Feature Importance Disparities for Data Bias Investigations

Peter W Chang [1] **Leor Fishman** [2] **Seth Neel** [1]

Abstract

It is widely held that one cause of downstream bias in classifiers is bias present in the training data. Rectifying such biases may involve context-dependent interventions such as training separate models on subgroups, removing features with bias in the collection process, or even conducting real-world experiments to ascertain sources of bias. Despite the need for such data bias investigations, few automated methods exist to assist practitioners in these efforts. In this paper, we present one such method that given a dataset X consisting of protected and unprotected features, outcomes y, and a regressor h that predicts y given X, outputs a tuple (f_j, g), with the following property: g corresponds to a subset of the training dataset (X, y), such that the jth feature f_j has much larger (or smaller) *influence* in the subgroup g, than on the dataset overall, which we call *feature importance disparity* (`FID`). We show across 4 datasets and 4 common feature importance methods of broad interest to the machine learning community that we can efficiently find subgroups with large `FID` values even over exponentially large subgroup classes and in practice these groups correspond to subgroups with potentially serious bias issues as measured by standard fairness metrics.

1. Introduction

Machine learning is rapidly becoming a more important, yet more opaque part of our lives and decision making – with increasingly high stakes use cases such as recidivism analysis (Angwin et al., 2016), loan granting and terms (Dastile et al., 2020) and child protective services (Keddell, 2019). One of the hopes of wide-scale ML deployment has been that those algorithms might be free of our human biases and imperfections. This hope was, unfortunately, naive. Over the last decade, an interdisciplinary body of research has shown that machine learning algorithms can be deeply biased in both subtle and direct ways (Barocas et al., 2019), and has focused on developing countless techniques to produce fairer models (Caton & Haas, 2023).

One of the primary causes of model bias is bias inherent in the training data, rather than an explicitly biased training procedure. While the majority of work on fairness seeks to remove bias by learning a fairer representation of the data (Zemel et al., 2013) or by explicitly constraining the downstream classifier to conform to a specific fairness notion (Hardt et al., 2016a; Agarwal et al., 2018b), fairness notions have been shown to be brittle and often times contradictory (Dwork & Ilvento, 2018; Kleinberg et al., 2016). More importantly, these approaches elide what could be a more important question for the practitioner: *What is the source of bias in the training data, and what subgroups in the data are being effected?*

We call the process of answering this question a *data bias investigation* (`DBI`), and in this paper we develop a technique to aid in a `DBI` by allowing an analyst to (efficiently and provably) identify structured subsets of the training data to focus their bias investigation. Prior work has typically focused on identifying such subsets by finding subsets of the training data that maximally violate a specific fairness criterion, which typically corresponds to the classifier having a higher error rate on the group than on the population. We take a very different approach; rather than optimizing for a specific fairness notion, we find subgroups where *a specific feature in the data has out-sized impact in the subgroup, relative to the population as a whole.*

To build some intuition for why feature importance disparities, `FID` as we call them, might be a useful notion when looking for dataset bias, consider the following example from Cynthia Dwork, widely regarded as pioneer in the field of algorithmic bias, in a 2015 in a New York Times interview (Miller, 2015): *Suppose we have a minority group in which bright students are steered toward studying math, and suppose that in the majority group bright students are steered instead toward finance. An easy way to find good students is to look for students studying finance, and if the minority is small, this simple classification scheme could find most of the bright students. But not only is it unfair*

[1]SAFR AI Lab, Harvard Business School & Kempner Institute, Boston, MA [2]Harvard College, Cambridge, MA. Correspondence to: Seth Neel <sethneel.ai@gmail.com>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

to the bright students in the minority group, it is also low utility.

Unpacking this example further, the feature `is-finance-major` is predictive in finding bright students in the population at large but not in the minority group. Meanwhile, the feature `is-math-major` is highly predictive in the minority group but not at all in the majority group. The classifier that only selects students who study finance is unfair to the minority group, exactly because of the differing importance of these features in the two groups. Once the subgroups have been identified, the actions the analyst then takes is then entirely context-dependent. This could include "easy" fixes like training a separate model on the subgroup found or excluding a specific feature from training, or more complex remedies like investigating how the specific feature or the outcome variable was collected in the group, and if there is bias in that collection process.

While simple models like decision trees or linear regression come equipped with intuitive notions of feature importance, in general there is no definitive notion of feature importance for complex models. Approaches that have garnered substantial attention include local model agnostic methods like LIME (Ribeiro et al., 2016), SHAP (Lundberg & Lee, 2017), model-specific saliency maps (Simonyan et al., 2013), and example-based counterfactual explanations (Molnar, 2022). Concerns about the stability and robustness of the most widely used feature importance notions, including the ones we study, have been raised (Dai et al., 2022; Agarwal et al., 2022a; Alvarez-Melis & Jaakkola, 2018; Bansal et al., 2020; Dimanov et al., 2020; Slack et al., 2020) and these notions are often at odds with each other, so none can be considered definitive (Krishna et al., 2022). Regardless of these limitations, these notions are used widely in practice today, and are still useful as a diagnostic tool as we eventually propose.

Fixing any of these notions of feature importance, given a small set of protected subgroups, it would be simple to iterate through the subgroups and features, compute the `FID` for each feature with respect to each subgroup, and then select the feature and subgroup that shows the largest disparity. However, it is also known in the fairness literature that while a classifier may look fair when comparing a given fairness metric across a handful of sensitive subgroups, when the notion of a sensitive subgroup is generalized to encompass combinations and interactions between sensitive features (known as *rich subgroups* (Kearns et al., 2019)), large disparities can emerge. We verify that this phenomenon of rich subgroups uncovering much larger disparities than marginal subgroups alone also holds for feature importance in Subsection 5.4. Even for simple definitions of rich subgroup such as conjunctions of binary features, the number of subgroups is exponential in the number of sensitive attributes, and so it is infeasible to compute the metric on each sensitive subgroup and find the largest value by brute force. This raises an obvious question in light of the prior discussion, although one that to the best of our knowledge has not been thoroughly studied: *When applied to classifiers and datasets where bias is a concern, do these feature importance notions uncover substantial differences in feature importance across rich subgroups, and can they be efficiently detected?*

1.1. Case Study: Data Bias Investigation on COMPAS

Before diving into the technical details of our method, we illustrate how our technique can yield interesting potential interventions when training a random forest classifier h to predict `two-year-recidivism` on the `COMPAS` dataset (Angwin et al., 2016). Using the feature importance method SHAP (Lundberg & Lee, 2017) and our Algorithm 1, we find that the feature `priors-count` is substantially more important when predicting `two-year-recidivism` on a subgroup largely defined by Native-American and African-American males that makes up $\sim 9\%$ of the training set (Figure 1). While we further discuss in Appendix A that disparities in feature importance do not guarantee that a subgroup has a fairness disparity, we show in Subsection 5.5 that empirically this is often the case. In this example, we find that conventional fairness metrics are slightly worse on this subgroup relative to the population, with $\sim 1\%$ lower accuracy and $\sim 1\%$ higher false positive rate (FPR). One simple solution to increase accuracy on the subgroup and potentially reduce the disparity is to train a separate model h_g for the subgroup, which we find drops the FPR in the subgroup by 7.5% and increases accuracy by 7%. Furthermore, the disparity between the average SHAP value for the feature `priors-count` using h_g, and the average SHAP value over whole population using h is more than halved. Another intuitive technical solution is to train a new model, h_{-f}, without the feature `priors-count`. With this solution, there is a moderate decrease in fairness disparity, but it also comes with a noticeable drop in model performance, likely due to removing the predictive power of the feature `priors-count`. On a qualitative level, identifying this subgroup could also motivate further research into how Native-American and African-American males in the `COMPAS` dataset are policed differently, possibly resulting in measurement biases in the `priors-count` feature or the `two-year-recidivism` outcome.

1.2. Results

While the prior example clearly illustrates the utility of our method at a high level, the devil is in the details, and in the rest of the paper, we formalize the notions of feature importance and protected rich subgroups along with our methods for efficiently detecting potentially biased subgroups.

Our most important contribution is introducing the notion

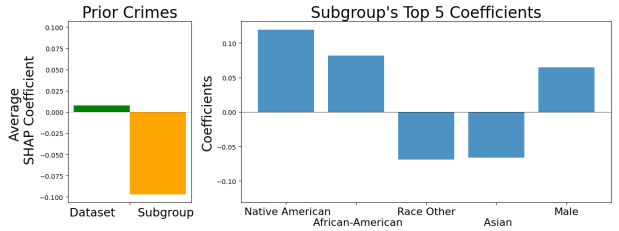


Figure 1: Exploring a high FID subgroup/feature pair for COMPAS. The first graph compares the average SHAP feature importance for priors−count in the subgroup vs. the dataset as a whole. The second graph shows the 5 largest coefficients of the linear function of sensitive attributes that define the subgroup.

of *feature importance disparity* in the context of recently developed feature importance notions, and with respect to rich subgroups (Definition 3.1). We categorize a feature importance notion as *separable* or not, based on whether it can be expressed as a sum over points in the subgroup (Definition 3.2) and define a variant of FID, the *average feature importance disparity* (AVG−FID, Definition 3.3). Our main theoretical contribution is Theorem 4.1 in Section 4, which says informally that although the problem of finding the maximal FID subgroup is NP-hard in the worst case (Appendix G), given access to an oracle for *cost-sensitive classification* with respect to the rich subgroup class $\mathcal{G}$, (Definition F.1), Algorithm 1 efficiently learns the subgroup with maximal FID for any separable feature importance notion.

In Section 5, we conduct a thorough empirical evaluation, auditing for large FID subgroups on the Student (Cortez & Silva, 2008), COMPAS (Angwin et al., 2016), Bank (Moro et al., 2014), and Folktables (Ding et al., 2021) datasets, using LIME, SHAP, saliency maps, and linear regression coefficient as feature importance notions. Our experiments establish the following: (i) Across all (dataset, importance notion) pairs, we can find subgroups defined as functions of sensitive features that have large FID with respect to a given feature (Table 1, Figures 2, 3). (ii) Inspecting the coefficients of these subgroups yields interesting discussion about potential dataset bias (Figure 1, Section 5.3). (iii) In about half the cases, rich subgroups yield higher out of sample FID compared to only searching subgroups defined by a single sensitive attribute, justifying the use of rich subgroups (Section 5.4). (iv) These subgroups have disparities in accepted fairness metrics such as demographic parity and calibration (Table 2). Conversely, rich subgroups that maximally violate fairness metrics also express large FID values (Table 3)

These results generalize out of sample, both in terms of the FID values found, and the sizes of the corresponding subgroups found (Appendix L). Taken together, these theoretical and empirical results highlight our methods as an important addition to the diagnostic toolkit for DBI in tabular datasets with sensitive features.

2. Related Work

There is substantial work investigating bias in the context of machine learning models and their training data (Barocas et al., 2019; Caton & Haas, 2023). We are motivated at a high level by existing work on dataset bias (Kamiran & Calders, 2012; Tommasi et al., 2017; Li & Vasconcelos, 2019), however, to the best of our knowledge, this is the first work investigating the disparity in feature importance values in the context of rich subgroups as a fairness diagnostic. For more related work, see Appendix C.

Anomalous Subgroup Discovery. In terms of approach, two closely related works are (Dai et al., 2022) and (Balagopalan et al., 2022) which link fairness concerns on sensitive subgroups with model explanation quality, as measured by properties like stability and fidelity. Our work differs in that we are focused on the magnitude of explanation disparities themselves rather than their "quality," and that we extend our results to the rich subgroup setting. Our algorithm for searching an exponentially large subgroup space is a novel and necessary addition to work in this space. Another area of research looks to prove that a chosen score function satisfies the linear time subset scanning property (Neill, 2012) which can then be leveraged to search the subgroup space for classifier bias (Zhang & Neill, 2016; Boxer et al., 2023) in linear time. While it is hard to say with absolute certainty that this approach would not be useful it is not immediately apparent how we would force a subset scanning method to optimize over *rich subgroups*.

Rich Subgroups and Multicalibration. At a technical level, the most closely related papers are (Kearns et al., 2018; Hebert-Johnson et al., 2018) which introduce the notion of the rich subgroup class $\mathcal{G}$ over sensitive features in the context of learning classifiers that are with respect to equalized odds or calibration. Our Algorithm 1 fits into the paradigm of "oracle-efficient" algorithms for solving constrained optimization problems introduced in (Agarwal et al., 2018a) and developed in the context of rich subgroups in (Kearns et al., 2018; 2019; Hebert-Johnson et al., 2018). There has been much recent interest in learning multicalibrated predictors because of connections to uncertainty estimation and omnipredictors (Hu et al., 2023; Gopalan et al., 2022; Jung et al., 2021). None of these works consider feature importance disparities.

Feature Importance Notions. For the field of interpretable or explainable machine learning, we refer to the survey by (Molnar, 2022). The most relevant works cover methods used to investigate the importance of a feature in a given

subset of the dataset. Local explanation methods assign a feature importance for every point (x, y) and define a notion of importance in a subgroup by summing or averaging over the points in the subgroup as we do in Definitions 3.2, 3.3.

3. Preliminaries

Let X^n represent our dataset, consisting of n individuals defined by the tuple $((x, x'), y)$ where $x \in \mathcal{X}_{sense}$ is the vector of protected features, $x' \in \mathcal{X}_{safe}$ is the vector of unprotected features, and $y \in \mathcal{Y}$ denotes the label. With $X = (x, x') \in \mathcal{X} = \mathcal{X}_{sense} \times \mathcal{X}_{safe} \subset \mathbb{R}^d$ denoting a joint feature, the data points (X, y) are drawn i.i.d. from a distribution $\mathcal{R}$. Let $h : \mathcal{X} \to \mathcal{Y}$ denote a classifier or regressor that predicts y from X. We define a *rich subgroup class* $\mathcal{G} = \{g_\alpha\}_{\alpha \in \Omega}$ as a collection of functions $g : \mathcal{X}_{sens} \to [0, 1]$, where $g(x')$ denotes the membership of point $X = (x, x')$ in group g. Note that this is the same subgroup definition as in (Kearns et al., 2018), but without the constraint that $g(x') \in \{0, 1\}$, which supports varying degrees of group membership. E.g. a biracial person may be .5 a member of two racial groups. Let $f_j, j \in [d]$ denote the jth feature in $\mathcal{X} \subset \mathbb{R}^d$. Then for a classifier h and subgroup $g \in \mathcal{G}$, let F be a feature importance notion where $F(f_j, g(X^n), h)$ denote the *importance h attributes to feature j in the subgroup* $g(X^n)$, and $F(f_j, X^n, h)$ be the importance h attributes to f_j on the entire dataset. We will provide more specific instantiations of F shortly, but we state our definition of FID in the greatest possible generality below.

Definition 3.1. (Feature Importance Disparity). Given a classifier h, a subgroup of X^n defined by $g \in G$, and a feature $f_j \in [d]$, then given a feature importance notion $F(\cdot)$, the feature importance disparity relative to g is defined as:

$$\texttt{FID}(f_j, g, h) = \mathbb{E}_{X \sim \mathcal{R}} |F(f_j, g(X^n), h) - F(f_j, X^n, h)|$$

We will suppress h and write $\texttt{FID}(j, g)$ unless necessary to clarify the classifier we are using. Now, given h and X^n, our goal is to find the feature subgroup pair $(j^*, g^*) \in [d] \times \mathcal{G}$ that maximizes $\texttt{FID}(j, g)$, or $(j^*, g^*) = \text{argmax}_{g \in \mathcal{G}, j \in [d]} \texttt{FID}(j, g)$.

We now get more concrete about our feature importance notion $F(\cdot)$. First, we define the class of *separable* feature importance notions:

Definition 3.2. (Locally Separable). A feature importance notion $F(\cdot)$ is locally separable if it can be decomposed as a point wise sum of local model explanation values F':

$$F(f_j, X^n, h) = \sum_{X \in X^n} F'(f_j, X, h)$$

It follows that for separable notions, $F(f_j, g(X^n), h) = \sum_{X \in X^n} g(X) F'(f_j, X, h)$. Given a local model explanation F', we can define a more specific form of FID, the *average feature importance disparity* (AVG-FID), which compares the average feature importance within a subgroup to the average importance on the dataset.

