Figure 5a), U_1, U_2, U_3, and U_4 are exogenous variables and X_1 (gender), X_2 (age), X_3 (credit amount), and X_4 (repayment duration in months) are endogenous variables. The set of structural equations, $\mathcal{F}$, dictate how the endogenous variables are derived. Some might take the form directly of an exogenous variable distribution, such as gender, X_1, and age, X_2, as shown for the German Credit Data setting in Figure 5a. Other structural equations might be a combination of an exogenous variable and other endogenous variables, like X_3 (credit amount) and X_4 (repayment duration). The causal mechanism of these variables' parents is dictated by structural functions. For the German Credit Data, these are $f_3(\cdot)$ and $f_4(\cdot)$. While these causal mechanisms are deterministic, the exogenous variables introduce randomness. For our simulated mortgage lending setting, we provide the parametric settings for the exogenous variables in Figure 1. For our real-world data settings, German Credit Data and First-Year Law School Success, we allow the exogenous distributions to be observed and in a non-parametric form. Therefore, SCMs are defined by all their components as $\mathcal{M} = (\mathcal{U}, \mathcal{X}, \mathcal{F})$.

To both calculate the recommendations, $x_i^{rec-SCM}$, for Equation 3 and the counterfactuals, x_i^{CF}, we utilize in Equation 8 for excess burden, we perform a hard intervention using the *do*-operator for observed data, and therefore, we use the abduction-action-prediction process [37]. We will discuss this in the form of the process for a singular instance, x_i, dictated by the German Credit Data in Figure 5a, where we perform an intervention on age to determine what would have happened to individual i if their age was x'. However, this process could take place for all instances of the dataset and for various different interventions (not just the intervention on age that we demonstrate below). The process takes the form of:

Abduction. Using the observed data for x_i, we calculate the exogenous variables.

$$\begin{aligned} u_{i1} &= x_{i1} \\ u_{i2} &= x_{i2} \\ u_{i3} &= x_{i3} - f_3(x_{i1}, x_{i2}) \\ u_{i4} &= x_{i4} - f_4(x_{i3}) \end{aligned}$$

Action. We update the structural equations to reflect the *hard intervention* of setting x_i's age to x'. This is expressed notationally as $do(X_2 = x')$ and some literature would refer to this new model as a "surgically modified" submodel, $\mathcal{M}_{x'}$, [37] where:

$$\begin{aligned} X_1 &:= U_1 \\ X_2 &:= x' \\ X_3 &:= f_3(X_1, X_2) + U_3 \\ X_4 &:= f_4(X_3) + U_4 \end{aligned}$$

Predict. Use the modified model from the *'action'* step, $\mathcal{M}_{x'}$, and the exogenous variables calculated in the *'abduction'* step to compute $x_i \mid do(X_2 = x')$:

$$\begin{aligned} x_{i1} &:= u_{i1} \\ x_{i2} &:= x' \\ x_{i3} &:= f_3(x_{i1}, x_{i2}) + u_{i3} \\ x_{i4} &:= f_4(x_{i3}) + u_{i4} \end{aligned}$$

Therefore, performing $do(X_2 = x')$ results in changes to x_{i2}, x_{i3}, and x_{i4}. Our research utilizes hard interventions, meaning that all intervened variables are overridden and all causal influences from parent nodes are not retained. This is the convention in the algorithmic recourse literature because discrete actions are imposed in the form of recommendations. With that said, other intervention mechanisms are discussed in [8].

Next, we will discuss the two functions we use SCMs for in our research, calculating x_i^{CF} and $x_i^{rec-SCM}$:

Calculating counterfactuals, x_i^{CF}: To calculate the counterfactual for an individual, x_i, we perform an intervention on the binary variable that defines a sensitive subpopulation for each data setting. This notationally takes the form of $x_i^{CF} = x_i \mid do(x_{ij} = 1 - x_{ij})$ or $x_i^{CF} = x_i \mid do(1 - x_{ij})$, for short. To ensure that all data types and ranges for the counterfactuals match the observed data types and ranges, we stochastically round all mismatched data types using a Bernoulli distribution with a fixed seed to ensure that identical

recommendations are rounded to the same values across our experiments for the same data settings. For example, for the German Credit Data, we rounded variable X_4 (duration of repayment in months) to integers for all counterfactuals. Additionally, we clipped all counterfactual values to ensure they were within the observed range for each variable in the original data distribution. We did this to ensure that we were using consistent data types and constraints to those we utilize for calculating the recommendations.

Calculating recommendations, $x_i^{rec-SCM}$: As noted in Section 2.1, the recommendation generator, $A_\theta(\cdot)$, identifies the recommendation, δ_i^*, such that $x_i \mid do(x_{ij} + \delta_{ij}^*)_{\forall j \in \delta_i^*}$ results in a favorable outcome. Therefore, the action set, δ_i^*, could consist of multiple interventions, unlike x_i^{CF}, which is solely an intervention on protected class membership. With that said, δ_i^* is constrained by a function, $F(\cdot)$, that ensures all action sets are actually implementable as noted in Section 2.1. For example, for all of our data settings, an action set could *not* consist of recommending that someone change their group membership for a protected class attribute like gender or race. For documentation of all the constraints encoded in $F(\cdot)$ for each data setting, reference Appendix D.1.

C Our Approach Appendices

C.1 Simultaneous Perturbation Stochastic Approximation (SPSA) and Gradient Descent Algorithm

We discuss stochastic finite difference methods conceptually, then Simultaneous Perturbation Stochastic Approximation (SPSA), and then how we incorporate SPSA into our gradient descent algorithm.

Finite difference methods are commonly used for gradient approximation for equations that are not easily differentiable. This is the case for training algorithmic decision-makers with loss functions that utilize the output of recommendation generator algorithms, such as $\mathcal{L}_\theta^*(\cdot)$ in Section 3.2, which depends on the output of $A_\theta(\cdot)$, where $A_\theta(\cdot)$ is a black-box system that is not easily differentiable. $\mathcal{L}_\theta^*(\cdot)$ becomes non-differentiable with respect to θ because the function for $A_\theta(\cdot)$ is unknown, and $A_\theta(\cdot)$ depends on θ for $h_\theta^*(\cdot)$. Therefore, it is possible to compute $\mathcal{L}_\theta^*(\cdot)$, but $\frac{\partial \mathcal{L}_\theta^*(\cdot)}{\partial \theta}$ is not easily calculable. Given that our optimization goal is to find the θ parameters for $h_\theta^*(\cdot)$ that minimize $\mathcal{L}_\theta^*(\cdot)$, the general premise of finite difference methods is to evaluate $\mathcal{L}_\theta^*(\cdot)$ with a slight fluctuation by a small value, c, also known as a perturbation, to a singular dimension of θ, denoted as $\theta_i^{\text{pert}} = \theta_i + c$. If one were to envision $\mathcal{L}_\theta^*(\cdot)$ as a function of that singular dimension, θ_i, and $\mathcal{L}_\theta^*(\cdot)$ is a smooth function in relation to θ_i, then the perturbation, θ_i^{pert}, either increases or decreases $\mathcal{L}_\theta^*(\cdot)$ compared to the original setting for θ_i. Therefore, if we denote the loss function for θ_i as $\mathcal{L}_\theta^{*\text{base}}(\cdot)$ and $\mathcal{L}_\theta^{*\text{pert}}(\cdot)$ for θ_i^{pert}, then the slope for $\mathcal{L}_\theta^*(\cdot)$ in relation to the domain θ_i to θ_i^{pert} is $\frac{\mathcal{L}_\theta^{*\text{pert}}(\cdot) - \mathcal{L}_\theta^{*\text{base}}(\cdot)}{c}$. Therefore, this slope calculation serves nicely as an approximation of $\frac{\partial \mathcal{L}_\theta^*(\cdot)}{\partial \theta_i}$, without the need to directly take the partial derivative of $\mathcal{L}_\theta^*(\cdot)$ with respect to θ_i. Therefore, to find all the parameters of θ for $h_\theta^*(\cdot)$, one could imagine using the approximation of the gradient $\frac{\partial \hat{\mathcal{L}}_\theta^*(\cdot)}{\partial \theta_i}$, in a gradient descent algorithm where for each update, a random dimension of θ is chosen, and a random perturbation size or/and perturbation direction is chosen for c, and for each update $\theta_i = \theta_i - \eta \frac{\partial \hat{\mathcal{L}}_\theta^*(\cdot)}{\partial \theta_i}$, where η is a learning rate and $\frac{\partial \hat{\mathcal{L}}_\theta^*(\cdot)}{\partial \theta_i}$ is calculated from the perturbation method described directly above. If $\mathcal{L}_\theta^*(\cdot)$ were convex, with enough gradient updates, this process would be guaranteed to find the θ for the global minimum of $\mathcal{L}_\theta^*(\cdot)$. This is not the case for $\mathcal{L}_\theta^*(\cdot)$, and we discuss this further below. Note that the specific process we describe in this paragraph for illustrative purposes relates to the process of stochastic forward difference perturbations for finite differences. There are various different schemes for finite differences for gradient approximation, not limited to the one described above.

The process described above of performing a perturbation for a single dimension of θ using a singular record requires that $\mathcal{L}_\theta^*(\cdot)$ be calculated twice: once for $\mathcal{L}_\theta^{*\text{base}}$ and once for $\mathcal{L}_\theta^{*\text{perturb}}$. As mentioned in Section 6, many of the recommendation generator algorithms are optimization solvers that use genetic algorithms, mixed integer programming or SAT solvers, therefore, they are computationally prohibitive to call for every update to a singular dimension of θ. Simultaneous perturbation stochastic approximation (SPSA) utilizes perturbations to approximate gradients, similarly to stochastic finite difference methods, but allows for simultaneous perturbations to occur at once, greatly reducing the number of times one needs to compute $\mathcal{L}_\theta^*(\cdot)$ and subsequently call $A_\theta(\cdot)$ [45]. More concretely, $\frac{\partial \hat{\mathcal{L}}_\theta^*(\cdot)}{\partial \theta}$ requires $2n$ calculations of $\mathcal{L}_\theta^*(\cdot)$ and $4n$ calls to $A_\theta(\cdot)$ if a singular perturbation of θ is performed at a time for a singular record. (Recall that, to calculate $\mathcal{L}_\theta^*(\cdot)$ we must compute $A_\theta(x_i)$ and $A_\theta(x_i^{CF})$.) When simultaneous perturbations are performed to estimate $\frac{\partial \hat{\mathcal{L}}_\theta^*(\cdot)}{\partial \theta}$, 2 calculations of $\mathcal{L}_\theta^*(\cdot)$ and 4 calls to $A_\theta(\cdot)$ are required . Obviously, this is scaled by the number of records in the training data.

To explain the SPSA algorithm, we provide the full pseudocode we use to train algorithmic decision-makers in Algorithm 1, which consists of a gradient descent algorithm using SPSA gradient approximation to update θ with random resets upon convergence.

The SPSA gradient approximation process happens iteratively, in Lines 19-22 of Algorithm 1. Importantly, in Line 19, Δ_k cannot have infinite inverse moments, meaning that sampling from a normal or uniform distribution is prohibited [46]. As is common practice, we sample Δ_k from a symmetric Rademacher distribution, which functions like a Bernoulli distribution set to $p = 0.50$ that produces either $[-1, +1]$. The necessity of this requirement follows from the inversion of Δ_k in Line 21.

To calculate c_k depending on the gradient update count, k, we use the following calculation from [46]:

$$c_k = \frac{c}{(k+1)^\lambda}$$

We use $\lambda = 0.101$, because, as noted by [46], this guarantees practical effectiveness and theoretical validity. Additionally, we use $c = 0.10$, as is common practice. To calculate η_k depending

Algorithm 1 Algorithm for Training Algorithmic Decision-Makers using Gradient Descent with SPSA for Gradient Approximation and Random Restarts upon Convergence

$\mathcal{L}_\theta^{*\min} = \infty$
$\theta^* =$ None
while time elapsed $<$ 12 hours **do**
  $\mathcal{L}_\theta^{*\text{prev}} = \infty$
  Randomly initialize θ using Glorot Uniform distribution for $h_\theta(\cdot)$
  $k = 0$
  for $j \leftarrow 0$ to $m-1$ **do**
    Calculate $\mathcal{L}_\theta^*(\cdot) \forall x_i$
    if $\mathcal{L}_\theta^*(\cdot) < \mathcal{L}_\theta^{*\min}$ **then** ▷ Keeping track of global minimum for $\mathcal{L}_\theta^*(\cdot)$.
      $\mathcal{L}_\theta^{*\min} = \mathcal{L}_\theta^*(\cdot)$
      $\theta^* = \theta$
    end if
    if $|\mathcal{L}_\theta^{*\text{prev}} - \mathcal{L}_\theta^*(\cdot)| < \epsilon_{\text{conv}}$ **then** ▷ Checking for convergence.
      break ▷ Exit for-loop.
    else
      $\mathcal{L}_\theta^{*\text{prev}} = \mathcal{L}_\theta^*(\cdot)$
    end if
    for each batch x_b in $\forall x_i$ of size n_{batch} **do** ▷ The batch division is randomized.
      $\Delta_k = \{\Delta_{k1}, \Delta_{k2}, ..., \Delta_{kn}\}$ ▷ Where each Δ_{ki} is independently sampled from a mean-zero distribution, and $|\Delta_k| = |\theta|$. (Note, the Δ symbol is common in the SPSA literature, and does not have the same meaning as those in Section 4 or Appendix D.4)
      Calculate $\mathcal{L}^*_{\theta+c_k\Delta_k}(\cdot)$ and $\mathcal{L}^*_{\theta-c_k\Delta_k}(\cdot)$ for x_b ▷ *All* the dimensions of θ are being perturbed simultaneously, for both $h^*_{\theta+c_k\Delta_k}(\cdot)$ and $h^*_{\theta-c_k\Delta_k}(\cdot)$. Therefore, $A_\theta(\cdot)$ is only called four times in this step for each record, rather than $4n$ times. c_k is an adaptive value per gradient update, k, we will define below.
      $\frac{\partial \hat{\mathcal{L}}_\theta^*(\cdot)}{\partial \theta} = \frac{\mathcal{L}^*_{\theta+c_k\Delta_k}(\cdot) - \mathcal{L}^*_{\theta-c_k\Delta_k}(\cdot)}{2c_k} \begin{bmatrix} 1/\Delta_{k1} \\ 1/\Delta_{k2} \\ \dots \\ 1/\Delta_{kn} \end{bmatrix}$
      $\theta = \theta - \eta_k \frac{\partial \hat{\mathcal{L}}_\theta^*(\cdot)}{\partial \theta}$ ▷ Note that η_k is an adaptive learning rate per gradient update, k, which we will define below.
      $k = k + 1$
    end for
  end for
end while
return $\{\mathcal{L}_\theta^{*\min}, \theta^*\}$ ▷ Returns model with global lowest loss, $\mathcal{L}_\theta^{*\min}$.

on the gradient update count, k, we use the following calculation from [46]:

$$\eta_k = \frac{a}{(A+k+1)^\alpha}$$

Similarly, we use $\alpha = 0.602$, because, as noted by [46], this guarantees practical effectiveness and theoretical validity. We use the common choice of $a = 0.16$, and to encourage more aggressive exploration of the θ domain in early iterations, we set $A = 0$, given that A is a stabilizing term that specifically impacts the learning rate for early iterations.

Spall [45] provides a proof of convergence to local minimum if $\mathcal{L}_\theta^*(\cdot)$ is non-convex and to a global minimum if $\mathcal{L}_\theta^*(\cdot)$ is convex, that adopts a set of assumptions. Therefore, our parametric choices satisfy the assumptions pertaining to the parameters listed above stated in [45].