Definition 3.3. (Average Case Locally Separable FID). For a $g \in \mathcal{G}$, let $|g| = \sum_{X \in X^n} g(X)$. Given a local model explanation $F'(\cdot)$, we define the corresponding:

$$\begin{aligned}\texttt{AVG-FID}(f_j, g, h) = &\mathbb{E}_{X^n \sim \mathcal{R}^n} \Big| \frac{1}{|g|} \sum_{X \in X^n} g(X) F'(f_j, X, h) \\ &- \frac{1}{n} \sum_{X \in X^n} F'(f_j, X, h) \Big|\end{aligned}$$

Note that AVG-FID is not equivalent to a separable FID, since we divide by $|g|$, impacting every term in the summation. In Section 4, we show that we can optimize for AVG-FID by optimizing a version of the FID problem with size constraints, which we can do efficiently via Algorithm 1.

This notion of *separability* is crucial to understanding the remainder of the paper. In Section 4, we show that for any *separable* FID, Algorithm 1 is an (oracle) efficient way to compute the largest FID subgroup of a specified size in polynomial time. By "oracle efficient," we follow (Agarwal et al., 2018a; Kearns et al., 2018) where we mean access to an optimization oracle that can solve (possibly NP-hard) problems. While this sounds like a strong assumption, in practice we can take advantage of modern optimization algorithms that can solve hard non-convex optimization problems (e.g. training neural networks). This framework has led to the development of many practical algorithms with a strong theoretical grounding (Agarwal et al., 2018a; Kearns et al., 2018; 2019; Hebert-Johnson et al., 2018), and as shown in Section 5 works well in practice here as well. The type of oracle we need is called a Cost Sensitive Classification (CSC) oracle, which we define in Appendix F.

4. Optimizing for AVG-FID

In this section, we show how to (oracle) efficiently compute the rich subgroup that maximizes the AVG-FID. Rather than optimize AVG-FID directly, our Algorithm 1 solves an optimization problem that maximizes the FID subject to a group size constraint:

$$\begin{aligned} &\max_{g \in \mathcal{G}} |F(f_j, g(X^n), h) - F(f_j, X^n, h)| \\ \text{s.t.} \quad &\Phi_L(g) \equiv \alpha_L - \frac{1}{n} \sum_{X \in X^n} g(X) \leq 0, \\ &\Phi_U(g) \equiv \frac{1}{n} \sum_{X \in X^n} g(X) - \alpha_U \leq 0 \end{aligned} \tag{1}$$

where Φ_L and Φ_U are "size violation" functions given a subgroup function g. We denote the optimal solution to Equation 1 by $g^*_{[\alpha_L, \alpha_U]}$. We focus on optimizing the constrained FID since the following primitive also allows us to efficiently optimize AVG-FID:

1. Discretize $[0, 1]$ into intervals $(\frac{i-1}{n}, \frac{i}{n}]_{i=1}^n$. Given feature f_j, compute $g^*_{(\frac{i-1}{n}, \frac{i}{n}]}$ for $i = 1...n$.

2. Outputting g_{k^*}, where $k^* = \text{argmax}_k \frac{k}{n} |F(f_j, g_k, h)|$ approximately maximizes the AVG-FID given an appropriately large number of intervals n.

Our proof for this is available in Appendix E. We now state our main theorem, which shows that we can solve the constrained FID problem in Equation 1 with polynomially many calls to $\text{CSC}_{\mathcal{G}}$.

Theorem 4.1. *Let F be a separable FID notion, fix a classifier h, subgroup class $\mathcal{G}$, and oracle $\text{CSC}_{\mathcal{G}}$. Then choosing accuracy constant ν and bound constant B and fixing a feature of interest f_j, we will run Algorithm 1 twice; once with FID given by F, and once with FID given by $-F$. Let $\hat{p}^T_{\mathcal{G}}$ be the distribution returned after $T = O(\frac{4n^2B^2}{\nu^2})$ iterations by Algorithm 1 that achieves the larger value of $\mathbb{E}[FID(j, g)]$. Then:*

$$\begin{aligned} FID(j, g^*_j) - \mathbb{E}_{g \sim \hat{p}^T_{\mathcal{G}}}[FID(j, g)] &\leq \nu \\ |\Phi_L(g)|, |\Phi_U(g)| &\leq \frac{1 + 2\nu}{B} \end{aligned} \tag{2}$$

We defer the proof of Theorem 4.1 to Appendix D. In summary, rather than optimizing over $g \in \mathcal{G}$, we optimize over distributions $\Delta(\mathcal{G})$. This allows us to cast the optimization problem in Equation 1 as a linear program so we can form the Lagrangian L, which is the sum of the feature importance values and the size constraint functions weighted by the dual variables λ, and apply strong duality. We can then cast the constrained optimization as computing the Nash equilibrium of a two-player zero-sum game, and apply the classical result of (Freund & Schapire, 1996) which says that if both players implement *no-regret* strategies, then we converge to the Nash equilibrium at a rate given by the average regret of both players converging to zero. Algorithm 1 implements the no-regret algorithm exponentiated gradient descent (Kivinen & Warmuth, 1997) for the max player, who optimizes λ, and best-response via a CSC solve for the min player, who aims to maximize subgroup disparity to optimize the rich subgroup distribution.

We note that rather than computing the group g that maximizes $\text{FID}(j, g)$ subject to the size constraint, our algorithm outputs a distribution over groups $\hat{p}^T_{\mathcal{G}}$ that satisfies this process *on average* over the groups. In theory, this seems like a drawback for interpretability. However, in practice we simply take the groups g_t found at each round and output the ones that are in the appropriate size range, and have largest FID values. The results in Section 5 validate that this heuristic choice is able to find groups that are both feasible and have large FID values. This method also generalizes out of sample showing that the FID is not artificially inflated by multiple testing (Appendix L). Moreover, our method provides a menu of potential groups $(g_t)_{t=1}^T$ that can be quickly evaluated for large FID, which can be a useful feature to find interesting biases not present in the maximal subgroup.

5. Experiments

Here we report the results of our empirical investigation across 16 different dataset/FID-notion pairings. These results confirm that our method can find large AVG-FID values corresponding to rich subgroups defined as simple functions of protected attributes (Table 1, Figures 2, 3), and are larger than those found by optimizing over marginal subgroups alone (Section 5.4). Moreover, in Section 5.5 we find that high AVG-FID subgroups tend to have significant disparities in traditional fairness metrics (Table 2), and that rich subgroups that maximize a fairness notion like FPR disparity also express high AVG-FID features, albeit smaller than those found by Algorithm 1 (Table 3). Perhaps most significantly, for a tool designed to assist the process of DBI, is that the high AVG-FID subgroups found correspond to subgroups and features that are suggestive of potential dataset bias, one example of which we covered in the initial case study and provide further examples of in Section 5.3. In Appendix J, we construct two synthetic datasets, one where the feature importance is the same across all subgroups, and one where there is a deliberately introduced disparity in a specific rich subgroup, in order to verify that our methods i) avoid false discovery and ii) can effectively pick out a high-FID subgroup. We also include results showing that the rich subgroups found generalize out of sample both in terms of AVG-FID and group size $|g|$ (Appendix L), are relatively robust to the choice of hypothesis class of h (Appendix M), and that our algorithms converge quickly in practice (Appendix O). The code used for our experiments is available at github.com/safr-ai-lab/xai-disparity.

5.1. Experimental Details

Datasets: We used four popular datasets for the experiments: Student, COMPAS, Bank, and Folktables. For each test, we used COMPAS twice, once predicting two-year recidivism and once predicting decile risk score (labeled COMPAS R and COMPAS D respectively). For each dataset, we specified "sensitive" features which are features generally covered by equal protection or privacy laws (e.g. race, gender, age, health data). Appendix I contains more details.

Computing the AVG-FID: We study 3 separable notions of FID based on local model explanations Local-Interpretable, Model-Agnostic (LIME) (Ribeiro et al., 2016), Shapley Additive Explanations (SHAP) (Lundberg & Lee, 2017), and saliency maps (GRAD) (Simonyan et al., 2013). For every method and dataset, we optimize the constrained FID over α ranges $(\alpha_L, \alpha_U) = \{[.01, .05], [.05, .1], [.1, .15], [.15, .2], [.2, .25]\}$. These small ranges allowed us to reasonably compare the FID values, reported in Table 1. Additionally, these ranges span subgroup sizes that may be of particular interest in fairness research and dataset auditing work. All values of AVG-FID reported in the results are *out of sample*; i.e. the AVG-FID values are computed on a test set that was not used to optimize the subgroups. Datasets were split into $80 - 20$ train-test split except for Student which was split $50 - 50$ due to its small size. Across all datasets, when the FID was LIME or SHAP, we set h to be a random forest, when it was GRAD we used logistic regression as it requires a classifier whose outputs are differentiable in the inputs. The exact choice of classifier does not have any notable impact on the outcomes as we discuss in Appendix M. Due to computation constraints, GRAD was only tested on the COMPAS R dataset. We defer the details in how we implemented the importance notions and Algorithm 1 to Appendix I.

Linear Feature Importance Disparity: In addition to the 3 separable notions of FID, we also studied an approach for a non-separable notion of importance. Linear regression (labeled LR in results) is a popular model that is inherently interpretable; the coefficients of a weighted least squares (WLS) solution represent the importance of each feature. We can thus define another variant of FID, the linear feature importance disparity (LIN-FID), as the difference in the WLS coefficient of feature f_j on subgroup g and on the dataset X^n. As LIN-FID is differentiable with respect to g, we are able to find a locally optimal g with high LIN-FID using a non-convex optimizer; we used ADAM. For details and proofs, see Appendix H.

5.2. Experimental Results

Table 1 summarizes the results of the experiments, which are visualized in Figure 2 on a log-ratio scale for better cross-notion comparison. Across each dataset and importance notion, our methods were able to find subgroups with high FID, often differing by orders of magnitude. For example, on Folktables with LIME as the importance notion, there is a subgroup on which age is on average 225 times more important than it is for the whole population. Table 1 also provides the defining features, listed as the sensitive features which have the largest coefficients in g.

A natural follow up question that arises from this experiment is what does the distribution of FIDs look like for a given dataset? Figure 3 shows a distribution of the 10 features on the Bank dataset with the highest FID values. As we can see, there are a few features where large FID subgroups can be found, but it tails off significantly. This pattern is replicated across all datasets and feature importance notions. This is a positive result for practical uses, as an analyst or domain expert can focus on a handful of features that perform drastically differently when performing a DBI.

Earlier in Section 1.1, we examined a specific case where the FID found as a result of our method revealed a biased subgroup where the fairness disparities could then be mitigated with targeted approaches. In the next section, across every dataset and feature importance notion, we find similar examples exposing some form of potential bias. We note that not every single (subgroup, feature) pair discovered necessarily implies a fairness concern. For example, AVG-FID could be driven in part by correlations between f_j and the sensitive attributes that define g. Since it remains true in all fairness work that two contexts that are statistically equivalent may have very different fairness implications in the real world, our method should be viewed as a tool to aid practitioners in DBI rather than as definitive proof of bias.

5.3. Discussion of High FID Subgroups

In Figure 4, we highlight selections of an interesting (feature, subgroup, method) pair for each dataset. Figure 4a shows that on the Student dataset the feature absences which is of near zero importance on the dataset as a whole, is very negatively correlated with student performance on a subgroup whose top 2 features indicate whether a student's parents are together, and if they live in an rural neighborhood. Figure 4b shows that on the COMPAS dataset with method GRAD, the feature arrested-but-with-no-charges is typically highly important when predicting two-year-recidivism. However, it carries significantly less importance on a subgroup that is largely defined as Native American males. When predicting the decile risk score on COMPAS, LIME indicates that age is not important on the dataset as a whole; however, for non-Native American, female minorities, older age can be used to explain a lower Decile Score. On the Bank dataset using LIN-FID, we see that a linear regression

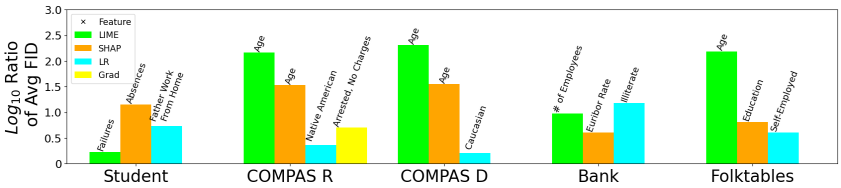


Figure 2: Summary of the highest FIDs found for each (dataset, method). This is displayed as $\left|log_{10}(R)\right|$ where R is the ratio of average importance per data point in g^* to the average importance on X for separable notions, or the ratio of coefficients for LIN-FID. This scale allows comparison across different importance notions. The feature associated with each g^* is written above the bar.

Table 1: Summary of the subgroup with highest AVG-FID for each experiment along with the corresponding feature, subgroup size, and defining features. Experiments were run across multiple (α_L, α_U) ranges with the highest AVG-FID found being displayed. $\mu(F)$ is the average feature importance value on the specified group.

Dataset	Notion	Feature f_j	$\mu(F(f_j, X))$	$\mu(F(f_j, g))$	$\lvert g \rvert$	Defining Features
Student	LIME	Failures	$-.006$	$-.011$	.01	Alcohol Use, Urban Home
	SHAP	Absences	$-.15$	-2.1	.02	Parent Status, Urban Home
	LR	Father WFH	21.7	-4.0	.03	Alcohol Use, Health
COMPAS R	LIME	Age	.0009	$-.14$	.05	Native-American
	SHAP	Age	.012	.41	.04	Asian-American
	LR	Native American	.5	1.17	.04	Asian/Hispanic-American
	GRAD	Arrest, No Charge	.09	.02	.05	Native-American
COMPAS D	LIME	Age	$-.0003$	$-.06$	.02	Native/Black-American
	SHAP	Age	.06	2.35	.07	Black/Asian-American
	LR	Caucasian	6.7	10.7	.04	Native-American
Bank	LIME	# of Employees	$-.003$	.03	.03	Marital Status
	SHAP	Euribor Rate	$-.004$	.016	.03	Marital Status
	LR	Illiterate	$-.07$	$-.0045$	.01	Age, Marital Status
Folktables	LIME	Age	$-.0007$	$-.11$	.21	Marital Status
	SHAP	Education	.023	.15	.03	Asian-American
	LR	Self-Employed	$-.26$	$-.06$	.02	White-American

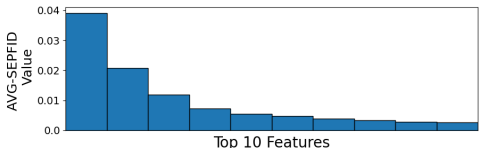
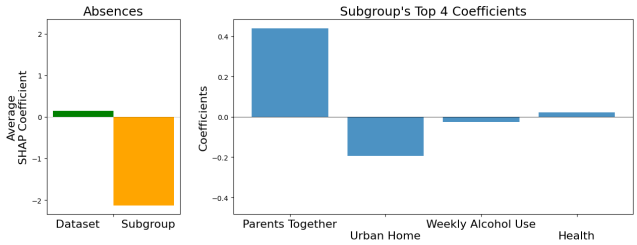


Figure 3: Distribution of AVG-FID on the top features from the BANK dataset using LIME. We see a sharp drop off in AVG-FID. This pattern is seen in all datasets and notions.