The loss function, $\mathcal{L}_\theta^*(\cdot)$, we propose in Section 3 is not convex, therefore, a singular run of the gradient descent algorithm will most likely not find a near-global minimum for $\mathcal{L}_\theta^*(\cdot)$. As shown in Algorithm 1, we perform a random reset to the parameters, θ, once our loss function $\mathcal{L}_\theta^*(\cdot)$ converges. The convergence criterion that we utilize is $\epsilon_{\text{conv}} = 0.01$ for all our results to allow for more random resets, and therefore, more aggressive searching of the parameter space for θ. Across all our models displayed in Table 9, a model is expected to perform a random reset 23.63 times within the 12 hour timeframe. While this does not guarantee that the global minimum or near-global minimum is found for all algorithmic

decision-makers, which is the case for all stochastic approximation methods for non-convex loss functions, it is inline with common practitioner heuristics for the number of random restarts necessary to find a near-optimal solution for low dimensional data settings like the ones utilized in this research. Lastly, we use a batch size of $n_{\text{batch}} = 350$ across all our data settings. This was dictated by the computational resources that we had available (60 GB of memory and 25 cores per training each algorithmic decision-maker) for parallel calls to our recommendation generator, $A_\theta(\cdot)$.

D Evaluation Appendices

D.1 Data Settings

D.1.1 Simulated Mortgage Lending Setting. Given that our method utilizes structural causal models (SCMs), it is important to have a setting where all necessary assumptions are satisfied. This provides a baseline for how our method would perform in a well-behaved scenario where we can guarantee all necessary assumptions. Therefore, these data are synthetic and produced by a simulation outline in Algorithm 2. Algorithm 2 shows the same underlying data generation process displayed in Figure 1. An important aspect to note is that the outcome variable of whether an individual is trustworthy for mortgage lending is directly estimated using solely an individual's credit score, but given the causal relationship of all the features, the outcome variable is correlated with gender, age, and proportion of cumulative credit available.

As described in Section 2.1, in Equation 2 and Equation 3, action sets must satisfy data setting-specific criteria to ensure that they are realistic and implementable for individuals. This is encoded as the constraint $\delta_i \in F(x_i)$. We include the constraints on action sets for the Simulated Mortgage Lending Data setting in Table 3.

We note that this is a simplified, simulated scenario rather than a complete and accurate model of mortgage lending. We do not make any claims here about the true functional relationships between variables that determine whether an individual is trustworthy for mortgage lending.

D.1.2 German Credit Data Setting. The SCM, including graph and structural equations, we use for the German Credit Data setting is displayed in Figure 5a. Note, this is the identical SCM for the German Credit Data setting that was used in [21] and we use the same constraints for action sets, $F(\cdot)$, utilized in [21], as well, as displayed in Table 4.

D.1.3 First-Year Law School Success Setting. The graph of the SCM we utilize for the First-Year Law School Success setting, displayed in Figure 5b, has been utilized in prior research including [3, 26]. It is important to note that we opt to exclude the outcome variable from the graph. In other research, this is often included, but it is a non-consequential choice, given that this outcome variable is a leaf node with no descendants, and therefore, no causal influence on any part of the data distribution. Additionally, we filter the First-Year Law School Success dataset [55] to only include students from the Southeast region of the United States, and only include Black and White students. The first choice is to ensure that the region is not an unaccounted-for confounder in our causal model. The second choice was determined by the current state of our research design. To fit each structural function, $f_3(\cdot)$ and $f_4(\cdot)$, we train two linear regressions with the features X_1 (race) and X_2 (gender). One linear regression is fitted with outcome variable X_3 (LSAT score) and the other with the outcome variable X_4 (undergraduate GPA). The coefficients for X_1 and X_2 for each linear regression were rounded to the nearest 0.5, to ensure that realistic values for GPA and LSAT score were produced, and adopted in the structural functions $f_3(\cdot)$ and $f_4(\cdot)$. Given that the intercepts for these linear regressions are estimates of signal that is not correlated with race or gender, we assume this signal is represented in the exogenous variable distributions for LSAT score and undergraduate GPA, which are observed and do not need codifying in the SCM. To the best of our knowledge, Bynum et al. [3] use a similar process for modeling structural functions.

Lastly, the constraints we impose on the action sets for the First-Year Law School Success data setting are displayed in Table 5.

D.2 Benchmark Methods

The four benchmark methods we use in Section 4, $h_\theta^{0/1}(\cdot)$, $h_\theta^{\text{eq-cost}}(\cdot)$, $h_\theta^{\text{bal-err}}(\cdot)$, and $h_\theta^{\text{bal+eq}}(\cdot)$, are explained in this appendix.

The benchmark method, $h_\theta^{0/1}(\cdot)$, only minimizes balanced 0/1 loss, therefore the loss we minimize for these models take the form of the following:

$$\mathcal{L}_\theta^{0/1}(\cdot) = \mathcal{L}_\theta^{\text{acc}}(\cdot) = \beta_{FP} * FPR + \beta_{FN} * FNR \tag{15}$$

For Equation 15, $\mathcal{L}_\theta^{\text{acc}}(\cdot)$ is identical to Equation 10 when $\beta_{TN}^{\text{burd}} = 0$. The β_{FP} and β_{FN} parameters we use for all data settings are defined in Table 6.

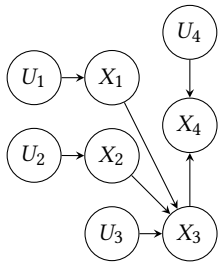
The benchmark method, $h_\theta^{\text{eq-cost}}(\cdot)$, is trained to minimize balanced 0/1 loss and equalize the cost of recommendations across sensitive subpopulations, using the formulation in Equation 5. It is important to note that the original formulation proposed in [15] is directly incorporated into a optimization problem, rather than an isolated fairness measurement. This means that Equation 5 was originally displayed as a constraint for training a classifier, with a parametric upper bound for the tolerated value of Equation 5. In reviewing our results for the German Credit Data, in comparison to their results for the German Credit Data, specifically in terms of the measurement for Equation 5, which is referred to as 'Δ_a for $\mathbb{E}$(Cost)' in Table 1, our adaption seems empirically neutral, if not beneficial. We train $h_\theta^{\text{eq-cost}}(\cdot)$ using the following loss function:

$$\begin{aligned}\mathcal{L}_\theta^{\text{eq-cost}}(\cdot) =& \mathcal{L}_\theta^{\text{acc}}(\cdot) + \beta^{\text{eq-cost}} * \alpha^{\text{eq-cost}},\\ \text{where } \alpha^{\text{eq-cost}} =& \left| \frac{1}{|\hat{S}_a^-|} \sum_{x_i \in \hat{S}_a^-} \text{cost}(x_i, \delta_i^*) - \frac{1}{|\hat{S}_{a'}^-|} \sum_{x_j \in \hat{S}_{a'}^-} \text{cost}(x_j, \delta_j^*) \right|,\end{aligned} \tag{16}$$

as defined in Equation 5.

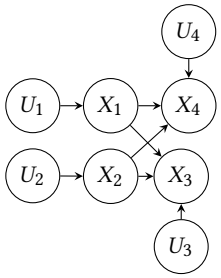
In Equation 16, as in Equation 5, for a sensitive attribute $x_{ij} \in x_i$, $\hat{S}_a^- = \{x_i \in x : h_\theta(x_i) = 0, x_{ij} = a\}$ and $\hat{S}_{a'}^- = \{x_i \in x : h_\theta(x_i) = 0, x_{ij} = a'\}$. We provide the setting for $\beta^{\text{eq-cost}}$ of $h_\theta^{\text{eq-cost}}(\cdot)$ in Table 8.