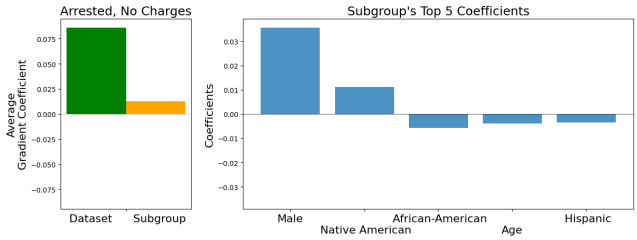
trained on points from a subgroup defined by older, single individuals, puts more importance on job=housemaid when predicting likelihood in signing up for an account. Finally on Folktables, we see that LIN-FID assigns much lower weight to the job=military feature among a subgroup that is mainly white and divorced people than in the overall dataset when predicting income. These interesting examples, in conjunction with the results reported in Table 1, highlight the usefulness of our method in finding subgroups where a concerned analyst or domain expert could dig deeper to determine how biases might be manifesting themselves in the data and how to correct for them.

5.4. Comparison of FID Values on Rich vs. Marginal Subgroups

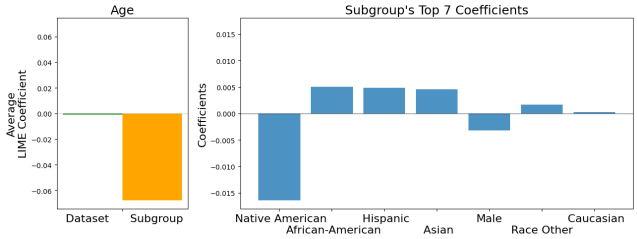
To better quantify the advantage of rich subgroups, we performed the same analysis but only searching over the marginal subgroup space. For each dataset and importance notion pair, we established the finite list of subgroups defined by a single sensitive characteristic and computed the FID for each of these subgroups. In Figure 5, we compare some of the maximal AVG-FID rich subgroups shown in Figure 2 to the maximal AVG-FID marginal subgroup for the same feature. In about half of the cases, the AVG-FID of the marginal subgroup was similar to the rich subgroup. In the other cases, expanding our subgroup classes to include rich subgroups defined by linear functions of the sensitive attributes enabled us to find a subgroup that had a higher AVG-FID. For example, in Figure 5b, we can see that on the COMPAS R dataset using GRAD as the importance notion, Arrested, No Charges had a rich subgroup with AVG-FID that was 4 times less than on the full dataset. However, we were unable to find any sub-



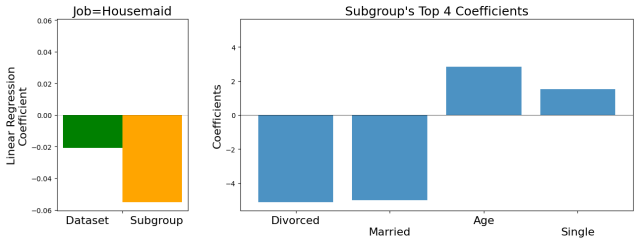
(a) Student: Predicting grade outcomes



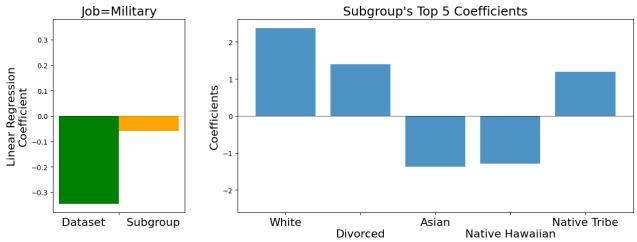
(b) COMPAS: Predicting 2-year recidivism



(c) COMPAS: Predicting decile risk score



(d) Bank: Predicting whether bank deposit is made



(e) Folktables: Predicting income >$50k

Figure 4: Exploration of key subgroup/feature pairs found for each dataset. The first graph shows the change in feature importance from whole dataset to subgroup. The second graph shows the main coefficients that define the subgroup.

group in the marginal space where the importance of the feature was nearly as different (comparisons for all datasets are available in Appendix K). In some cases, the marginal subgroup performs slightly better than the rich subgroup (Figure 6). This happens when using rich subgroups does not offer any substantial advantage over marginal subgroups, and the empirical error tolerance in Algorithm 1 stops the convergence early.

Perhaps an even more important practical advantage of Algorithm 1's ability to optimize over rich subgroups, is that it allows protected subgroups to be defined as functions of continuous variables. For example, `age` is easily included in our formulation, while capturing `age` with marginal subgroups requires first bucketing into age groups and then one-hot encoding these groups, which comes with statistical, explanatory, and computational drawbacks. As mentioned in Section 3, our framework also allows individuals to be part of multiple groups, for a example a multiracial individual who might be better represented as a fractional member of different racial groups, rather than a member of a single discrete one. This kind of data would be impossible to capture with marginal subgroups.

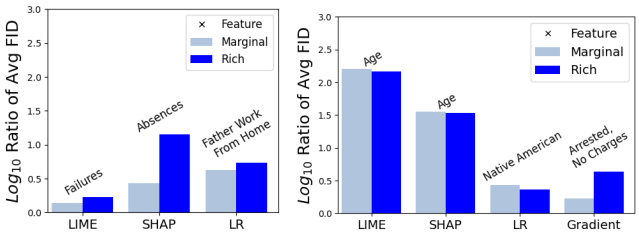
5.5. Fairness Metrics

While large `AVG-FID` values with respect to a given feature and importance notion do not guarantee disparities in common fairness metrics, which are not typically defined in terms of a specific reference feature, it is natural to ask if these notions are correlated: do subgroups with large `AVG-FID` have large disparities in fairness metrics, and do subgroups that have large disparities in fairness metrics have particularly large `AVG-FID` values for some feature?

We examine the first question in Table 2. We find that these high `AVG-FID` subgroups tend to have significant disparities in traditional fairness metrics. Although the metrics are not always *worse* on g, this reinforces the intuition that subgroups with high `AVG-FID` require greater scrutiny. In Table 3, we study the converse question, where we use the GerryFair code of (Kearns et al., 2018) to find rich subgroups that maximally violate FPR disparity, and then compute the `AVG-FID` on those subgroups. We find that they also have features with high `AVG-FID`, albeit not as large as those found by Algorithm 1, which explicitly optimizes for `AVG-FID`. These two results highlight the usefulness of our method in identifying potentially high risk subgroups.

6. Discussion

In this paper we establish feature importance disparities as an important tool to aid in `DBI`s. One benefit to our work, is that as progress is made in feature importance methods, they can be leveraged in our Algorithm 1 (if they are sep-



(a) Student (b) COMPAS R

Figure 5: Comparisons of some maximal FID rich subgroups to the maximal FID marginal subgroup on the same feature using the same log-scale as in Figure 2. The feature associated with the subgroups is written above each bar.

Table 2: Fairness metrics of high AVG−FID subgroups. COMPAS D and Student were excluded since they use non-binary y, making classification metrics less comparable. We measured the 3 fairness types outlined by (Barocas et al., 2019): $P(\hat{Y}=1)$, true/false positive rates, and expected calibration error. *<metric>*$_\Delta$ is the metric on g minus the metric on X.

Dataset	Notion	F	Defining Features of g	$\hat{Y}_\Delta$	TPR_Δ	FPR_Δ	ECE_Δ
COMPAS R	LIME	Age	Native-American	−.16	−.2	−.07	.24
	SHAP	Age	Asian-American	.37	.26	.4	−.12
	GRAD	Arrest, No Charge	Native-American	−.24	−.35	−.12	.37
Bank	LIME	# of Employees	Marital Status	.11	.08	.07	−.15
	SHAP	Euribor Rate	Marital Status	.11	.08	.07	−.17
Folktables	LIME	Age	Marital Status	−.15	−.09	−.07	.19
	SHAP	Education	Asian-American	.17	.2	.05	−.09

Table 3: Comparing top features and respective AVG−FID of g found via our method (AVG−FID$_{FID}$) and found by (Kearns et al., 2018) (AVG−FID$_{gerry}$). As in Table 2, COMPAS D and Student were excluded.

Dataset	Notion	F_{FID}	AVG−FID$_{FID}$	F_{gerry}	AVG−FID$_{gerry}$
COMPAS R	LIME	Age	.14	Age	.04
	SHAP	Age	.4	Age	.06
	GRAD	Arrest, No Charge	.09	Male	.03
Bank	LIME	# of Employees	.03	# of Employees	.008
	SHAP	Euribor Rate	.016	Emp Var Rate	.004
Folktables	LIME	Age	.11	Age	.05
	SHAP	Education	.13	Age	.05

arable). As we discuss, given observed feature/subgroup disparities, there are many possible next steps an analyst could take to root out dataset bias, from interventions in the training or feature selection process, to more involved investigations into how the data was collected. This perspective is complementary to recent work in the fairness literature that focuses on understanding and documenting data sources (Gebru et al., 2018; Fabris et al., 2022). On the algorithms side, there has been little work to systematically identify the sources of bias in the data, as opposed to developing methods that remove bias during model training or representation learning. This paper represents one attempt at developing this kind of "data-centric" fairness method through the lens of feature importance. There is much more work that can be done to determine what the right subset of features to collect are, how to detect and mitigate biases in the data collection process, and which subsets of data or features should be retained prior to model training.

Acknowledgements

The authors would like to thank Cynthia Dwork for the initial conversations that motivated the problem direction.

Impact Statement

This paper presents work whose goal is to empower practitioners to investigate potential sources of bias in common datasets used for machine learning. As such, this work has the potential to contribute to fairer and more democratized algorithmic decision making across many application domains. We emphasize, as is a common refrain in the related body of literature, that no method is a silver bullet towards mitigating bias in ML systems.

References

Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., and Wallach, H. A reductions approach to fair classification. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 60–69. PMLR, 10–15 Jul 2018a. URL https://proceedings.mlr.press/v80/agarwal18a.html.

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. M. A reductions approach to fair classification. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 60–69. PMLR, 2018b. URL http://proceedings.mlr.press/v80/agarwal18a.html.

Agarwal, C., Johnson, N., Pawelczyk, M., Krishna, S., Saxena, E., Zitnik, M., and Lakkaraju, H. Rethinking stability for attribution-based explanations, 2022a.

Agarwal, C., Krishna, S., Saxena, E., Pawelczyk, M., Johnson, N., Puri, I., Zitnik, M., and Lakkaraju, H. Openxai: Towards a transparent evaluation of model explanations. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 15784–15799. Curran Associates, Inc., 2022b. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/65398a0eba88c9b4a1c38ae405b125ef-Paper-Datasets_and_Benchmarks.pdf.

Alvarez-Melis, D. and Jaakkola, T. S. On the robustness of interpretability methods, 2018.

Angwin, J., Larson, J., Mattu, S., and Kirchner, L. Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks., 2016. URL https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing. [accessed: 15.11.2020].

Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., and Müller, K.-R. How to explain individual classification decisions. *Journal of Machine Learning Research*, 11(61):1803–1831, 2010. URL http://jmlr.org/papers/v11/baehrens10a.html.

Balagopalan, A., Zhang, H., Hamidieh, K., Hartvigsen, T., Rudzicz, F., and Ghassemi, M. The road to explainability is paved with bias: Measuring the fairness of explanations. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, pp. 1194–1206, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393522. doi: 10.1145/3531146.3533179. URL https://doi.org/10.1145/3531146.3533179.

Bansal, N., Agarwal, C., and Nguyen, A. Sam: The sensitivity of attribution methods to hyperparameters, 2020.

Barocas, S., Hardt, M., and Narayanan, A. *Fairness and Machine Learning: Limitations and Opportunities*. fairmlbook.org, 2019. http://www.fairmlbook.org.

Begley, T., Schwedes, T., Frye, C., and Feige, I. Explainability for fair machine learning. *CoRR*, abs/2010.07389, 2020. URL https://arxiv.org/abs/2010.07389.

Bolukbasi, T., Chang, K.-W., Zou, J., Saligrama, V., and Kalai, A. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pp. 4356–4364, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

Boxer, K. S., au2, E. M. I., and Neill, D. B. Auditing predictive models for intersectional biases, 2023.

Buolamwini, J. and Gebru, T. Gender shades: Intersectional accuracy disparities in commercial gender classification. In Friedler, S. A. and Wilson, C. (eds.), *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pp. 77–91. PMLR, 23–24 Feb 2018. URL https://proceedings.mlr.press/v81/buolamwini18a.html.

Bureau, U. C. 2014-2018 acs pums data dictionary, 2020. URL https://www2.census.gov/programs-surveys/acs/tech_docs/pums/data_dict/PUMS_Data_Dictionary_2014-2018.pdf.

Caton, S. and Haas, C. Fairness in machine learning: A survey. *ACM Comput. Surv.*, aug 2023. ISSN 0360-0300. doi: 10.1145/3616865. URL https://doi.org/10.1145/3616865. Just Accepted.

Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2):153–163, 2017. doi: 10.1089/big.2016.0047. URL https://doi.org/10.1089/big.2016.0047. PMID: 28632438.

Cortez, P. and Silva, A. Using data mining to predict secondary school student performance. In *Proceedings of 5th FUture BUsiness TEChnology Conference*, pp. 5–12, 2008. URL http://www3.dsi.uminho.pt/pcortez/student.pdf.

Dai, J., Upadhyay, S., Aivodji, U., Bach, S. H., and Lakkaraju, H. Fairness via explanation quality. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*. ACM, jul 2022. doi: 10.1145/3514094.3534159. URL https://doi.org/10.1145%2F3514094.3534159.

Dastile, X., Celik, T., and Potsane, M. Statistical and machine learning models in credit scoring: A systematic literature survey. *Applied Soft Computing*, 91:106263, 2020. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2020.106263. URL https://www.sciencedirect.com/science/article/pii/S1568494620302039.

Dimanov, B., Bhatt, U., Jamnik, M., and Weller, A. You shouldn't trust me: Learning models which conceal unfairness from multiple explanation methods. In *SafeAI@AAAI*, 2020. URL https://api.semanticscholar.org/CorpusID:211838210.

Ding, F., Hardt, M., Miller, J., and Schmidt, L. Retiring adult: New datasets for fair machine learning. *Advances in Neural Information Processing Systems 34 (NeurIPS)*, 2021. URL https://par.nsf.gov/biblio/10341458.

Dwork, C. and Ilvento, C. Fairness under composition. *CoRR*, abs/1806.06122, 2018. URL http://arxiv.org/abs/1806.06122.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pp. 214–226, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450311151. doi: 10.1145/2090236.2090255. URL https://doi.org/10.1145/2090236.2090255.

Fabris, A., Messina, S., Silvello, G., and Susto, G. A. Algorithmic fairness datasets: the story so far. *CoRR*, abs/2202.01711, 2022. URL https://arxiv.org/abs/2202.01711.

Freund, Y. and Schapire, R. E. Game theory, on-line prediction and boosting. In *Proceedings of the Ninth Annual Conference on Computational Learning Theory*, COLT '96, pp. 325–332, New York, NY, USA, 1996. Association for Computing Machinery. ISBN 0897918118. doi: 10.1145/238061.238163. URL https://doi.org/10.1145/238061.238163.

Gebru, T., Morgenstern, J., Vecchione, B., Vaughan, J. W., Wallach, H. M., III, H. D., and Crawford, K. Datasheets for datasets. *CoRR*, abs/1803.09010, 2018. URL http://arxiv.org/abs/1803.09010.

Gopalan, P., Kalai, A. T., Reingold, O., Sharan, V., and Wieder, U. Omnipredictors. In Braverman, M. (ed.), *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, volume 215 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 79:1–79:21, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-217-4. doi: 10.4230/LIPIcs.ITCS.2022.79. URL https://drops.dagstuhl.de/opus/volltexte/2022/15675.

Grabowicz, P. A., Perello, N., and Mishra, A. Marrying fairness and explainability in supervised learning. In *2022 ACM Conference on Fairness, Accountability, and Transparency*. ACM, jun 2022. doi: 10.1145/3531146.3533236. URL https://doi.org/10.1145%2F3531146.3533236.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 3315–3323, 2016a. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/9d2682367c3935defcb1f9e247a97c0d-Paper.pdf.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pp. 3323–3331, Red Hook, NY, USA, 2016b. Curran Associates Inc. ISBN 9781510838819.

Hebert-Johnson, U., Kim, M., Reingold, O., and Rothblum, G. Multicalibration: Calibration for the (Computationally-identifiable) masses. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1939–1948. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/hebert-johnson18a.html.

Hu, L., Livni Navon, I. R., Reingold, O., and Yang, C. Omnipredictors for constrained optimization. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp.

13497–13527. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/hu23b.html.

Ingram, E., Gursoy, F., and Kakadiaris, I. A. Accuracy, fairness, and interpretability of machine learning criminal recidivism models. In *2022 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT)*, pp. 233–241, Los Alamitos, CA, USA, dec 2022. IEEE Computer Society. doi: 10.1109/BDCAT56447.2022.00040. URL https://doi.ieeecomputersociety.org/10.1109/BDCAT56447.2022.00040.

Joseph, M., Kearns, M., Morgenstern, J., Neel, S., and Roth, A. Meritocratic fairness for infinite and contextual bandits. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, pp. 158–163, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450360128. doi: 10.1145/3278721.3278764. URL https://doi.org/10.1145/3278721.3278764.

Jung, C., Lee, C., Pai, M., Roth, A., and Vohra, R. Moment multicalibration for uncertainty estimation. In Belkin, M. and Kpotufe, S. (eds.), *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 2634–2678. PMLR, 15–19 Aug 2021. URL https://proceedings.mlr.press/v134/jung21a.html.

Kamiran, F. and Calders, T. Data preprocessing techniques for classification without discrimination. *Knowledge and information systems*, 33(1):1–33, 2012. doi: https://doi.org/10.1007/s10115-011-0463-8.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2564–2572. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/kearns18a.html.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. An empirical study of rich subgroup fairness for machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, pp. 100–109, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361255. doi: 10.1145/3287560.3287592. URL https://doi.org/10.1145/3287560.3287592.

Keddell, E. Algorithmic justice in child protection: Statistical fairness, social justice and the implications for practice. *Social Sciences*, 8(10), 2019. ISSN 2076-0760. doi: 10.3390/socsci8100281. URL https://www.mdpi.com/2076-0760/8/10/281.

Kindler, J. A simple proof of sion's minimax theorem. *Am. Math. Mon.*, 112(4):356–358, 2005. URL http://www.jstor.org/stable/30037472.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Kivinen, J. and Warmuth, M. K. Exponentiated gradient versus gradient descent for linear predictors. *Inf. Comput.*, 132(1):1–63, 1997. doi: 10.1006/inco.1996.2612. URL https://doi.org/10.1006/inco.1996.2612.

Kleinberg, J., Mullainathan, S., and Raghavan, M. Inherent Trade-Offs in the Fair Determination of Risk Scores. In Papadimitriou, C. H. (ed.), *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 43:1–43:23, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-029-3. doi: 10.4230/LIPIcs.ITCS.2017.43. URL http://drops.dagstuhl.de/opus/volltexte/2017/8156.

Kleinberg, J. M., Mullainathan, S., and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. In *Information Technology Convergence and Services*, 2016. URL https://api.semanticscholar.org/CorpusID:12845273.

Krishna, S., Han, T., Gu, A., Pombra, J., Jabbari, S., Wu, S., and Lakkaraju, H. The disagreement problem in explainable machine learning: A practitioner's perspective. *CoRR*, abs/2202.01602, 2022. URL https://arxiv.org/abs/2202.01602.

Kusner, M. J., Loftus, J., Russell, C., and Silva, R. Counterfactual fairness. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/a486cd07e4ac3d270571622f4f316ec5-Paper.pdf.

Li, Y. and Vasconcelos, N. Repair: Removing representation bias by dataset resampling. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*

(CVPR), pp. 9564–9573, 2019. doi: 10.1109/CVPR.2019.00980.

Liu, J., Zhong, C., Seltzer, M., and Rudin, C. Fast sparse classification for generalized linear and additive models. In Camps-Valls, G., Ruiz, F. J. R., and Valera, I. (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 9304–9333. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/liu22f.html.

Lundberg, S. M. Explaining quantitative measures of fairness. In *Fair & Responsible AI Workshop@ CHI2020*, 2020.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Miller, C. Algorithms and bias: Q. and a. with cynthia dwork, Aug 2015.

Molnar, C. *Interpretable Machine Learning*. 2 edition, 2022. URL https://christophm.github.io/interpretable-ml-book.

Moro, S., Cortez, P., and Rita, P. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014. ISSN 0167-9236. doi: https://doi.org/10.1016/j.dss.2014.03.001. URL https://www.sciencedirect.com/science/article/pii/S016792361400061X.

Neill, D. B. Fast Subset Scan for Spatial Pattern Detection. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 74(2):337–360, 01 2012. ISSN 1369-7412. doi: 10.1111/j.1467-9868.2011.01014.x. URL https://doi.org/10.1111/j.1467-9868.2011.01014.x.

Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. On fairness and calibration. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/b8b9c74ac526fffbeb2d39ab038d1cd7-Paper.pdf.

Quinlan, J. R. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986. doi: 10.1007/BF00116251. URL https://doi.org/10.1007/BF00116251.

Ribeiro, M. T., Singh, S., and Guestrin, C. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013. URL http://dblp.uni-trier.de/db/journals/corr/corr1312.html#SimonyanVZ13.

Slack, D., Hilgard, S., Jia, E., Singh, S., and Lakkaraju, H. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, pp. 180–186, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450371100. doi: 10.1145/3375627.3375830. URL https://doi.org/10.1145/3375627.3375830.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3319–3328. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/sundararajan17a.html.

Tommasi, T., Patricia, N., Caputo, B., and Tuytelaars, T. A deeper look at dataset bias. In *Domain adaptation in computer vision applications*, pp. 37–55. Springer, 2017.

Zemel, R. S., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 325–333. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/zemel13.html.

Zhang, Z. and Neill, D. B. Identifying significant predictive bias in classifiers. *ArXiv*, abs/1611.08292, 2016. URL https://api.semanticscholar.org/CorpusID:2256765.

A. Limitations

Importantly, we eschew any broader claims that large `FID` *necessarily* implies a mathematical conclusion about the *fairness* of the underlying classification model in all cases. It is known that even the most popular and natural fairness metrics are impossible to satisfy simultaneously, and so we would run up against the problem of determining what it means for a model to be *fair* (Chouldechova, 2017; Kleinberg et al., 2017). By detecting anomalous subgroups with respect to feature importance, our approach can signal to a domain expert that perhaps there are issues such as feature collection or measurement bias. This will facilitate the next steps of testing the resulting hypotheses, and ultimately intervening to address disparities and improve fairness outcomes. Concerns about the stability and robustness of the most widely used feature importance notions, including the ones we study, have been raised (Dai et al., 2022; Agarwal et al., 2022a; Alvarez-Melis & Jaakkola, 2018; Bansal et al., 2020; Dimanov et al., 2020; Slack et al., 2020) and these notions are often at odds with each other, so none can be considered definitive (Krishna et al., 2022). Regardless of these limitations, these notions are used widely in practice today, and are still useful as a diagnostic tool as we propose here in order to uncover potentially interesting biases. Lastly, our methods, like nearly all prior works on fairness, require tabular datasets that have defined the sensitive features apriori, a process more difficult in text or image datasets where bias is still a concern (Buolamwini & Gebru, 2018; Bolukbasi et al., 2016). Overall, the methods developed here represent a part of the algorithmic toolkit that domain experts may use in rooting out bias.

B. Reproducibility

Specific details for the experiments such as the hyperparameters used are available in Appendix I. The source code used for these experiments is provided in the supplementary material. Specifically, `run_separable.py` and `run_linear.py` are the scripts where the importance notion (Appendix I.2), dataset (Appendix I.3), and other parameters are specified before running. The `experiments/` directory contains scripts used for the comparison of rich and marginal subgroups as seen in Appendix K and for the fairness comparison experiments in Subsection 5.5.

C. Additional Related Work

Fairness in Machine Learning. Much of the work in fairness in machine learning typically concerns the implementation of a new fairness notion in a given learning setting; either an individual fairness notion (Dwork et al., 2012; Joseph et al., 2018), one based on equalizing a statistical rate across protected subgroups (Hardt et al., 2016b; Pleiss et al., 2017), or one based on an underlying causal model (Kusner et al., 2017). With a given notion of fairness in hand, approaches to learning fair classifiers can be typically classified as “in-processing", or trying to simultaneously learn a classifier and satisfy a fairness constraint, “post-processing" which takes a learned classifier and post-processes it to satisfy a fairness definition (Hardt et al., 2016b), or most closely related to the motivation behind this paper, pre-processing the data to remove bias. Existing work on dataset bias serve as high level motivation for our work.

Feature Importance Notions. The local explanation methods mentioned in Section 2 include model-agnostic methods like LIME or SHAP (Ribeiro et al., 2016; Lundberg & Lee, 2017), methods like saliency maps (Simonyan et al., 2013; Sundararajan et al., 2017; Baehrens et al., 2010) that require h to be differentiable in x, or model-specific methods that depend on the classifier. In addition to these explanation methods, there are also global methods that attempt to explain the entire model behavior and so can be run on the entire subgroup. Our `LIN-FID` method as described in Appendix H is a global method that relies on training an inherently interpretable model (linear regression) on the subgroup and inspecting its coefficients. Other inherently interpretable models that could be used to define a notion of subgroup importance include decision trees (Quinlan, 1986) and generalized additive models (Liu et al., 2022).

Fairness and Interpretability. Although no existing work examines the role of feature importance notions in detecting disparities in rich subgroups, there is a small amount of existing work examining explainability in the context of fairness. The recent (Grabowicz et al., 2022) formalizes induced discrimination as a function of the SHAP values assigned to sensitive features, and proposes a method to learn classifiers where the protected attributes have low influence. (Begley et al., 2020) applies a similar approach, attributing a models overall unfairness to its individual features using the Shapley value, and proposing an intervention to improve fairness. (Ingram et al., 2022) examines machine learning models to predict recidivism, and empirically shows tradeoffs between model accuracy, fairness, and interpretability.

Additionally, (Lundberg, 2020) decomposes feature attribution explanations and fairness metrics into additive components and observes the relationship between the fairness metrics and input features. Our work does not try to decompose fairness

metrics into additive components and also focuses on non-additive feature explanations. Furthermore, our consideration of rich subgroups is a novel addition to the space.

D. Computing the Optimal Subgroup

Algorithm 1 Iterative Constrained Optimization

Input: Dataset $X^n, |X^n| = n$, hypothesis h, feature of interest f_j, separable feature importance function F, size constraints α_L and α_U, size violation indicators Φ_L and Φ_U, size penalty bound B, CSC oracle for $\mathcal{G}$, $CSC_{\mathcal{G}}(c^0, c^1)$, accuracy ν.
Initialize:
Feature importance vector $\mathbf{C} = (F(f_j, X_i, h))_{i=1}^n$
Gradient weight parameter $\theta_1 = (0, 0)$
Learning rate $\eta = \frac{\nu}{2n^2 B}$
for $t = 1, 2, ...$ **do**
  # Exponentiated gradient weights
  $\lambda_{t,0} = B\frac{exp(\theta_{t,0})}{1+exp(\theta_{t,1})}, \quad \lambda_{t,1} = B\frac{exp(\theta_{t,1})}{1+exp(\theta_{t,0})}$

  # Costs vector
  $c_t^1 = (\mathbf{C}_i - \lambda_{t,0} + \lambda_{t,1})_{i=1}^n$

  # Get g with max disparity computed via CSC oracle
  $g_t = CSC_{\mathcal{G}}(\mathbf{0}, c_t^1)$

  # Compute Lagrangian
  $\hat{p}_{\mathcal{G}}^t = \frac{1}{t}\sum_{t'=1}^t g_{t'}$
  $\lambda_t' = (B\Phi_L(\hat{p}_{\mathcal{G}}^t), B\Phi_U(\hat{p}_{\mathcal{G}}^t))$
  $\overline{L} = L(\hat{p}_{\mathcal{G}}^t, \lambda_t')$

  $\hat{p}_{\lambda}^t = \frac{1}{t}\sum_{t'=1}^t (\lambda_{t',0}, \lambda_{t',1})$
  $g_t' = CSC_{\mathcal{G}}(\mathbf{0}, (\mathbf{C}_i - \hat{p}_{\lambda_0}^t + \hat{p}_{\lambda_1}^t)_{i=1}^n)$
  $\underline{L} = L(g_t', \hat{p}_{\lambda}^t)$

  $v_t = \max\left(|L(\hat{p}_{\mathcal{G}}^t, \hat{p}_{\lambda}^t) - \underline{L}|, |\overline{L} - L(\hat{p}_{\mathcal{G}}^t, \hat{p}_{\lambda}^t)|\right)$

  # Check termination condition
  if $v_t \leq v$ **then**
    Return $\hat{p}_{\mathcal{G}}^t, \hat{p}_{\lambda}^t$
  end if

  # Exponentiated gradient update
  Set $\theta_{t+1} = \theta_t + \eta(\alpha_L - |g_t|, |g_t| - \alpha_U)$
end for

We start by showing that for the unconstrained problem, computing the subgroup g_j^* that maximizes $\texttt{FID}(f_j, g, h)$ over $\mathcal{G}$ can be computed in two calls to CSC$_{\mathcal{G}}$ when F is separable.

Lemma D.1. *If F is separable and CSC$_{\mathcal{G}}$ is a CSC oracle for $\mathcal{G}$, then for any feature f_j, g_j^* can be computed with two oracle calls.*

Proof. By definition $g_j^* = \text{argmax}_{g\in\mathcal{G}}\texttt{FID}(j, g) = \text{argmax}_{g\in\mathcal{G}}|F(f_j, X^n, h) - F(f_j, g, h)| = \text{argmax}_{g\in\{g^+, g^-\}}\texttt{FID}(j, g)$, where $g^+ = \text{argmax}_{g\in\mathcal{G}}F(f_j, g, h), g^- = \text{argmin}_{g\in\mathcal{G}}F(f_j, g, h)$. By the definition of separability, we can write

$$F(f_j, g(X^n), h) = \sum_{X\in g(X^n)} F'(f_j, X, h) = \sum_{i=1}^n g(X_i)F'(f_j, X_i, h)$$

Then letting $c_k^0 = 0$ and $c_k^1 = -F'(f_j, X_k, h)$ for $k = 1, \ldots n$, we see that $g^+ = \text{CSC}_g((c_k^0, c_k^1)), g^- = \text{CSC}_g((c_k^0, -c_k^1))$. This establishes the claim. □

Theorem 4.1: Let F be a separable notion, fix a classifier h, subgroup class $\mathcal{G}$, and oracle $\text{CSC}_{\mathcal{G}}$. Then fixing a feature of interest f_j, we will run Algorithm 1 twice; once with FID given by F, and once with FID given by $-F$. Let $\hat{p}_{\mathcal{G}}^T$ be the distribution returned after $T = O(\frac{4n^2B^2}{\nu^2})$ iterations by Algorithm 1 that achieves the larger value of $\mathbb{E}[\texttt{FID}(j, g)]$. Then:

$$\begin{aligned} \texttt{FID}(j, g_j^*) - \mathbb{E}_{g \sim \hat{p}_{\mathcal{G}}^T}[\texttt{FID}(j, g)] &\leq \nu \\ |\Phi_L(g)|, |\Phi_U(g)| &\leq \frac{1 + 2\nu}{B} \end{aligned} \tag{3}$$

Proof. We start by transforming our constrained optimization into optimizing a $\min - \max$ objective. The $\min$ player, referred to as the *subgroup player* will be solving a CSC problem over the class $\mathcal{G}$ at each iteration, while the $\max$ player, called the *dual player*, will be adjusting the dual weights λ on the two constraints using the exponentiated gradient algorithm (Kivinen & Warmuth, 1997). By Lemma D.2 (Freund & Schapire, 1996), we know that if each player implements a *no-regret* strategy, then the error of subgroup found after T rounds is sub-optimal by at most the average cumulative regret of both players. The regret bound for the exponentiated gradient descent ensures this occurs in *poly(n)* rounds.