The benchmark method, $h_\theta^{\text{bal-err}}(\cdot)$, is trained to minimize balanced 0/1 loss and equalize the error rates (false positive rate and



$$X_1 := U_1$$
$$X_2 := U_2$$
$$X_3 := f_3(X_1, X_2) + U_3$$
$$X_4 := f_4(X_3) + U_4$$

(a) The structural causal model (SCM) $\mathcal{M}$ for the German Credit Data setting. Endogenous variables: X_1 is gender. X_2 is age. X_3 is credit amount. X_4 is repayment duration in months. Endogenous variables are non-parametric observed data distributions. Structural functions: $f_3(X_1, X_2) = 550 * X_1 + 4.5 * X_2$. $f_4(X_3) = .0025 * X_3$. Outcome variable (not shown above): whether an individual is trustworthy for credit.



$$X_1 := U_1$$
$$X_2 := U_2$$
$$X_3 := f_3(X_1, X_2) + U_3$$
$$X_4 := f_4(X_1, X_2) + U_4$$

(b) The structural causal model (SCM) $\mathcal{M}$ for First-Year Law School Success Data setting. Endogenous variables: X_1 is race (Black or White). X_2 is gender. X_3 is LSAT score. X_4 is undergraduate GPA. Endogenous variables are non-parametric observed data distributions. Structural functions: $f_3(X_1, X_2) = -8.5 * X_1 + 0.5 * X_2$. $f_4(X_1, X_2) = -0.3 * X_1 - 0.2 * X_2$. Outcome variable (not shown above): whether an individual performs above average in their first-year of law school.

Figure 5: Structural Causal Models (SCM) for German Credit Data and First-Year Law School Success settings. The SCM for the Simulated Mortgage Lending setting is shown in Figure 1. For more information on structural causal models, reference Appendix B.

Algorithm 2 Process for Generating Simulated Mortgage Lending Data

for i = {1,2,...,1000} **do**
  $x_{i1} = u_{i1} \sim \text{Bernoulli}(0.50)$ ▷ Sampling gender for individual i.
  $x_{i2} = u_{i2} = \infty$
  while $x_{i2} < 18$ or $x_{i2} > 75$ **do**
    $x_{i2} = u_{i2} \sim \mathcal{N}(38, 22)$ ▷ Sampling age for individual i, rounded to integer values.
  end while
  $u_{i3} \sim \text{Uniform}(0.40, 1.09)$ ▷ Rounded to the hundredths place value.
  $x_{i3} = -0.005 * x_{i2} + u_{i3}$ ▷ Calculating proportion of cumulative credit line available.
  $u_{i4} \sim \text{Uniform}(200, 400)$ ▷ Rounded to integer value.
  $x_{i4} = x_{i1} + 50 * x_{i2} + 400 * x_{i3} + u_{i4}$ ▷ Calculating credit score for individual i.
end for
$\mu_4 = \mathbb{E}_{\forall x_i}[x_i]$ ▷ Calculate mean of credit scores across all individuals.
$s_4 = \text{std}_{\forall x_i}(x_i)$ ▷ Calculate standard deviation of credit scores for all individuals.
for i = {1,2,...,1000} **do**
  $z_{i4} = \frac{x_{i4} - \mu_4}{s_4}$ ▷ Standardizing credit score for individual i.
  $y_i \sim \text{Bernoulli}(\sigma(\mathcal{N}(0.05 * z_{i4}, 0.20)))$ ▷ Drawing outcome variable of mortgage trustworthiness using credit score for individual i. σ is the sigmoid function.
end for
return $\{x_{i1}, x_{i2}, x_{i3}, x_{i4}, y_i\} \forall i \in \{1, 2, ..., 1000\}$

false negative rate) across sensitive subpopulations. Given that part of our method is aimed at addressing the issue of imbalanced rates of erroneously granting and denying access to the favorable decision across sensitive subpopulation, as motivated in Example 1 of Section 2.2, this takes the form of imbalanced error rates across sensitive subpopulations, and therefore, is an appropriate benchmark. We train $h_\theta^{\text{bal-err}}(\cdot)$ using the following loss functions:

Variable	Description	Actionability	Direction	Mutability	Min	Max	Data Type
X_1	gender	non-actionable		non-mutable			
X_2	age	actionable	increase	mutable		75 years-old	integer
X_3	prop. of cuml. credit available	actionable	any direction	mutable	0.045	0.99	real
X_4	credit score	non-actionable		mutable	321	847	integer

Table 3: Feasibility and plausibility constraints for action sets encoded in $F(\cdot)$ for the Simulated Mortgage Lending setting ($n = 1000$). Actionability refers to whether a feature can be used as an action in a recommendation. Direction describes how a feature can be modified. Mutability describes whether a feature can change due to a hard intervention. All actionable features are mutable, but some mutable features are not actionable. Min and max constraints are set for mutable features based on the bounds observed in the data distribution for each feature. Data type refers to data type constraints placed on feature values.

Variable	Description	Actionability	Direction	Mutability	Min	Max	Data Type
X_1	gender	non-actionable		non-mutable			
X_2	age	actionable	increase	mutable		75 years-old	integer
X_3	credit (in Deutsche Marks)	actionable	any direction	mutable	250	18424	real
X_4	loan duration (in months)	non-actionable		mutable	4	72	integer

Table 4: Feasibility and plausibility constraints for action sets encoded in $F(\cdot)$ for the German Credit Data setting ($n = 1000$). Actionability refers to whether a feature can be used as an action in a recommendation. Direction describes how a feature can be modified. Mutability describes whether a feature can change due to a hard intervention. All actionable features are mutable, but some mutable features are not actionable. Min and max constraints are set for mutable features based on the bounds observed in the data distribution for each feature. Data type refers to data type constraints placed on feature values.

Variable	Description	Actionability	Direction	Mutability	Min	Max	Data Type
X_1	race (Black or White)	non-actionable		non-mutable			
X_2	gender	non-actionable		non-mutable			
X_3	LSAT score	actionable	any direction	mutable	17	48	real
X_4	undergraduate GPA	non-actionable		non-mutable			

Table 5: Feasibility and plausibility constraints for action sets encoded in $F(\cdot)$ for the First-Year Law School Success Data setting ($n = 2421$). Note, the LSAT score uses the scoring schema between 1995-2005. Actionability refers to whether a feature can be used as an action in a recommendation. Direction describes how a feature can be modified. Mutability describes whether a feature can change due to a hard intervention. All actionable features are mutable, but some mutable features are not actionable. Min and max constraints are set for mutable features based on the bounds observed in the data distribution for each feature. Data type refers to data type constraints placed on feature values.

$$\mathcal{L}_\theta^{\text{bal-err}}(\cdot) = \mathcal{L}_\theta^{\text{acc}}(\cdot) + \beta^{\text{bal-err}}\left(\frac{\alpha^{bal-FPR} + \alpha^{bal-FNR}}{2}\right),$$
$$\text{where } \alpha^{\text{bal-FPR}} = \left|\mathbb{E}_{x_i \sim S_a^-}[\mathbb{I}(h_\theta(x_i) = 1)] - \mathbb{E}_{x_j \sim S_{a'}^-}[\mathbb{I}(h_\theta(x_j) = 1)]\right|$$
$$\text{and } \alpha^{\text{bal-FNR}} = \left|\mathbb{E}_{x_k \sim S_a^+}[\mathbb{I}(h_\theta(x_k) = 0)] - \mathbb{E}_{x_l \sim S_{a'}^+}[\mathbb{I}(h_\theta(x_l) = 0)]\right|. \tag{17}$$

For Equation 17, similarly to Equation 13, $S_a^- = \{x_i : y_i = 0, x_{ij} = a\}$, $S_{a'}^- = \{x_i : y_i = 0, x_{ij} = a'\}$, $S_a^+ = \{x_i : y_i = 1, x_{ij} = a\}$ and $S_{a'}^+ = \{x_i : y_i = 1, x_{ij} = a'\}$. We provide the setting for $\beta^{\text{bal-err}}$ of $h_\theta^{\text{bal-err}}(\cdot)$ in Table 8.