As in (Kearns et al., 2018; Agarwal et al., 2018a), we first relax Equation 1 to optimize over all *distributions* over subgroups, and we enforce that our constraints hold in expectation over this distribution. Our new optimization problem becomes:

$$\begin{aligned} \min_{p_g \in \Delta(\mathcal{G})} \quad & \mathbb{E}_{g \sim p_g}[\sum_{i=1}^{n} g(x_i) F'(f_j, x_i, h)] \\ \text{s.t.} \quad & \mathbb{E}_{g \sim p_g}[\Phi_L(g)] \leq 0 \\ & \mathbb{E}_{g \sim p_g}[\Phi_U(g)] \leq 0 \end{aligned} \tag{4}$$

We note that while $|\mathcal{G}|$ may be infinite, the number of distinct labelings of X by elements of $\mathcal{G}$ is finite; we denote the number of these by $|\mathcal{G}(X)|$. Then since Equation 4 is a finite linear program in $|\mathcal{G}(X)|$ variables, it satisfies strong duality, and we can write:

$$\begin{aligned} (p_g^*, \lambda^*) &= \text{argmin}_{p_g \in \Delta(\mathcal{G})} \text{argmax}_{\lambda \in \Lambda} \mathbb{E}_{g \sim p_g}[L(g, \lambda)] = \text{argmin}_{p_g \in \Delta(\mathcal{G})} \text{argmax}_{\lambda \in \Lambda} L(p_g, \lambda) \\ \text{with} \quad L(g, \lambda) &= \sum_{x \in X} g(x) F(f_j, x, h) + \lambda_L \Phi_L + \lambda_U \Phi_U, \quad L(p_g, \lambda) = \mathbb{E}_{g \sim p_g}[L(g, \lambda)] \end{aligned}$$

As in (Kearns et al., 2018) $\Lambda = \{\lambda \in \mathbb{R}^2 \mid \|\lambda\|_1 \leq B\}$ is chosen to make the domain compact, and does not change the optimal parameters as long as B is sufficiently large, i.e. $\|\lambda^*\|_1 \leq B$. In practice, this is a hyperparameter of Algorithm 1, similar to (Agarwal et al., 2018a; Kearns et al., 2018). Then we follow the development in (Agarwal et al., 2018a; Kearns et al., 2018) to show that we can compute (p_g^*, λ^*) efficiently by implementing *no-regret* strategies for the subgroup player (p_g) and the dual player (λ).

Formally, since $\mathbb{E}_{g \sim p_g}[L(g, \Lambda)]$ is bi-linear in p_g, λ, and $\Lambda, \Delta(\mathcal{G})$ are convex and compact, by Sion's minimax theorem (Kindler, 2005):

$$\min_{p_g \in \Delta(G)} \max_{\lambda \in \Lambda} L(p_g, \lambda) = \max_{\lambda \in \Lambda} \min_{p_g \in \Delta(G)} L(p_g, \lambda) = \text{OPT} \tag{5}$$

Then by Theorem 4.5 in (Kearns et al., 2018), we know that if (p_g^*, λ^*) is a ν-approximate min-max solution to Equation 5 in the sense that

$$\begin{aligned} \text{if:} \quad & L(p_g^*, \lambda^*) \leq \min_{p \in \Delta(\mathcal{G})} L(p, \lambda^*) + \nu, L(p_g, \lambda) \geq \max_{\lambda \in \Lambda} L(p^*, \lambda), \\ \text{then:} \quad & F(f_j, p_g^*, h) \leq OPT + 2\nu, \quad |\Phi_L(g)|, |\Phi_U(g)| \leq \frac{1 + 2\nu}{B} \end{aligned} \tag{6}$$

So in order to compute an approximately optimal subgroup distribution p_g^*, it suffices to compute an approximate min-max solution of Equation 5. In order to do that we rely on the classic result of (Freund & Schapire, 1996) that states that if the subgroup player best responds, and if the dual player achieves low regret, then as the average regret converges to zero, so does the sub-optimality of the average strategies found so far.

Lemma D.2 ((Freund & Schapire, 1996)). *Let $p_1^\lambda, \dots p_T^\lambda$ be a sequence of distributions over Λ, played by the dual player, and let $g^1, \dots g^T$ be the subgroup players best responses against these distributions respectively. Let $\hat{\lambda}_T = \frac{1}{T}\sum_{t=1}^T p_t^\lambda, \hat{p}_g = \frac{1}{T}\sum_{t=1}^T g_t$. Then if*

$$\sum_{t=1}^T \mathbb{E}_{\lambda \sim p_t^\lambda}[L(g_t, \lambda)] - \min_{\lambda \in \Lambda} \sum_{t=1}^T [L(g_t, \lambda)] \leq \nu T,$$

Then $(\hat{\lambda}_T, \hat{p}_g)$ is a ν-approximate minimax equilibrium of the game.

To establish Theorem 4.1, we need to show (i) that we can efficiently implement the subgroup players best response using $\text{CSC}_\mathcal{G}$ and (ii) we need to translate the regret bound for the dual players best response into a statement about optimality, using Lemma D.2. Establishing (i) is immediate, since at each round t, if $\lambda_{t,0} = \mathbb{E}_{p_t^\lambda}[\lambda_L], \lambda_{t,1} = \mathbb{E}_{p_t^\lambda}[\lambda_U]$, then the best response problem is:

$$\text{argmin}_{p_g \in \Delta(G)} \mathbb{E}_{g \sim p_g}[\sum_{x \in X} g(x) F(f_j, x, h) + \lambda_{t,0}\Phi_L + \lambda_{t,1}\Phi_U]$$

Which can further be simplified to:

$$\text{argmin}_{g \in G} \sum_{x \in X} g(x)(F(f_j, x, h) - \lambda_L + \lambda_U) \tag{7}$$

This can be computed with a single call of $\text{CSC}_\mathcal{G}$, as desired. To establish (ii), the no-regret algorithm for the dual player's distributions, we note that at each round the dual player is playing online linear optimization over 2 dimensions. Algorithm 1 implements the exponentiated gradient algorithm (Kivinen & Warmuth, 1997), which has the following guarantee proven in Theorem 1 of (Agarwal et al., 2018a), which follows easily from the regret bound of exponentiated gradient (Kivinen & Warmuth, 1997), and Lemma D.2:

Lemma D.3 ((Agarwal et al., 2018a)). *Setting $\eta = \frac{\nu}{2n^2 B}$, Algorithm 1 returns $\hat{p}_\lambda^T$ that is a ν-approximate min-max point in at most $O(\frac{4n^2B^2}{\nu^2})$ iterations.*

Combining this result with Equation 5 completes the proof.

□

E. Proof of `AVG-FID` Primitive

In Section 4, we presented our approach that optimizing for `FID` constrained across a range of subgroup sizes will allow us to efficiently optimize for `AVG-FID`. We provide a more complete proof of that claim here:

Let g^* be the subgroup that maximizes `AVG-FID`. Without loss of generality, $g^* = \text{argmax}_{g \in \mathcal{G}} \frac{1}{n|g|} \sum g(x) F'(f_j, X, h)$ (we drop the absolute value because we can also set $F' = -F$). Then it is necessarily true, that g^* also solves the constrained optimization problem $\text{argmax}_{g \in \mathcal{G}} \frac{1}{n} \sum g(x) F'(f_j, X, h)$ such that $|g| = |g^*|$, where we have dropped the normalizing term $\frac{1}{|g|}$ in the objective function, and so we are maximizing the constrained `FID`.

Now consider an interval $I = [|g^*| - \alpha, |g^*| + \alpha]$, and suppose we solve $g_I^* = \text{argmax}_{g \in \mathcal{G}} \frac{1}{n} \sum g(x) F'(f_j, X, h)$ such that $g \in I$. Then since $g^* \in I$, we know that $\frac{1}{n} \sum g^* F'(f_j, X, h) \leq \frac{1}{n} \sum g_I^*(x) F'(f_j, X, h)$. This implies that:

$$\begin{aligned}\texttt{AVG-FID}(g_I^*) &\geq \frac{1}{|g_I^*|}\frac{1}{n}\sum g^*(x)F'(f_j, X, h)\\ &= \texttt{AVG-FID}(g^*) + (\frac{1}{|g_I^*|+\alpha} - \frac{1}{|g_I^*|})\texttt{FID}(g^*)\\ &= \texttt{AVG-FID}(g^*) - \frac{\alpha}{|g^*|(|g^*|+\alpha)}\cdot\texttt{FID}(g^*)\end{aligned}$$

Given the above derivation, as $\alpha \to 0$, we have $\texttt{AVG-FID}(g_I^*) \to \texttt{AVG-FID}(g^*)$.

Hence we can compute a subgroup g that approximately optimizes the AVG-FID if we find an appropriately small interval I aroudn $|g^*|$. Since the discretization in Section 4 covers the unit interval, we are guaranteed for sufficiently large n to find such an interval.

F. Cost Sensitive Classifier, CSC$_{\mathcal{G}}$

Definition F.1. (Cost Sensitive Classification) A Cost Sensitive Classification (CSC) problem for a hypothesis class $\mathcal{G}$ is given by a set of n tuples $\{(X_i, c_i^0, c_i^1)\}_{i=1}^n$, where c_i^0 and c_i^1 are the costs of assigning labels 0 and 1 to X_i respectively. A CSC oracle finds the classifier $\hat{g} \in \mathcal{G}$ that minimizes the total cost across all points:

$$\hat{g} = \operatorname*{argmin}_{g\in\mathcal{G}} \sum_i \Big(g(X_i)c_i^1 + (1-g(X_i))c_i^0\Big) \tag{8}$$

Algorithm 2 CSC$_{\mathcal{G}}$

Input: Dataset $X \subset \mathbb{R}^{d_{sens}} \times \mathbb{R}^{d_{safe}}$, costs $(c^0, c^1) \in \mathbb{R}^n$
Let X_{sens} consist of the sensitive attributes x of each $(x, x') \in X$.

 # learn to predict the cost c^0
Train linear regressor $r_0 : \mathbb{R}^{d_{sens}} \to \mathbb{R}$ on dataset (X_{sens}, c^0)

 # learn to predict the cost c^1
Train linear regressor $r_1 : \mathbb{R}^{d_{sens}} \to \mathbb{R}$ on dataset (X_{sens}, c^1)

 # predict 0 *if the estimated* $c_0 < c_1$
Define $g((x, x')) := \mathbf{1}\{(r_0 - r_1)(x) > 0\}$
Return g

G. NP-Completeness

We will show below that the fully general version of this problem (allowing any poly-time F) is NP complete. First, we will define a decision variant of the problem:

$$\delta_{X,F,h,A} = \max_{g\in\mathcal{G}, f_j} (|F(f_j, g, h) - F(f_j, X, h)|) \geq A$$

Note that a solution to the original problem trivially solves the decision variant. First, we will show the decision variant is in NP, then we will show it is NP hard via reduction to the max-cut problem.

Lemma G.1. *The decision version of this problem is in NP.*

Proof. Our witness will be the subset g and feature f_j such that

$$(|F(f_j, g, h) - F(f_j, X, h)|) \geq A$$

Given these 2, evaluation of the absolute value is polytime given that F is polytime, so the solution can be verified in polytime. □

Lemma G.2. *The decision version of this problem is NP hard.*

Proof. We will define our variables to reduce our problem to maxcut(Q, k). Given a graph defined with $V,\ E$ as the vertex and edge sets of Q (with edges defined as pairs of vertices), we will define our F, X, G, A, and h as follows:

$$\begin{aligned}
X &= V \\
h &= \text{constant classifier, maps every value to 1} \\
\mathcal{G} &= \mathcal{P}(V) \text{ i.e. all possible subsets of vertices} \\
F(f_j, g, h) &= |x \in E : x[0] \in g, x[1] \in g^c| \\
&\quad \text{–i.e. } F(j, g, h) \text{ returns the number of} \\
&\quad \text{edges cut by a particular subset, ignoring} \\
&\quad \text{its first and third argument.} \\
&\quad \text{(this is trivially computable in polynomial} \\
&\quad \text{time by iterating over the set of edges).} \\
A &= k
\end{aligned}$$

Note that $F(f_j, X, h) = 0$ by definition, and that $F \geq 0$. Therefore, $|F(f_j, g, h) - F(f_j, X, h)| = F(f_j, g, h)$, and we see that $(|F(f_j, g, h) - F(f_j, X, h)|) \geq A$ if and only if g is a subset on Q that cuts at least $A = k$ edges. Therefore an algorithm solving the decision variant of the feature importance problem also solves maxcut. □

H. Linear Feature Importance Disparity

The *non-separable* FID notion considered in this paper corresponds to training a model that is inherently interpretable on only the data in the subgroup g, and comparing the influence of feature j to the influence when trained on the dataset as a whole. Since all of the points in the subgroup can interact to produce the interpretable model, this notions typically are not separable. Below we formalize this in the case of linear regression, which is the non-separable notion we investigate in the experiments.

Definition H.1. (Linear Feature Importance Disparity). Given a subgroup g, let $\theta_g = \inf_{\theta \in \mathbb{R}^d} \mathbb{E}_{(X,y)\sim\mathcal{R}}[g(X)(\theta' X - y)^2]$, and $\theta_{\mathcal{R}} = \inf_{\theta \in \mathbb{R}^d} \mathbb{E}_{(X,y)\sim\mathcal{R}}[(\theta' X - y)^2]$. Then if e_j is the jth basis vector in $\mathbb{R}^d$, we define the *linear feature importance disparity* (LIN-FID) by

$$\texttt{LIN-FID}(j, g) = |(\theta_g - \theta_{\mathcal{R}}) \cdot e_j|$$

LIN-FID(j, g) is defined as the difference between the coefficient for feature j when training the model on the subgroup g, versus training the model on points from $\mathcal{R}$. Expanding Definition H.1 using the standard weighted least squares estimator (WLS), the feature importance for a given feature f_j and subgroup $g(X)$ is:

$$F_{lin}(j, g) = \big((Xg(X)X^T)^{-1}(X^T g(X)Y)\big) \cdot e_j, \tag{9}$$

Where $g(X)$ is a diagonal matrix of the output of the subgroup function. The coefficients of the linear regression model on the dataset X can be computed using the results from ordinary least squares (OLS): $(XX^T)^{-1}(X^T Y) \cdot e_j$.

We compute argmax$_{g\in G}$LIN-FID $=$ argmax$_{g\in G}|F_{lin}(j, X^n) - F_{lin}(j, g)|$ by finding the minimum and maximum values of $F_{lin}(j, g)$ and choosing the one with the larger difference. For the experiments in Section 5, we use logistic regression as the hypothesis class for g because it is non-linear enough to capture complex relationships in the data, but maintains interpretability in the form of its coefficients, and importantly because Equation 9 is then differentiable in the parameters θ of $g(X) = \sigma(X \cdot \theta), \sigma(x) = \frac{1}{1+e^{-x}}$. Since Equation 9 is differentiable in θ, we can use non-convex optimizers like SGD or ADAM to maximize Equation 9 over θ.

While this is an appealing notion due to its simplicity, it is not relevant unless the matrix $Xg(X)X^T$ is of full rank. We ensure this first by lower bounding the size of g via a size penalty term $P_{size} = \max(\alpha_L - |g(X_{train})|, 0) + \max(|g(X_{train})| -$

$\alpha_U, 0)$, which allows us to provide α constraints in the same manner as in the separable approach. We also add a small l_2 regularization term ϵI to $X^T g(X) X$. This forces the matrix to be invertible, avoiding issues with extremely small subgroups. Incorporating these regularization terms, Equation 9 becomes:

$$F_{lin}(j, g) = \lambda_s \cdot \left((X\sigma(X \cdot \theta_L^T)X^T + \epsilon I)^{-1} (X^T \sigma(X \cdot \theta_L^T) Y) \cdot e_j \right) + \lambda_c \cdot P_{size} \tag{10}$$

We note that LIN-FID is a similar notion to that of LIME (Ribeiro et al., 2016), but LIME estimates a local effect around each point which is then summed to get the effect in the subgroup, and so it is *separable*. It is also the case that F_{lin} is non-convex as shown below:

Lemma H.2. *F_{lin} as defined in Equation 9 is non-convex.*

Proof. We will prove this by contradiction. Assume F_{lin} is convex, which means the Hessian is positive semi-definite everywhere. First we will fix $(Xg(X)X^T))^{-1}$ to be the identity matrix, which we can do without loss of generality by scaling g by a constant. This scaling will not affect the convexity of F_{lin}.