Lastly, for the benchmark method, $h_\theta^{\text{bal+eq}}(\cdot)$, we minimize balanced 0/1 loss, and equalize cost of recommendations *and* error rates across sensitive subpopulations. As mentioned in Section 4, this benchmark serves to demonstrate that jointly optimizing for predictive fairness and equalized cost of recommendations is not competitive compared to our method. Therefore, we use the following loss function to train $h_\theta^{\text{bal+eq}}(\cdot)$:

$$\mathcal{L}_\theta^{\text{bal+eq}} = \mathcal{L}_\theta^{\text{acc}}(\cdot) + \beta^{\text{eq-cost}} * \alpha^{\text{eq-cost}} + \beta^{\text{bal-err}}\left(\frac{\alpha^{bal-FPR} + \alpha^{bal-FNR}}{2}\right) \tag{18}$$

For Equation 18, $\alpha^{\text{eq-cost}}$ is defined in Equation 16 and Equation 5. $\alpha^{bal-FPR}$ and $\alpha^{bal-FNR}$ are defined in Equation 17. $\beta^{\text{eq-cost}}$ and $\beta^{\text{bal-err}}$ for $h_\theta^{\text{bal+eq}}(\cdot)$ are shown in Table 8.

The loss functions, $\mathcal{L}_\theta^{0/1}(\cdot)$, $\mathcal{L}_\theta^{\text{eq-cost}}(\cdot)$, $\mathcal{L}_\theta^{\text{bal-err}}(\cdot)$, and $\mathcal{L}_\theta^{\text{bal+eq}}(\cdot)$ for $h_\theta^{0/1}(\cdot)$, $h_\theta^{\text{eq-cost}}(\cdot)$, $h_\theta^{\text{bal-err}}(\cdot)$, and $h_\theta^{\text{bal+eq}}(\cdot)$, respectively, take a similar form to the loss function, $\mathcal{L}_\theta^*(\cdot)$, we use to train our method, $h_\theta^*(\cdot)$. Therefore, we use the identical training method we discussed in Section 4 of gradient descent using SPSA for gradient approximation with random sets of θ post-convergence. We outline this process in detail in Appendix C.1, with pseudocode in Algorithm 1. For Algorithm 1 for these benchmark methods, as well as all other

models that use loss functions other than $\mathcal{L}^*_\theta(\cdot)$, we substitute the respective loss function in the pseudocode for $\mathcal{L}^*_\theta(\cdot)$. Note that all other parameters pertaining to the gradient descent algorithm, including batch size and convergence criterion, and SPSA gradient approximation method are identical across all models, including those for these benchmark methods.

D.3 Model Parameters

For each data setting, across all models used in our research, the same β_{FP} and β_{FN} are utilized. These are displayed in Table 6. Additionally, as discussed in Section 3.2, a ϵ_{corr} penalty is added to the false positive rate, where $\epsilon_{\text{corr}} \approx \beta^{\text{burd}}_{TN}\mathbb{E}_{x_i \sim TN}[b(x_i)]$ when $\beta^{\text{burd}}_{TN} > 0$. We approximate a fixed $\mathbb{E}_{x_i \sim TN}[b(x_i)]$ for each data setting, using a default logistic regression classifier, that we use across all models per data setting when $\beta^{\text{burd}}_{TN} > 0$. These values of $\mathbb{E}_{x_i \sim TN}[b(x_i)]$ for ϵ_{corr} are also displayed in Table 6.

In determining the weights we used for each data setting, displayed in Table 6, the weights β_{FP} and β_{FN} must be set accordingly to ensure there exists a θ for $\mathcal{L}^*_\theta(\cdot)$ where:

$$\beta_{FP} \gg \beta_{FN} * FNR + \mathcal{L}^{\text{burd}}_\theta(\cdot) + \mathcal{L}^{\text{exc-burd}}_\theta(\cdot) + \mathcal{L}^{\text{bal-exc-burd}}_\theta(\cdot) \quad (19)$$

If a solution for θ does not exist that satisfies Equation 19, given β_{FP} and β_{FN}, the model that minimizes $\mathcal{L}^*_\theta(\cdot)$ will provide all or nearly all positive predictions. Therefore, the different parameter weights for each data setting in Table 6 were determined to ensure the constraint in Equation 19 was met. Note that we use ϵ_{corr} to solely mitigate the issue discussed in Section 3.2 of minimization of burden through flipping true negatives to false positives, therefore, specifically addressing an issue the $\mathcal{L}^{\text{burd}}_\theta(\cdot)$ component presents, in isolation. The weights for β_{FP} and β_{FN} are chosen to address an issue that is present for $\mathcal{L}^*_\theta(\cdot)$ at large. We frame this specifically in Equation 19 as a consideration for $\mathcal{L}^*_\theta(\cdot)$ for the weights assigned in Table 6, because $\beta_{FN} * FNR + \mathcal{L}^{\text{burd}}_\theta(\cdot) + \mathcal{L}^{\text{exc-burd}}_\theta(\cdot) + \mathcal{L}^{\text{bal-exc-burd}}_\theta(\cdot)$ serves as an upper bound across all our models for each data setting, but it is clear that this consideration exists for all loss functions we utilize in our research, and therefore, this is why we use fixed β_{FP} and β_{FN} weights across all models given the data setting.

Table 7 presents all the weights we utilize for $h^*_\theta(\cdot)$, all the ablated models, $h^{\text{burd}}_\theta(\cdot)$, $h^{\text{exc-burd}}_\theta(\cdot)$, and $h^{\text{bal-exc-burd}}_\theta(\cdot)$, and models that are trained to prioritize access for false negatives over true negatives by placing more weight on minimizing burden for false negatives over true negatives. These models are denoted as $h^{\text{burd}}_{\theta,\text{FN>TN}}(\cdot)$ and $h^*_{\theta,\text{FN>TN}}(\cdot)$, and their weights are displayed in Table 7 and the results of these models are displayed in Table 9. Lastly, we present the weights for all benchmark methods in Table 8, and discuss the loss functions used for these benchmark methods in Appendix D.2. **All models, including those displayed in Table 7 and Table 8, use the weights per data setting displayed in Table 6.**

For all models other than $h^{0/1}_\theta(\cdot)$, all weights not pertaining to β_{FP} and β_{FN} sum to 6. This can easily be observed by looking at the sum of the rows for all models in Table 7 and Table 8, besides the row for $h^{0/1}_\theta(\cdot)$. Additionally, the metrics, including burden, excess burden, and the benchmark metrics have a minimum value of 0 and maximum value of 1. This presents a level of stability across all models per data setting, making them relatively comparable to each other in terms of the weight they assign to objectives other than predictive accuracy in the loss functions.

D.4 Full Results

Table 9 contains the full set of results across of all our evaluation settings and benchmark methods. We describe the benchmark methods, $h^{0/1}_\theta(\cdot)$, $h^{\text{eq-cost}}_\theta(\cdot)$, $h^{\text{bal-err}}_\theta(\cdot)$ and $h^{\text{bal+eq}}_\theta(\cdot)$, extensively in Section 4 and Appendix D.2. Our method, $h^*_\theta(\cdot)$, is described thoroughly in Section 3.2 and Section 4. The ablated classifiers, $h^{\text{burd}}_\theta(\cdot)$, $h^{\text{exc-burd}}_\theta(\cdot)$, and $h^{\text{bal-exc-burd}}_\theta(\cdot)$, are described in Section 4.1. Models trained to prioritize accessibility for those who are false negatives, $h^{\text{burd}}_{\theta,\text{FN>TN}}(\cdot)$ and $h^*_{\theta,\text{FN>TN}}(\cdot)$, are discussed in Appendix D.3. It is important to note that false negatives are the result of an algorithmic decision-maker being unable to detect a substantial enough signal in their features to discern them from true negatives. Therefore, in many regards, these models' ($h^{\text{burd}}_{\theta,\text{FN>TN}}(\cdot)$ and $h^*_{\theta,\text{FN>TN}}(\cdot)$) ability to give more access to false negatives relies on the assumption that they are able to fit for a signal of these false negatives. If the models were effectively able to do that, they would be classifying these false negatives as true positives. Regardless, these models present a specific functionality of our method that could be useful to practitioners in other contexts.

Lastly, we present an algorithmic decision-maker that is trained to only minimize excess burden and balance excess burden across sensitive subpopulations. Therefore, this model is not concerned with lowering the cost of recommendations or maximizing access to recommendations. We denote these algorithmic decision-makers as $h^{\text{all-exc-burd}}_\theta(\cdot)$ in Table 9 with the weights they utilize specified in Table 7. We provide this setting for the use-case where access to realistically implementable recommendations is not a goal, meaning, Principle A is not applicable. Adopting this criterion for a data setting should be taken with caution for two reasons. First, all objectives we use to train $h^*_\theta(\cdot)$ work in tandem. For example, lowering the expected cost of recommendations, often, also lowers the absolute difference of expected costs of recommendations across sensitive subpopulations. Second, as pointed out in Section 3.2 and cited in [47], there are ethical issues associated with providing challenging (high-cost) recommendations for individuals to implement. We address this in our work by minimizing expected burden. Therefore, $h^{\text{all-exc-burd}}_\theta(\cdot)$ should only be used in select settings where it has been evaluated as essential by domain experts.

Data Setting	β_{FP}	β_{FN}	ϵ_{corr} for all models where $\beta_{TN}^{\text{burd}} > 0$ in Table 7
Simulated Mortgage Lending	4	4	$0.70\beta_{TN}^{\text{burd}}$
German Credit Data	6	6	$0.30\beta_{TN}^{\text{burd}}$
First-Year Law School Success	8	8	$0.85\beta_{TN}^{\text{burd}}$

Table 6: Weights for false positive rates and false negative rates used for each data setting across all models. ϵ_{corr} is used when $\beta_{TN}^{\text{burd}} > 0$, where $\epsilon_{\text{corr}} \approx \beta_{TN}^{\text{burd}} \mathbb{E}_{x_i \sim TN}[b(x_i)]$ and $\mathbb{E}_{x_i \sim TN}[b(x_i)]$ is calculated using a default classifier and is fixed for each data setting. These values of $\mathbb{E}_{x_i \sim TN}[b(x_i)]$ are displayed in the right-most column of this table.

Algorithmic Decision-Maker	β_{FN}^{burd}	β_{TN}^{burd}	$\beta_{FN}^{\text{exc-burd}}$	$\beta_{TN}^{\text{exc-burd}}$	$\beta_{-}^{\text{bal-exc-burd}}$	$\beta_{+}^{\text{bal-exc-burd}}$
h^{burd}	3	3	0	0	0	0
$h^{\text{exc-burd}}$	0	0	3	3	0	0
$h^{\text{bal-exc-burd}}$	0	0	0	0	3	3
$h^{\text{all-exc-burd}}$	0	0	1.5	1.5	1.5	1.5
$h^{\text{burd}}_{\text{FN>TN}}$	4	2	0	0	0	0
$h^{*}_{\text{FN>TN}}$	1.33	0.67	1	1	1	1
h^{*}	1	1	1	1	1	1

Table 7: Weights utilized in the loss functions for all models, including our method, $h_\theta^*(\cdot)$, the ablated models, $h_\theta^{\textbf{burd}}(\cdot)$, $h_\theta^{\textbf{exc-burd}}(\cdot)$, and $h_\theta^{\textbf{bal-exc-burd}}(\cdot)$, models that prioritize access to recommendations for false negatives over true negatives, $h_{\theta,\textbf{FN>TN}}^{\textbf{burd}}(\cdot)$ and $h_{\theta,\textbf{FN>TN}}^{*}(\cdot)$, and an algorithmic decision-maker that only minimizes excess burden and balances excess burden across sensitive subpopulations, $h_\theta^{\textbf{all-exc-burd}}(\cdot)$ (discussed more in Appendix D.4). For readability in the table, we neglect the θ subscript and $(\cdot)$ notation. Note that all algorithmic decision-makers in this table use the weights listed in Table 6, as well.

Algorithmic Decision-Maker	$\beta^{\text{eq-cost}}$	$\beta^{\text{bal-err}}$
$h^{0/1}$	0	0
$h^{\text{eq-cost}}$	6	0
$h^{\text{bal-err}}$	0	6
$h^{\text{bal+eq}}$	3	3

Table 8: Weights used for each algorithmic decision-maker used as a benchmark for our methods. These weights are used in the loss functions described in Appendix D.2. These benchmark methods are also described at a high-level in Section 4. For readability in the table, we neglect the θ subscript and $(\cdot)$ notation. Note that all algorithmic decision-makers in this table use the weights listed in Table 6, as well.