Now, we have the simpler form of $F_{lin} = (X^T g(X) Y) \cdot e_j$. We then can compute the values of the Hessian:

$$\frac{\partial F^2}{\partial^2 g} \quad = \quad (X^T g''(X) Y) \cdot e_j$$

Consider the case where X^T is a 2×2 matrix with rows $1, 0$ and $0, -1$ and Y is a vector of ones. If g weights the second column (i.e. feature) greater than the first, then the output Hessian will be positive semi-definite. But if g weights the first column greater than the first, then it will be negative semi-definite. Since the Hessian is not positive semi-definite everywhere, F_{lin} must be non-convex over the space of g. □

This means the stationary point we converge to via gradient descent may only be locally optimal. In Section 5, we optimize Equation 10 using the ADAM optimizer (Kingma & Ba, 2015). Additional details about implementation and parameter selection are in Appendix I. Despite only locally optimal guarantees, we were still able to find (feature, subgroup) pairs with high LIN-FID for all datasets.

I. Experimental Details

I.1. Algorithmic Details

Separable Case. In order to implement Algorithm 1 over a range of $[\alpha_L, \alpha_U]$ values, we need to specify our dual norm B, learning rate η, number of iterations used T, rich subgroup class $\mathcal{G}$, and the associated oracle CSC$_\mathcal{G}$. We note that for each feature f_j, Algorithm 1 is run twice; one corresponding to maximizing $\texttt{FID}(f_j, g, h)$ and the other minimizing it. Note that in both cases our problem is a minimization, but when maximizing we simply negate all of the point wise feature importance values $F(f_j, x_i, h) \to -F(f_j, x_i, h)$. In all experiments our subgroup class $\mathcal{G}$ consists of linear threshold functions over the sensitive features: $\mathcal{G} = \{\theta \in \mathbb{R}^{d_{sens}} : \theta((x, x')) = \mathbf{1}\{\theta' x > 0\}$. We implement CSC$_\mathcal{G}$ as in (Agarwal et al., 2018a; Kearns et al., 2018) via linear regression, see Algorithm 2 in Appendix F. To ensure the dual player's response is strong enough to enforce desired size constraints, we empirically found that setting the hyperparameter $B = 10^4 \cdot \mu(f_j)$ worked well on all datasets, where $\mu(f_j)$ is the average absolute importance value for feature j over X. We set the learning rate for exponentiated gradient descent to $\eta = 10^{-5}$. Empirical testing showed that $\eta \cdot B$ should be on the order of $\mu(f_j)$ or smaller to ensure proper convergence. We found that setting the error tolerance hyperparameter $\nu = .05 \cdot \mu(f_j) \cdot n \cdot \alpha_L$ worked well in ensuring good results with decent convergence time across all datasets and values of α. For all datasets and methods we ran for at most $T = 5000$ iterations, which we observe empirically was large enough for FID values to stabilize and for $\frac{1}{T}\sum_{t=1}^{T} |g_t| \in [\alpha_L, \alpha_U]$, with the method typically converging in $T = 3000$ iterations or less. See Appendix N for a sample of convergence plots.

Non-Separable Case. For the non-separable approach, datasets were once again split into train and test sets. For Student, it was split 50-50, while COMPAS, Bank, and Folktables were split 80-20 train/test. The 50-50 split for Student was chosen so that a linear regression model would be properly fit on a small $g(X_{test})$. The parameter vector θ for a logistic regression classifier was randomly initialized with a PyTorch random seed of 0 for reproducability. We used an ADAM (Kingma & Ba, 2015) optimizer with a learning rate of .05 as our heuristic solver for the loss function.

To enforce subgroup size constraints, $\lambda_s P_{size}$ must be on a significantly larger order than $\lambda_c F_{lin}(j, g)$. Empirical testing found that values of $\lambda_s = 10^5$ and $\lambda_c = 10^{-1}$ returned appropriate subgroup sizes and also ensured smooth convergence. The optimizer ran until it converged upon a minimized linear regression coefficient, subject to the size constraints. Experimentally, this took at most 1000 iterations, see Appendix O for a sample of convergence plots. After solving twice for the minimum and maximum $F_{lin}(j, g)$ values and our subgroup function g is chosen, we fit the linear regression on both X_{test} and $g(X_{test})$ to get the final `FID`.

I.2. `FID` Notions

LIME: A random forest model h was trained on dataset X^n. Then each data point along with the corresponding probability outputs from the classifier were input into the LIME Tabular Explainer Python module. This returned the corresponding LIME explanation values.

SHAP: This was done with the same method as LIME, except using the SHAP Explainer Python module.

Vanilla Gradient: Labeled as *GRAD* in charts, the vanilla gradient importance notion was computed using the Gradient method from the OpenXAI library (Agarwal et al., 2022b). This notion only works on differentiable classifiers so in this case, h is a logistic regression classifier. We found there was no substantial difference between the choice of random forest or logistic regression for h when tested on other importance notions (Appendix L). Due to constraints on computation time, this method was only tested on the COMPAS dataset (using `Two Year Recidivism` as the target variable).

Linear Regression: For the linear regression notion, the subgroup g was chosen to be in the logistic regression hypothesis class. For a given subgroup $g(X)$, the weighted least squares (WLS) solution is found whose linear coefficients θ_g then define the feature importance value $e_j \cdot \theta_g$.

For details on the consistency of these importance notions, see Appendix P.

I.3. Datasets

These four datasets were selected on the basis of three criterion: (i) they all use features which could be considered *sensitive* to make predictions about individuals in a context where bias in a significant concern (ii) they are heavily used datasets in research on interpretability and fairness, and as such issues of bias in the datasets should be of importance to the community, and (iii) they trace out a range of number of datapoints and number of features and sensitive features, which we summarise in Table 4. For each dataset, we specified features that were "sensitive." That is, when searching for subgroups with high `FID`, we only considered rich subgroups defined by features generally covered by equal protection or privacy laws (e.g. race, gender, age, health data).

Student: This dataset aims to predict student performance in a Portugese grade school using demographic and familial data. For the purposes of this experiment, the target variable was math grades at the end of the academic year. Student was by far the smallest of the four datasets with 395 data points. The sensitive features in Student are `gender`, `parental status`, `address` (urban or rural), `daily alcohol consumption`, `weekly alcohol consumption`, and `health`. `Age` typically would be considered sensitive but since in the context of school, age is primarily an indicator of class year, this was not included as a sensitive feature. The categorical features `address`, `Mother's Job`, `Father's Job`, and `Legal Guardian` were one hot encoded.

COMPAS: This dataset uses a pre-trial defendant's personal background and past criminal record to predict risk of committing new crimes. To improve generalizability, we removed any criminal charge features that appeared fewer than 10 times. Binary counting features (e.g. `25-45 yrs old` or `5+ misdemeanors`) were dropped in favor of using the continuous feature equivalents. Additionally, the categorical variable `Race` was one-hot encoded. This brought the total number of features to 95. The sensitive features in COMPAS are `age`, `gender`, and `race` (Caucasian, African-American, Asian, Hispanic, Native American, and Other). For COMPAS, we ran all methodologies twice, once using the binary variable, `Two Year Recidivism`, as the target variable and once using the continuous variable `Decile Score`. `Two Year Recidivism` is what the model is intended to predict and is labeled as *COMPAS R* in the results. Meanwhile, `Decile Score` is what the COMPAS system uses in practice to make recommendations to judges and is labeled as *COMPAS D* in the results.

Bank: This dataset looks at whether a potential client signed up for a bank account after being contacted by marketing personnel. The sensitive features in Bank are `age` and `marital status` (married, single, or divorced). The `age` feature

in Bank is a binary variable representing whether the individual is above the age of 25.

Folktables: This dataset is derived from US Census Data. Folktables covers a variety of tasks, but we used the ACSIncome task, which predicts whether an individual makes more than $50k per year. The ACSIncome task is meant to mirror the popular Adult dataset, but with modifications to address sampling issues. For this paper, we used data from the state of Michigan in 2018. To reduce sparseness of the dataset, the `Place of Birth` feature was dropped and the `Occupation` features were consolidated into categories of work as specified in the official Census dictionary (Bureau, 2020), (e.g. people who work for the US Army, Air Force, Navy, etc. were all consolidated into `Occupation=Military`). The sensitive features in Folktables are `age`, `sex`, `marital status` (married, widowed, divorced, separated, never married/under 15 yrs old), and `race` (Caucasian, African-American, Asian, Native Hawaiian, Native American singular tribe, Native American general, Other, and 2+ races).

Table 4: Summary of Datasets

Dataset	Data Points	# of Features	# of Sensitive Features
Student	395	32	6
COMPAS	6172	95	8
Bank	30488	57	4
Folktables Income	50008	52	16

J. Synthetic Experiment

In addition to the empirical experiments on real-world datasets, we generated two synthetic datasets and used them to validate our methods in a controlled environment. In our baseline experiment, we created a dataset where the outcome y is independent of the sensitive features to confirm that our algorithm does not result in any false discovery. Next, we modified the distribution of the outcome for a subset of individuals, injecting a large `FID` in the subgroup g for feature f_j. We then confirmed that our algorithm is able to find that feature importance disparity. We discuss the dataset generation, experimental setup, and results from those two experiments here.

J.1. Baseline Case

Experimental Setup: We generated a synthetic dataset of size $n = 4000$. Each person in the dataset had randomly generated sensitive features: `age`, `sex`, and `race`. `sex` and `race` were drawn based on US Census data and `age` $\sim \mathcal{N}(50, 7)$. Three more variables were generated for each individual: a binary variable x_1 and normally distributed variables $x_2 \sim \mathcal{N}(100, 5)$ and $x_3 \sim \mathcal{N}(100, 5)$. These three additional variables were drawn independently of the sensitive features and each other. We then generated outcome $y \sim x_1 + x_2 + x_3 + \eta, \eta \sim \mathcal{N}(0, 1)$; note that y is generated from the same model for any sensitive group, so there should be no subgroups with large `AVG-FID`. We then trained a random forest model on this dataset and computed feature importance values using SHAP.

Results: We summarize the (f_j, g) pairs with the largest `AVG-FID` in Table 5. As expected, we see that Algorithm 1 does not find any significant `AVG-FID` for this baseline case. `AVG-FID` is not exactly zero, which is expected because `AVG-FID` is measured as an absolute value, meaning that any difference in feature importance due to random variation will result in a non-zero value.

J.2. Injected Case

Experimental Setup: In our second experiment, we started with the same individuals generated in the baseline case but we injected FID for a subgroup g of older, hispanic individuals consisting of approximately 13% of the population. For this second dataset, if an individual was in g, then we generated $y \sim x_1 + 50 * x_2 + x_3 + \eta, \eta \sim \mathcal{N}(0, 1)$. Otherwise, y was generated as in the baseline case. Also as in the baseline scenario, we used a random forest model with SHAP as the feature importance notion.

Results: As seen in Table 5, the three features with the highest FID subgroups found were `Hispanic`, `age`, and x_2. Finding x_2 was expected, but it is not unusual for `Hispanic` and `age` to also be found since in our synthetic example, y is dependent on these two features for points in g and completely independent for X/g. The fourth largest `AVG-FID` found

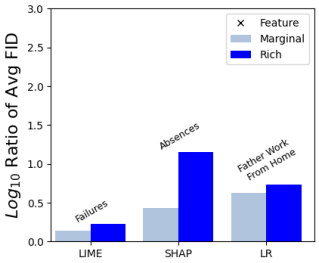
Table 5: Summary of the top (f_j, g) pairs found for the two synthetic dataset experiments. $\mu(F)$ is the average feature importance value on the specified group. We can see that in the baseline experiment, there was very little AVG−FID. In the injected case, Algorithm 1 found very large AVG−FID subgroups on the three features which were effected by the injection. The next largest pair in the injected case had AVG−FID comparable to the baseline case.

Experiment	Feature f_j	$\lvert g \rvert$	$\mu(F(f_j, X))$	$\mu(F(f_j, g))$	AVG−FID
Baseline	x_3	.13	$-.61$	$-.48$	**.13**
	x_2	.11	-1.16	-1.23	**.06**
	x_1	.12	$-.05$	$-.01$	**.04**
Injected	Hispanic-American	.15	.86	28.0	**27.13**
	Age	.17	3.57	-7.07	**10.6**
	x_2	.15	-1.56	.22	**1.78**
	Black-American	.15	.01	.04	**.03**

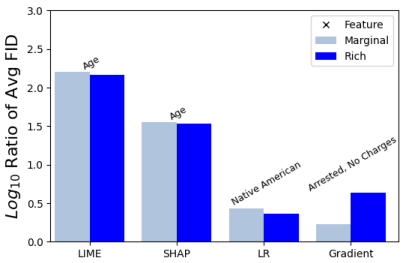
was significantly smaller than the top three and is comparable in magnitude to that of the baseline case. The subgroup found by Algorithm 1 for the top feature captured 64% of the older Hispanic subgroup where the disparity was injected. This is not a perfect result, but was obtained without extensive tuning and illustrates our method can detect an injected disparity in a controlled environment.

K. Comparison of FID Values on Rich vs. Marginal Subgroups

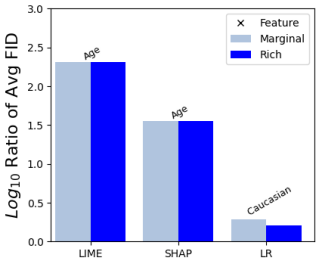
This appendix provides expanded information from Section 5.4. Here we are justifying the use of rich subgroups by searching for maximal AVG−FID subgroups in the marginal subgroup space. Marginal subgroups are those defined by a single sensitive characteristic making them straightforward to search. In Figure 6, we compare the maximal AVG−FID rich subgroups shown in Figure 2 to the maximal AVG−FID marginal subgroup for the same feature. In about half the cases, expanding our subgroup classes to include rich subgroups defined by linear functions of the sensitive attributes enabled us to find a subgroup that had a higher AVG−FID. In the other cases, the AVG−FID of the marginal subgroup was similar to the rich subgroup. Sometimes, the marginal subgroup outperformed the rich subgroup; this happens when using rich subgroups does not offer any substantial advantage over marginal subgroups, and the empirical error tolerance in Algorithm 1 stopped the convergence early.



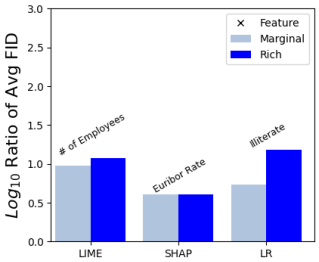
(a) Student



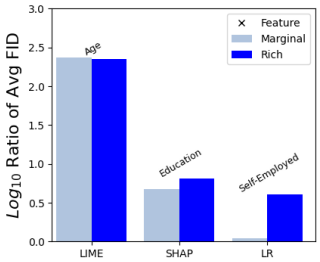
(b) COMPAS R



(c) COMPAS D



(d) Bank



(e) Folktables

Figure 6: Comparison of the maximal FID rich subgroups from Figure 2 to the maximal FID marginal subgroup on the same feature. This is displayed as $|log_{10}(R)|$ where R is the ratio of average importance per data point for separable notions and the ratio of coefficients for the linear coefficient notion. The feature associated with the subgroups is written above each bar.