Data Setting	Model	Algorithmic Decision-Maker	Balanced 0/1 Loss	Coverage	$\lvert\Delta_a\rvert$ for Coverage	$\mathbb{E}$(Cost)	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) ($\alpha^{\text{eq-cost}}$)	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) $\forall\, y_i = 1$	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) $\forall\, y_i = 0$	$\alpha^{\text{ind-fair}}$	$\lvert\Delta_a\rvert$ for FPR	$\lvert\Delta_a\rvert$ for FNR
Simulated Mortgage Lending	lr	$h^{0/1}$	0.508	0.684	0.156	0.094	0.036 †	0.042 †	0.034 †	0.108 ‡	0.289	0.272
		$h^{\text{eq-cost}}$	0.511	0.729	0.225 †	0.090 †	0.013 †	0.015 †	0.016 †	0.052 ‡	0.240	0.232
		$h^{\text{bal-err}}$	0.501	0.667	0.049	0.100	0.022 †	0.025 †	0.027 †	0.026 ‡	0.047	0.053
		$h^{\text{bal+eq}}$	0.498	0.738	0.046	0.082	0.010 †	0.015 †	0.016 †	0.013 ‡	0.062	0.044
		h^{burd}	0.505	**1.000**	**0.000** †	**0.051**	0.012 †	0.013 †	0.013 †	0.030 ‡	0.153	0.138
		$h^{\text{exc-burd}}$	0.509	0.761	0.030	0.087	0.015	0.018 †	0.020 †	0.006	0.057	0.055
		$h^{\text{bal-exc-burd}}$	0.509	0.794	**0.023**	0.084 †	0.011 †	0.015 †	0.016 †	0.004 ‡	0.052	0.055
		$h^{\text{all-exc-burd}}$	0.507	0.810	**0.024**	0.086	0.012	0.018	0.017	0.004	0.057	0.052
		$h^{\text{burd}}_{\text{FN>TN}}$	0.508	**1.000**	**0.000** †	**0.047**	0.014 †	0.017 †	0.016 †	0.046 ‡	0.204	0.188
		$h^{*}_{\text{FN>TN}}$	0.507	**0.987**	**0.005**	**0.053**	0.008	0.012	**0.010** †	0.008	0.063	0.052
		h^{*}	0.509	**0.998**	**0.000**	**0.059**	**0.006**	**0.008**	**0.010**	0.003	0.058	0.052
	MLP (1x2)	$h^{0/1}$	0.514	0.653	0.203 †	0.089 †	0.026 †	0.027 †	0.029 †	0.088 ‡	0.294	0.311
		$h^{\text{eq-cost}}$	0.511	0.771	0.117 †	0.089	0.013 †	0.018 †	0.018 †	0.051 ‡	0.268	0.271
		$h^{\text{bal-err}}$	0.505	0.763	0.031	0.093	0.018 †	0.014 †	0.019 †	0.014 ‡	0.056	0.055
		$h^{\text{bal+eq}}$	0.504	0.812	0.035	0.088	0.011	0.012	0.017	0.012	0.053	0.053
		h^{burd}	0.510	**1.000**	**0.000**	**0.051**	0.010	0.012	**0.011**	0.023	0.112	0.124
		$h^{\text{exc-burd}}$	0.508	0.770	0.035	**0.081**	0.012	0.013 †	0.017 †	0.013	0.054	0.060
		$h^{\text{bal-exc-burd}}$	0.506	0.795	0.028	0.084	0.014	0.018 †	0.014 †	**0.002***	0.062	0.053
		$h^{\text{all-exc-burd}}$	0.509	0.870	0.035	0.083	0.010	0.017	0.017	**0.006***	0.060	0.050
		$h^{\text{burd}}_{\text{FN>TN}}$	0.511	**1.000**	**0.000** †	**0.044**	0.013 †	0.015 †	0.014 †	0.041 ‡	0.167	0.187
		$h^{*}_{\text{FN>TN}}$	0.514	**0.992**	**0.003**	**0.054**	**0.008**	0.010	**0.011**	0.009	0.063	0.050
		h^{*}	0.511	**0.999**	**0.001**	**0.058**	**0.007**	0.010	**0.011**	**0.004***	0.054	0.047
	MLP (1x4)	$h^{0/1}$	0.515	0.585	0.196 †	0.095	0.029 †	0.030 †	0.034 †	0.113 ‡	0.296	0.300
		$h^{\text{eq-cost}}$	0.510	0.717	0.212 †	0.082 †	0.014 †	0.019 †	0.016 †	0.080 ‡	0.257	0.258
		$h^{\text{bal-err}}$	0.507	0.805	0.034	0.090 †	0.012 †	0.013 †	0.017 †	0.013 ‡	0.051	0.057
		$h^{\text{bal+eq}}$	0.508	0.809	0.039	0.089	0.010	0.018	0.015	0.017	0.057	0.059
		h^{burd}	0.512	**1.000**	**0.000**	**0.050**	0.011	0.014	0.012 †	0.027	0.132	0.124
		$h^{\text{exc-burd}}$	0.507	0.800	0.041	0.084	0.011	0.017	0.015 †	**0.006***	0.055	0.053
		$h^{\text{bal-exc-burd}}$	0.510	0.789	0.024	0.088	0.016	0.014 †	0.025 †	**0.006***	0.054	0.047
		$h^{\text{all-exc-burd}}$	0.509	0.839	0.021	0.083	0.011	0.016	0.016	**0.005***	0.054	0.057
		$h^{\text{burd}}_{\text{FN>TN}}$	0.515	**1.000**	**0.000** †	**0.042**	0.017 †	0.017 †	0.014 †	0.048 ‡	0.172	0.194
		$h^{*}_{\text{FN>TN}}$	0.511	**0.988**	**0.004**	**0.054**	0.008	0.012 †	0.012 †	0.010	0.061	0.054
		h^{*}	0.514	**1.000**	**0.000**	**0.054**	**0.006**	**0.009**	**0.009**	**0.003***	0.067	0.059
German Credit Data	lr	$h^{0/1}$	0.412	0.992	0.006	0.083	0.019	0.019	0.022	0.123	0.388	0.353
		$h^{\text{eq-cost}}$	0.414	0.966	0.027	0.078	0.014	0.012	0.022	0.110 ‡	0.318	0.281
		$h^{\text{bal-err}}$	0.433	0.936	0.026	0.082	0.013	0.014	0.022	0.054	0.081	0.060
		$h^{\text{bal+eq}}$	0.430	0.952	0.032	0.086	0.012	0.015	0.021	0.048	0.077	0.067
		h^{burd}	0.419	**0.998**	**0.003**	**0.043**	**0.007**	**0.009**	**0.009**	0.066	0.304	0.252
		$h^{\text{exc-burd}}$	0.424	0.985	0.012	0.074	0.010	0.010	0.018	0.044	0.18	0.126
		$h^{\text{bal-exc-burd}}$	0.430	0.966	0.016	0.073	0.010	0.011	0.016	**0.033***	0.138	0.088
		$h^{\text{all-exc-burd}}$	0.427	0.956	0.021	0.080	**0.009**	**0.010**	0.017	**0.035***	0.145	0.092
		$h^{\text{burd}}_{\text{FN>TN}}$	0.424	0.996	0.006	**0.049**	0.010	0.011	**0.013**	0.070	0.311	0.243
		$h^{*}_{\text{FN>TN}}$	0.431	0.993	0.006	**0.054**	**0.007**	**0.008**	**0.012**	0.040	0.153	0.111
		h^{*}	0.423	0.996	0.006	**0.057**	**0.008**	**0.009**	**0.015**	**0.036***	0.165	0.107
	MLP (1x2)	$h^{0/1}$	0.408	0.975	0.014	0.083	0.018	0.017	0.025	0.135	0.411	0.383
		$h^{\text{eq-cost}}$	0.414	0.957	0.031	0.078	0.012	0.012	0.017	0.110 ‡	0.337	0.306
		$h^{\text{bal-err}}$	0.424	0.892	0.043	0.083	0.015	0.015	0.024	0.069	0.077	0.068
		$h^{\text{bal+eq}}$	0.431	0.891	0.037	0.078	0.014	0.016	0.023	0.054	0.089	0.078
		h^{burd}	0.423	**0.997**	**0.006**	**0.041**	**0.007**	**0.008**	**0.010**	0.052	0.277	0.231
		$h^{\text{exc-burd}}$	0.420	0.978	0.017	0.077	**0.008**	**0.009**	0.014	0.057	0.176	0.110
		$h^{\text{bal-exc-burd}}$	0.426	0.943	0.023	0.078	0.010	0.012	0.016 †	**0.029***	0.128	0.087
		$h^{\text{all-exc-burd}}$	0.420	0.966	0.019	0.081	0.010	0.012	0.017	**0.036***	0.148	0.085
		$h^{\text{burd}}_{\text{FN>TN}}$	0.421	**0.998**	**0.003**	**0.048**	0.013	0.014	0.014	0.078	0.362	0.299
		$h^{*}_{\text{FN>TN}}$	0.431	**0.994**	0.007	**0.054**	**0.008**	**0.008**	0.016	0.040	0.170	0.120
		h^{*}	0.429	0.993	**0.005**	**0.055**	**0.007**	**0.007**	**0.013**	**0.034***	0.143	0.113

This table continues onto the next page.

Data Setting	Model	Algorithmic Decision-Maker	Balanced 0/1 Loss	Coverage	$\lvert\Delta_a\rvert$ for Coverage	$\mathbb{E}$(Cost)	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) ($\alpha^{\text{eq-cost}}$)	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) $\forall\, y_i = 1$	$\lvert\Delta_a\rvert$ for $\mathbb{E}$(Cost) $\forall\, y_i = 0$	$\alpha^{\text{ind-fair}}$	$\lvert\Delta_a\rvert$ for FPR	$\lvert\Delta_a\rvert$ for FNR
German Credit Data	MLP (1x4)	$h^{0/1}$	0.409	0.978	0.010	0.083	0.014	0.014	0.021	0.134	0.377	0.359
		$h^{\text{eq-cost}}$	0.408	0.964	0.029	0.077	0.011 †	0.011 †	0.017 †	0.107 ‡	0.344	0.307
		$h^{\text{bal-err}}$	0.427	0.877	0.046	0.083	0.015	0.016	0.024	0.058	0.070	0.064
		$h^{\text{bal+eq}}$	0.426	0.909	0.040	0.084	0.013	0.014	0.021	0.062	0.091	0.058
		h^{burd}	0.426	**0.998**	0.003	**0.045**	0.009	0.010	**0.013**	0.067	0.320	0.261
		$h^{\text{exc-burd}}$	0.418	0.958	0.019	0.078	**0.009**	0.011	0.016	0.055	0.188	0.135
		$h^{\text{bal-exc-burd}}$	0.427	0.969	0.017	0.080	**0.008**	0.010	0.016	**0.035***	0.133	0.079
		$h^{\text{all-exc-burd}}$	0.428	0.960	0.013	0.078	0.010	0.010	0.017	**0.029***	0.140	0.088
		$h^{\text{burd}}_{\text{FN>TN}}$	0.424	**0.997**	0.005	**0.048**	0.010	0.011	0.013	0.081	0.347	0.268
		$h^{*}_{\text{FN>TN}}$	0.428	0.991	0.009	**0.052**	**0.007**	**0.007**	**0.013**	**0.038***	0.153	0.117
		h^{*}	0.427	0.993	0.005	**0.054**	**0.007**	**0.008**	0.013	**0.029***	0.170	0.103
First-Year Law School Success	lr	$h^{0/1}$	0.417	0.734	0.529	0.196	0.288 †	0.267 †	0.285 †	––– ‡	0.635	0.678
		$h^{\text{eq-cost}}$	0.419	0.684	0.898 †	0.145 †	0.311 †	0.066 †	0.301 †	0.219 ‡	0.669	0.746
		$h^{\text{bal-err}}$	0.450	0.718	0.123	0.215 †	0.099 †	0.106 †	0.093 †	0.156 ‡	0.076	0.116
		$h^{\text{bal+eq}}$	0.449	0.886	0.098	0.189 †	0.053 †	0.091 †	0.045 †	0.139 ‡	0.092	0.121
		h^{burd}	0.416	**0.978**	0.079	0.168	0.238 †	0.217 †	0.233 †	0.362 ‡	0.673	0.602
		$h^{\text{exc-burd}}$	0.426	**0.976**	0.088	0.154	0.187 †	0.175 †	0.182 †	0.358 ‡	0.595	0.514
		$h^{\text{bal-exc-burd}}$	0.451	0.795	0.102	0.193	0.069	0.112 †	0.061	0.151 ‡	0.068	0.100
		$h^{\text{all-exc-burd}}$	**0.460**	**0.961**	0.043	0.135	0.036	0.049 †	0.039	0.153 ‡	0.084	0.118
		$h^{\text{burd}}_{\text{FN>TN}}$	0.414	**0.949**	0.132	0.191	0.293 †	0.264 †	0.287 †	0.557 ‡	0.746	0.682
		$h^{*}_{\text{FN>TN}}$	0.450	**1.000**	**0.000**	**0.115**	**0.024**	0.035 †	**0.024**	0.140 ‡	0.140	0.142
		h^{*}	0.448	**1.000**	**0.000** †	**0.120**	**0.022** †	0.042 †	**0.021** †	0.141 ‡	0.143	0.138
	MLP (1x2)	$h^{0/1}$	0.414	0.766	0.507	0.194	0.298 †	0.239 †	0.301 †	0.600 ‡	0.656	0.707
		$h^{\text{eq-cost}}$	0.419	0.681	0.931	0.149	0.343 †	0.120 †	0.318 †	––– ‡	0.664	0.741
		$h^{\text{bal-err}}$	0.454	0.661	0.097	0.215 †	0.071 †	0.112 †	0.059 †	0.133 ‡	0.070	0.099
		$h^{\text{bal+eq}}$	0.447	0.848 †	0.062 †	0.180 †	0.051 †	0.089 †	0.042 †	0.145 ‡	0.080	0.130
		h^{burd}	0.417	**0.985**	0.042	0.166	0.215 †	0.218 †	0.206 †	0.369 ‡	0.654	0.552
		$h^{\text{exc-burd}}$	0.423	**0.971**	0.112	0.151	0.180 †	0.166 †	0.177 †	0.368 ‡	0.618	0.519
		$h^{\text{bal-exc-burd}}$	0.449	0.783	0.080	0.192	0.065 †	0.092 †	0.057 †	0.173 ‡	0.059	0.121
		$h^{\text{all-exc-burd}}$	0.461	**0.995**	**0.005**	**0.125**	**0.017**	**0.043*** †	**0.021**	0.153 ‡	0.080	0.123
		$h^{\text{burd}}_{\text{FN>TN}}$	0.415	**0.973**	0.079	0.181	0.260 †	0.237 †	0.256 †	0.536 ‡	0.731	0.684
		$h^{*}_{\text{FN>TN}}$	0.450	**0.999**	**0.001**	**0.113**	**0.022**	**0.039*** †	**0.020**	0.136 ‡	0.158	0.150
		h^{*}	0.450	**1.000**	**0.000**	**0.119**	**0.018**	**0.033*** †	**0.017**	0.140 ‡	0.124	0.137
	MLP (1x4)	$h^{0/1}$	0.415	0.786	0.507	0.187	0.297 †	0.233 †	0.306 †	0.452 ‡	0.650	0.696
		$h^{\text{eq-cost}}$	0.420	0.670	0.878 †	0.148 †	0.244 †	0.283 †	0.302 †	––– ‡	0.646	0.711
		$h^{\text{bal-err}}$	0.447	0.787	0.104	0.203	0.075 †	0.107 †	0.065 †	0.200 ‡	0.087	0.106
		$h^{\text{bal+eq}}$	0.445	0.883	0.095	0.185 †	0.054 †	0.082 †	0.049 †	0.119 ‡	0.107	0.125
		h^{burd}	0.417	**0.987**	**0.039**	0.166	0.220	0.216 †	0.211	0.322 ‡	0.658	0.545
		$h^{\text{exc-burd}}$	0.424	**0.985**	0.051	0.151	0.179 †	0.172 †	0.175 †	0.345 ‡	0.615	0.515
		$h^{\text{bal-exc-burd}}$	0.449	0.851	0.080	0.185	0.055	0.080 †	0.052	0.163 ‡	**0.057**	0.119
		$h^{\text{all-exc-burd}}$	**0.459**	**0.977**	**0.016**	**0.130**	**0.032**	**0.051** †	0.033	0.152 ‡	0.089	0.122
		$h^{\text{burd}}_{\text{FN>TN}}$	0.417	**0.966**	0.095	0.186	0.259 †	0.248 †	0.251 †	0.386 ‡	0.744	0.657
		$h^{*}_{\text{FN>TN}}$	0.448	**1.000**	**0.000**	**0.110**	**0.025**	**0.042**	**0.023**	0.131 ‡	0.158	0.156
		h^{*}	0.446	**1.000**	**0.000** †	**0.120**	**0.021** †	**0.031** †	**0.020** †	0.148 ‡	0.144	0.143

Table 9: Results are shown for all classifiers: logistic regression (lr); multi-layer perceptron with 1 layer and 2 hidden units (MLP (1x2)); and multi-layer perceptron with 1 layer and 4 hidden units, (MLP (1x4)). For readability, we neglect the θ subscript and $(\cdot)$ notation. All weight settings are displayed in Table 7. Settings for β_{FP}, β_{FN}, and ϵ_{corr} differ per data setting and appear in Table 6. Each value is averaged over 50 runs, each with a unique seed for a 70-30 train-test split. Classifiers are fit on training data, and results in the table reflect test data metrics. Balanced 0/1 is $\frac{1}{2}FPR + \frac{1}{2}FNR$. Coverage is the proportion of individuals with unfavorable outcomes who received a recommendation. $\mathbb{E}$(Cost) is average cost for those with unfavorable outcomes and coverage. $|\Delta_a|$ is the absolute difference in the relevant metric across sensitive subpopulations, stratified by positive class ($y_i = 1$) and negative class ($y_i = 0$), where applicable. $\alpha^{\text{ind-fair}}$ is defined in Equation 6. Bolded values indicate statistically significant improvements ($p < 0.05$, one-tailed t-test) over *all* the benchmark methods, except in the 'Balanced 0/1 Loss' column, where bold represents significant increases (i.e., decreases in predictive accuracy). † represents missingness (e.g. no individuals with recommendations for some runs) . ‡ represents no values found for $\alpha^{\text{ind-fair}}$ for some runs. * represents statistical significance was evaluated against all benchmarks that were not missing but not the full set of benchmarks. – represents for all runs $\alpha^{\text{ind-fair}}$ could not be calculated (insufficient coverage for x_i or x_i^{CF}).