L. Statistical Validity of Results: Generalization of FID and $|g|$

When confirming the validity of our findings, there are two potential concerns: (1) Are the subgroup sizes found in-sample approximately the same on the test set and (2) do the FID's found on the training set generalize out of sample? Taken together, (1) and (2) are sufficient to guarantee our maximal AVG-FID values generalize out of sample.

In Figure 7, we can see that when we take the maximal subgroup found for each feature f_j, g_j^*, and compute it's size $|g_j^*|$ on the test set, for both the separable and non-separable methods it almost always fell within the specified $[\alpha_L, \alpha_U]$ range; the average difference in $|g_j^*(X_{train})|$ and $|g_j^*(X_{test})|$ was less than .005 on all notions of feature importance and all datasets except for Student, which was closer to .025 due to its smaller size. A few rare subgroups were significantly outside the desired α range, which was typically due to the degenerate case of the feature importance values all being 0 for the feature in question. Additional plots for all (dataset, notion) pairs are in Figures 9, 10.

In Figure 8, we compare AVG-FID(f_j, g_j^*, X_{train}) to AVG-FID(f_j, g_j^*, X_{test}), or LIN-FID in the case of the linear regression notion, to see how FID generalizes. The separable notions all generalized very well, producing very similar AVG-FID values for in and out of sample tests. The non-separable method still generalized, although not nearly as robustly, with outlier values occurring. This was due to ill-conditioned design matrices for small subgroups leading to instability in fitting the least squares estimator. In Appendix P, we investigate the robustness of the feature importance notions, evaluated on the entire dataset. We find that the coefficients of linear regression are not as stable, indicating the lack of generalization in Figure 8 could be due to the feature importance notion itself lacking robustness, rather than an over-fit selection of g_j^*.

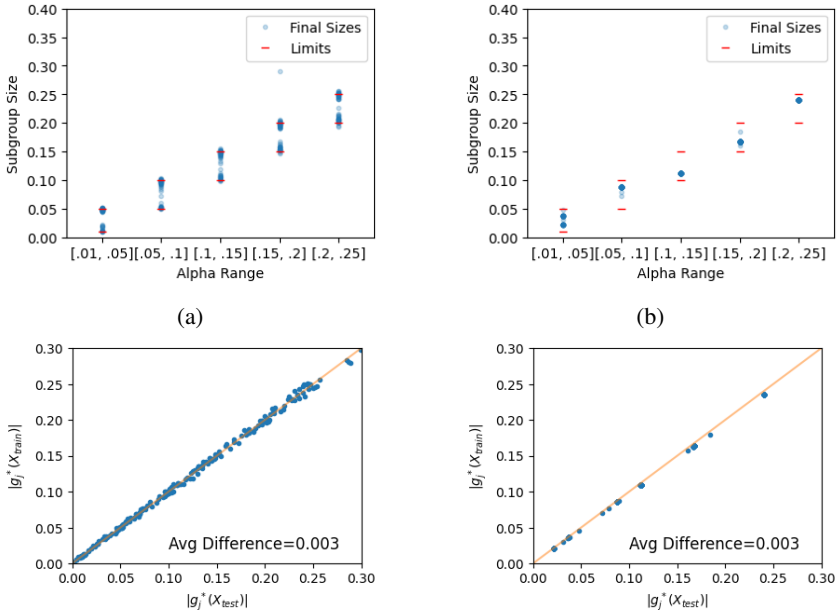


Figure 7: Generalizability of $|g|$ on the Folktables dataset. (a) Size outputs from Algorithm 1 for all features and separable notions and (b) from optimizing Equation 10 for `LIN-FID` show that our size constraints hold in-sample. (c) Plots the corresponding values of $|g_j^*(X_{train})|$ vs $|g_j^*(X_{test})|$ for separable notions and (d) for `LIN-FID`, showing that the subgroup size generalizes out of sample.

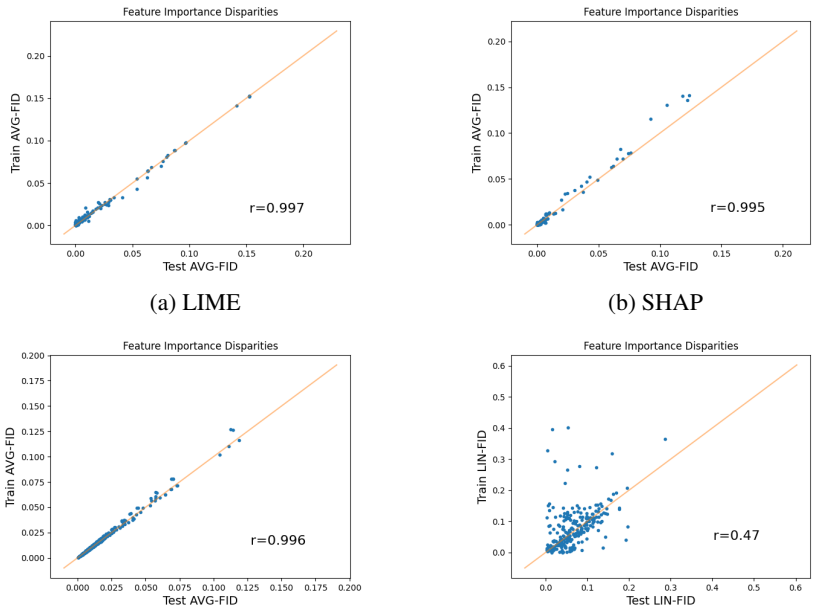


Figure 8: Out of sample generalization of the methods. Each dot represents a feature, plotting `FID` on X_{test} vs on X_{train}. All are computed on the Folktables dataset except (c) is computed on COMPAS R. The diagonal line represents perfect generalization and the Pearson correlation coefficient is displayed in figure. The non-separable approach suffers from the instability of the WLS method.

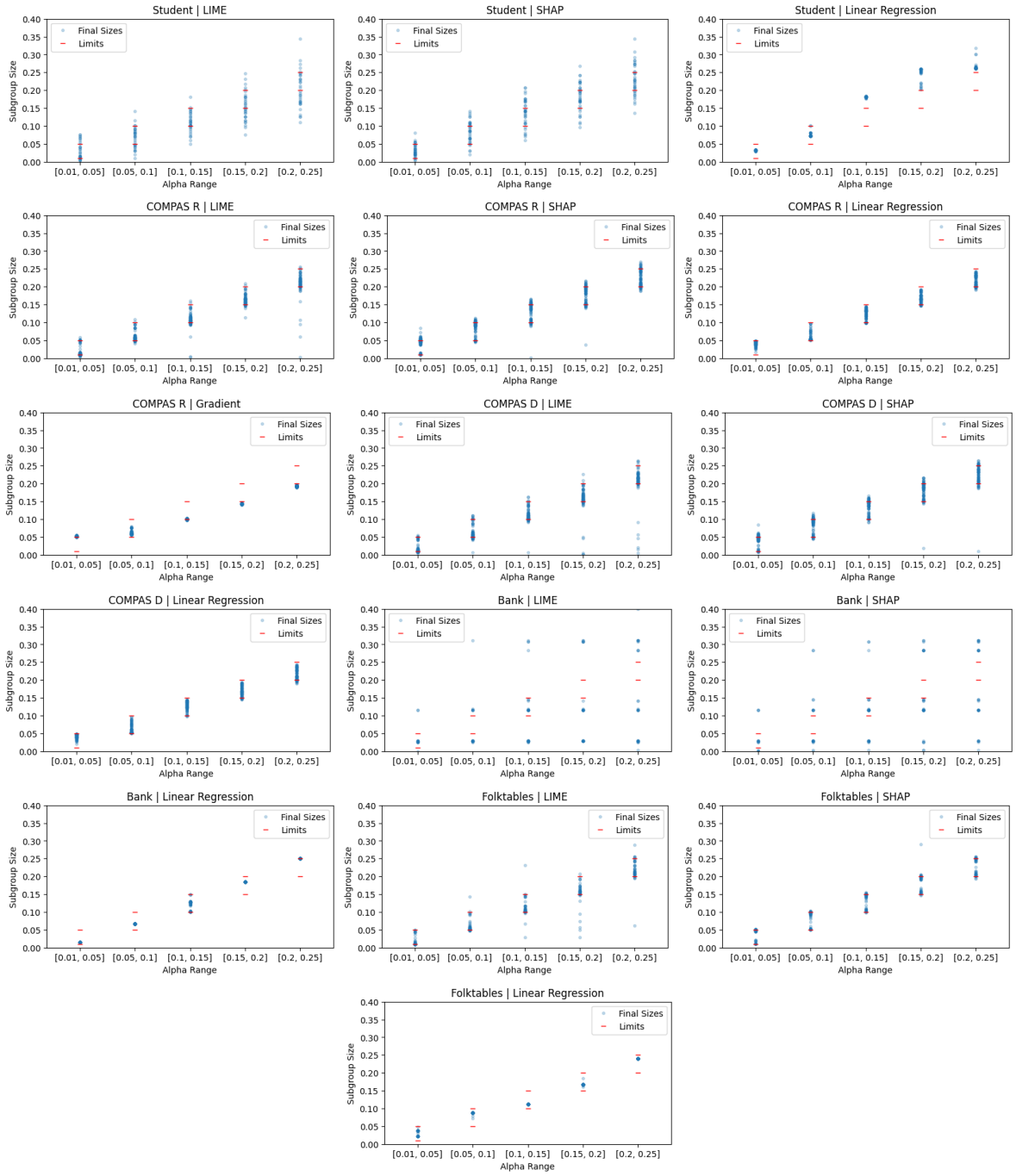
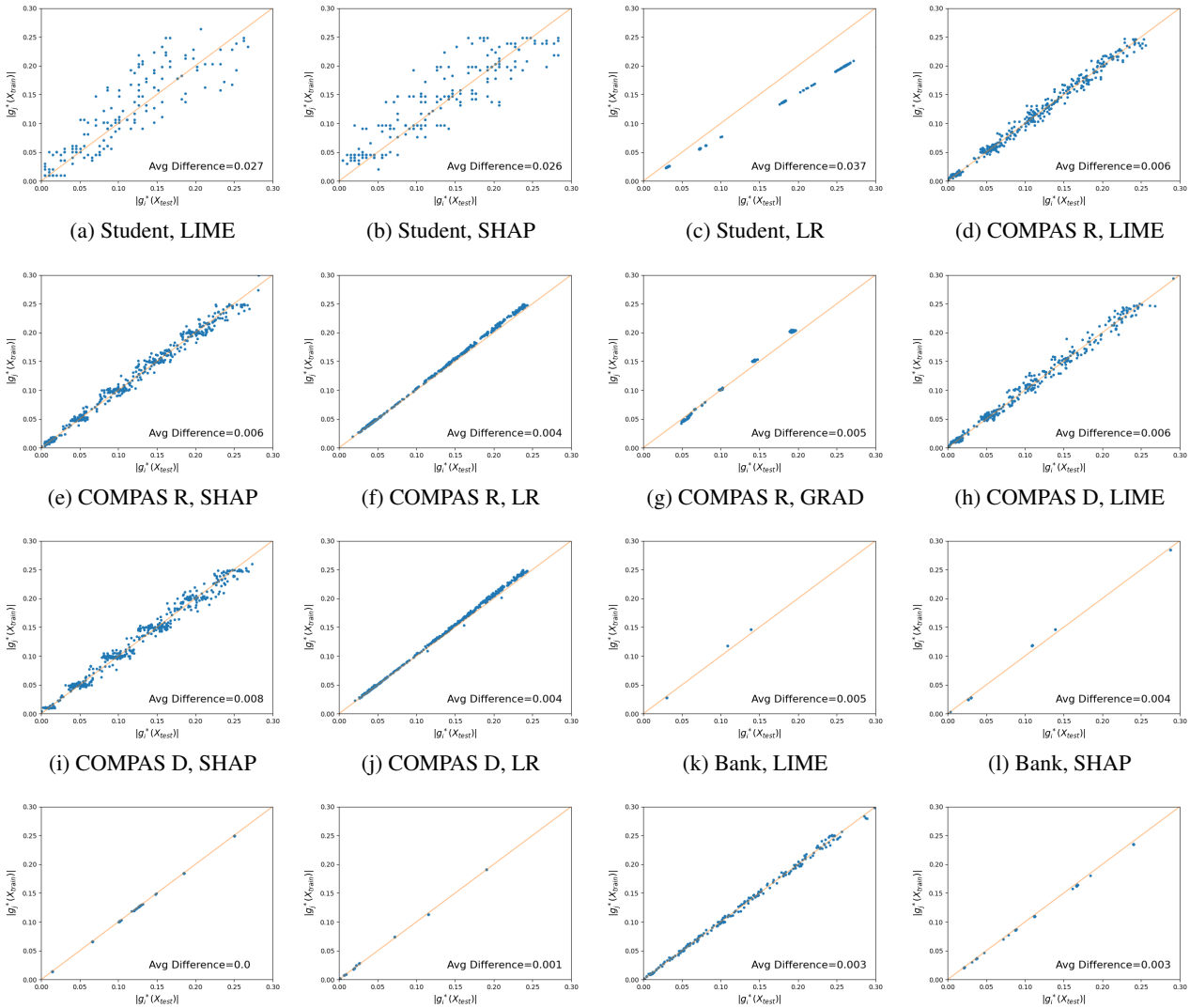


Figure 9: Final subgroup sizes of $g(X_{test})$ compared with α range. These almost always fall within the correct size range. Student has the largest errors given the small dataset size. Some subgroups fell significantly outside the expected ranged, mostly due to many importance values, $F(f_j, X, h)$, being zero for a given feature.



(a) Student, LIME (b) Student, SHAP (c) Student, LR (d) COMPAS R, LIME

(e) COMPAS R, SHAP (f) COMPAS R, LR (g) COMPAS R, GRAD (h) COMPAS D, LIME

(i) COMPAS D, SHAP (j) COMPAS D, LR (k) Bank, LIME (l) Bank, SHAP

(m) Bank, LR (n) Folk, LIME (o) Folk, SHAP (p) Folk, LR

Figure 10: Comparing $|g_j^*(X_{train})|$ and $|g_j^*(X_{test})|$. We can see that the size of the subgroup was consistent between the train and test set showing good generalization. The average difference was only noticeable on the Student dataset, due to its smaller size.

Table 6: Comparing results between using random forest and logistic regression as the hypothesis class for classifier h using LIME as the importance notion on the COMPAS R dataset. Here we display the features with the highest AVG-FID, the subgroup size $|g|$, and the AVG-FID. We can see that the choice of hypothesis class h does not substantially affect the output. We used random forest for all of our main experiments.

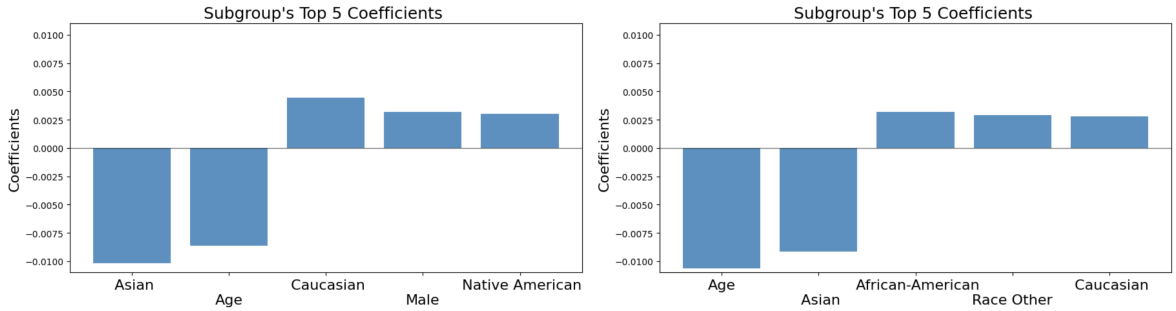
h = Random Forest			h = Logistic Regression		
Feature	Size	AVG-FID	Feature	Size	AVG-FID
Age	$.05-.1$	.144	Age	$.05-.1$	.21
Priors Count	$.01-.05$	.089	Priors count	$.01-.05$	.092
Juv Other Count	$.01-.05$	.055	Juv Other Count	$.01-.05$	.055
Other Features	-	$<.025$	Other Features	-	$<.025$

h = Random Forest			h = Logistic Regression		
Feature	Size	AVG-FID	Feature	Size	AVG-FID
Age	$.01-.05$	.4	Age	$.01-.05$	.21
Priors Count	$.01-.05$	.11	Priors count	$.01-.05$	.14
Other Features	-	$<.05$	Other Features	-	$<.05$

Table 7: Same as Table 6 except using SHAP as the importance notion. With SHAP, there were fewer features with significant AVG-FID before dropping off but in both cases, the choice of h did not significantly affect the outcome.

M. Choice of Hypothesis Class

One ablation study we explored was the choice of classification model h. While the main experiments used a random forest model, we also explored using a logistic regression model. The logistic regression model was implemented with the default sklearn hyperparameters. We found that the results are roughly consistent with each other no matter the choice of h. In Table 6 and Table 7, we see that the features with the highest AVG-FID, their subgroup sizes, and the AVG-FID values are consistent between the choice of hypothesis class. We then looked further into the features that were used to define these subgroups. In Figure 11, we see that the subgroups with high AVG-FID for the feature Age were both defined by young, non-Asians. Similarly consistent results were found across all feature importance notions and datasets.



(a) h = Random Forest, f_{j^*} = Age

(b) h = Logistic Regression, f_{j^*} = Age

Figure 11: Comparing the choice of hypothesis class of h. Here we show the defining coefficients for the highest AVG-FID subgroup found on the COMPAS R dataset using LIME as the feature importance notion. For the feature Age, we find that young and non-Asian were the two most defining coefficients for g^* no matter which choice of h.

We acknowledge that this ablation study is not the most extensive study, however given that these initial results show no significant differences between choice of hypothesis class, we decided to proceed in the main experiments to present all results using random forest as the hypothesis class. This is certainly an area that could be explored in future studies.

N. Algorithm 1 Optimization Convergence

Here are additional graphs showing examples of the convergence of Algorithm 1. Data was tracked every 10 iterations, recording the Lagrangian values (to compute the error $v_t = max(|L(\hat{p}^t_{\mathcal{G}}, \hat{p}^t_{\lambda}) - \underline{L}|, |\overline{L} - L(\hat{p}^t_{\mathcal{G}}, \hat{p}^t_{\lambda})|)$), the subgroup size, and AVG-FID value, graphed respectively in Figure 12. We can see AVG-FID value moving upward, except when the subgroup size is outside the α range, and the Lagrangian error converging upon the set error bound v before terminating.

While Theorem 1 states that convergence time may grow quadratically, in practice we found that computation time was not a significant concern. The time for convergence varied slightly based on dataset but for the most part, convergence for a given feature was achieved in a handful of iterations that took a few seconds to compute. Features which took several thousand iterations could take around 30 minutes to compute on larger datasets.

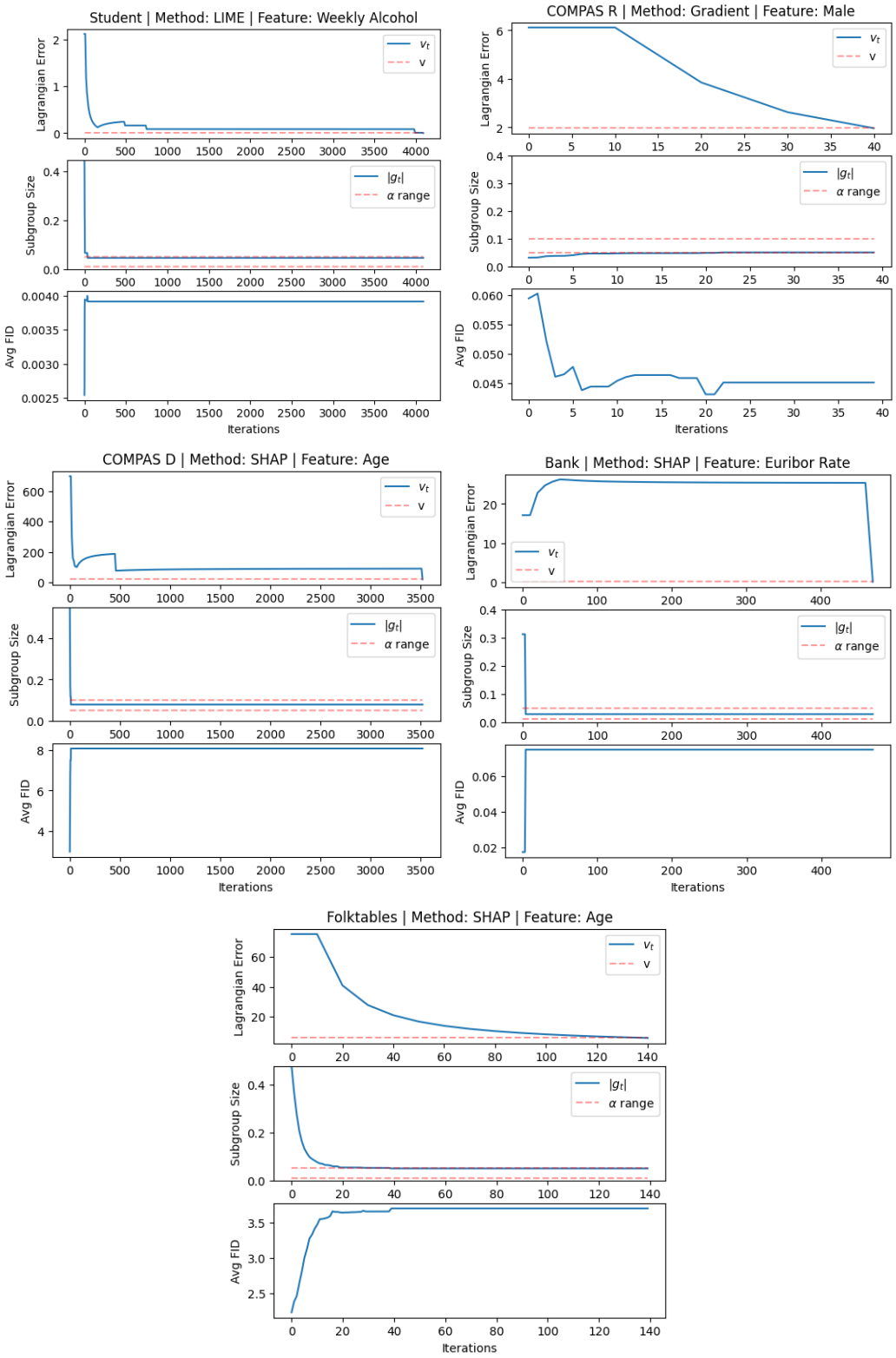


Figure 12: Plots detailing the convergence of Algorithm 1. The top plot shows the error convergence, i.e. the max difference in Lagrangian values between our solution and the min/max-players' solution. The other two plots display the subgroup size and AVG−FID of the solution. Convergence almost always happened in fewer than 5000 iterations, allaying concerns about theoretical run time.

O. Non-Separable Optimization Convergence

Here are additional graphs showing the convergence in the non-separable approach. Using the loss function that rewards minimizing the linear regression coefficient (or maximizing it) and having a size within the alpha constraints, we typically reach convergence after a few hundred iterations. In Figure 13, we can see in the respective upper graphs that the subgroup size converges to the specified α range and stays there. Meanwhile, in the lower graph, we see the `LIN-FID` attempt to maximize but oscillates as the appropriate size is found.

Convergence using this method almost always took < 1000 iterations. Running this for all features took around 2 hours to compute on the largest datasets. The optimization was run using GPU computing on NVIDIA Tesla V100s.

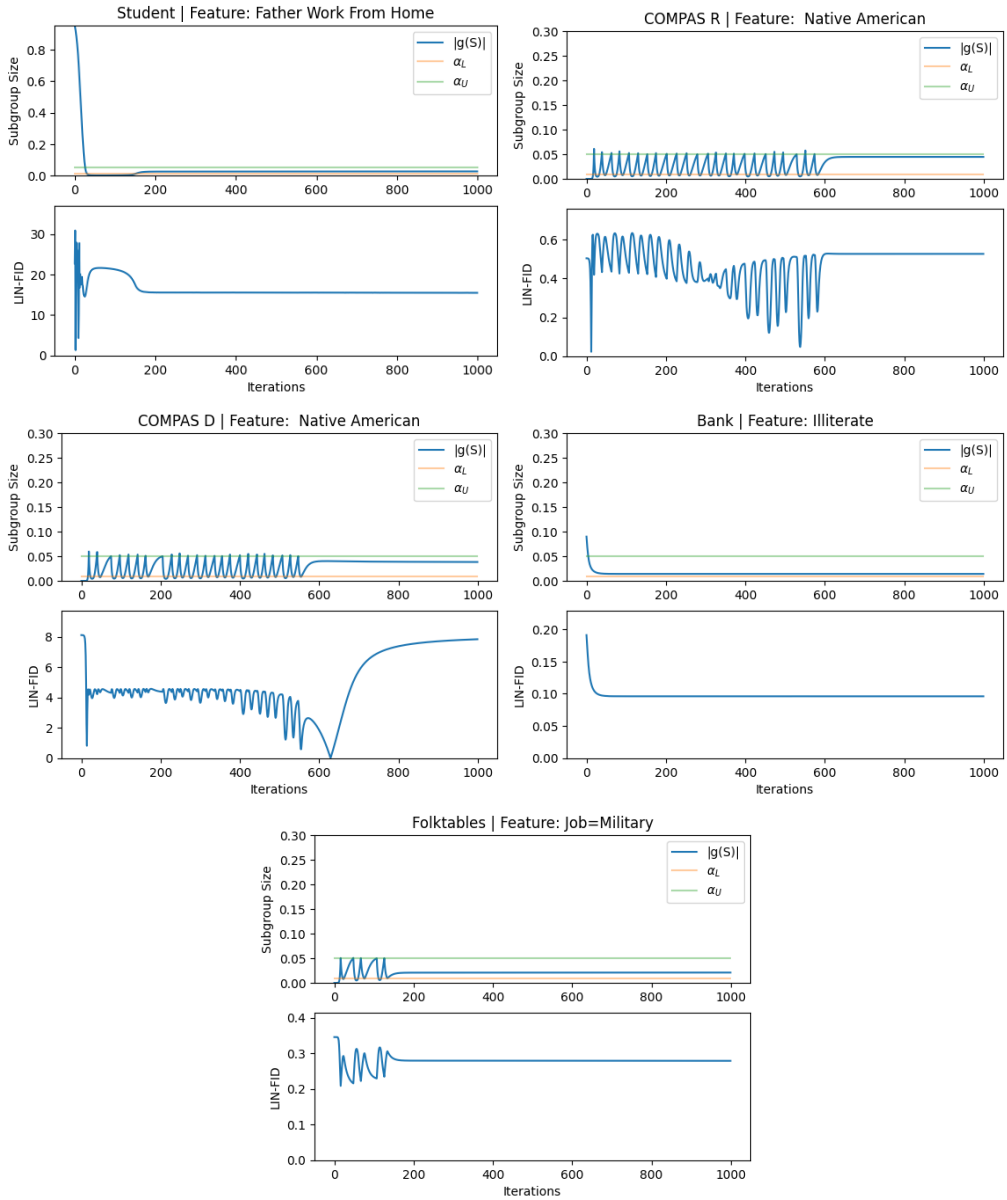


Figure 13: Plots of subgroup size and linear regression coefficient of g over the training iterations of the Adam optimizer. For each dataset, the feature with the highest `LIN-FID` was displayed.

P. Importance Notion Consistency

To see how consistent importance notion methods were, we plotted the values of $F(f_j, X_{test}, h)$ against $F(f_j, X_{train}, h)$ with each point representing a feature f_j of the COMPAS dataset. The closer these points track the diagonal line, the more consistent a method is in providing the importance values. As we can see in Figure 14, LIME and GRAD are extremely consistent. Linear regression is less consistent, due to instability in fitting the least squares estimator on ill-conditioned design matrices. SHAP is also inconsistent in its feature importance attribution, however the `AVG-FID` still generalized well as seen in Figure 8. This could mean that while SHAP is inconsistent from dataset to dataset, it is consistent relative to itself. i.e. if $F(j, X_{train}) > F(j, X_{test})$ then $F(j, g(X_{train})) > F(f, X_{test})$ meaning the `AVG-FID` value would remain the same.

These inconsistencies seem to be inherent in some of these explainability methods as noted in other research (Krishna et al., 2022; Dai et al., 2022; Agarwal et al., 2022a; Alvarez-Melis & Jaakkola, 2018; Bansal et al., 2020). Exploring these generalization properties would be an exciting future direction for this work.

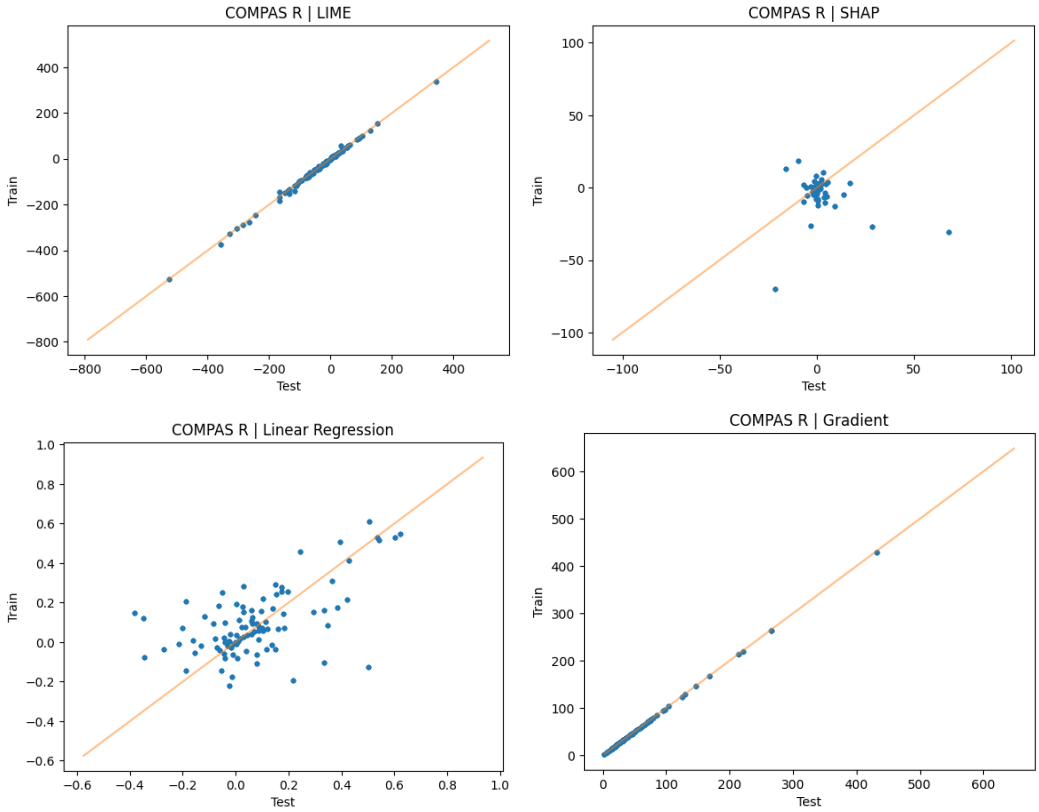


Figure 14: Consistencies of importance notions. Each point represents a feature, the x-value is $F(j, X_{test})$, and y-value is $F(j, X_{train})$. The closer the points are to the diagonal, the more consistent the notion is.

Q. Ethical Review

There were no substantial ethical issues that came up during this research process. The datasets used are all publicly available, de-identified, and have frequently been used in fair machine learning research. There was no component of this research that sought to re-identify the data or use it in any fashion other than to test our methodology.