Fair Classification with Partial Feedback: An Exploration-Based Data Collection Approach

Vijay Keswani [*][1] **Anay Mehrotra** [*][2] **L. Elisa Celis** [2]

Abstract

In many predictive contexts (e.g., credit lending), true outcomes are *only* observed for samples that were positively classified in the past. These past observations, in turn, form training datasets for classifiers that make future predictions. However, such training datasets lack information about the outcomes of samples that were (incorrectly) negatively classified in the past and can lead to erroneous classifiers. We present an approach that trains a classifier using available data and comes with a family of exploration strategies to collect outcome data about subpopulations that otherwise would have been ignored. For any exploration strategy, the approach comes with guarantees that (1) all sub-populations are explored, (2) the fraction of false positives is bounded, and (3) the trained classifier converges to a "desired" classifier. The right exploration strategy is context-dependent; it can be chosen to improve learning guarantees and encode context-specific group fairness properties. Evaluation on real-world datasets shows that this approach consistently boosts the *quality* of collected outcome data and improves the fraction of true positives for all groups, with only a small reduction in predictive utility.

1. Introduction

Machine Learning (ML) classifiers are increasingly being used to aid decision-making in high-stake contexts such as credit lending, healthcare, and criminal justice. However, their real-world deployment still faces several practical challenges, including selective availability of ground truth labels, errors in collected data, and distribution shifts (Kleinberg et al., 2018; Barocas et al., 2023; Sambasivan et al., 2021). Of specific interest is the challenge of learning an accurate classifier in the *partial feedback setting* where ground truth labels are *only* observed for positively classified samples. For instance, a bank observes whether an individual repays a loan only after granting them a loan or a doctor only observes the effect of a health intervention only if it is used.

Classification models are trained using prior outcome labeled data, often under the assumption that future samples follow the same distribution as the training data. However, in partial feedback settings, the unavailability of outcomes for unobserved samples can distort the data distribution. For example, suppose at year j, a set of applicants S_j apply for a loan. A classifier (trained on past data) assesses the default risk of all applicants and accepts the applications $L_j \subset S_j$ and rejects $U_j := S_j \setminus L_j$. Following these decisions, the true default outcomes are only observed for L_j, which is then added to training data to learn future classifiers. However, since L_j can have a different distribution than S_j, the training data composed of $\{L_j\}_{j \in 1,2,\ldots}$ can misrepresent the population distribution and train erroneous classifiers. Classification errors due to partial feedback are indeed quite prevalent in practice, e.g., in lending (Pacchiano et al., 2021), assessing bail decisions (Lakkaraju et al., 2017; Kleinberg et al., 2018), and predictive policing (Ensign et al., 2018).

Issues arising from partial feedback are further compounded when decision-making processes are biased and display disparate performance across protected attributes (e.g., gender or race) (Mehrabi et al., 2021). Biases affect many applications where ML algorithms are currently employed, e.g. bail decisions (Arnold et al., 2018), loan applications (Martinez and Kirchner, 2021), and policing (Brantingham, 2017). The impact of social biases can be significant in the partial feedback setting: Revisiting the lending example, suppose a classifier f is used to predict the default risk for applicants S_j. Further, suppose that f assigns disparately higher default risk to individuals from group z_1 compared to group z_2. As a result, a relatively larger fraction of individuals from z_2 will be assigned a loan, and since we only observe default outcomes for positively classified samples, we will have more information about the risk associated with individuals from z_2 than z_1. Future classifiers trained using this labeled data will propagate, or even exacerbate, biases against z_1.

[*]Equal contribution [1]Duke University [2]Yale University. Correspondence to: Vijay Keswani <vijay.keswani@duke.edu>, Anay Mehrotra <anaymehrotra1@gmail.com>.

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

The pervasiveness of partial feedback in practice makes it necessary to study methods for *fair* data collection and training in this setting. One way of addressing this problem is by assigning positive predictions to all samples for which outcomes have not been observed in the past. This way we improve the quality of collected labeled data and, by extension, the predictive accuracy of trained classifiers. However, positive predictions often entail high-stakes decisions (e.g., giving loans or predicting disease occurrence) and *false positives* can have significant negative impact on individual and institutional utility. Hence, data collection in the partial feedback setting is a challenging task. Prior work have proposed certain solutions for this problem; however, their real-world applicability has been limited. They either (1) rely on strong assumptions about the accuracy of past decisions (De-Arteaga et al., 2018), (2) assume that sufficiently *diverse* outcomes have already been observed (Coston et al., 2021) – which is not true when past data is limited for marginalized groups, or (3) classify a large number of samples positively in the beginning to gather outcome information (Bechavod et al., 2019) – which results in large number of false positives in the initial iterations. *Given these limitations, we ask if there are robust approaches for data collection and training in the partial feedback setting that achieve (a) high cumulative and iteration-wise utility and (b) low disparate impact across demographic groups?*

Our Contributions

We study the problem of data collection for accurate classification in the partial feedback setting (Section 2). Our proposed framework (Algorithm 1) operates in an *iterative setting*, where in each iteration a set of unlabeled samples are given as input, and our framework uses the exploitation-exploration paradigm to predict the outcomes for these samples. Using the classifiers trained in all previous iterations, we first identify the "exploit" region, i.e., the part of the domain where accurate outcome information is available, and use the trained classifier to make predictions for samples from this region. The region beyond the "exploit" region is called the "explore" region and we provide *a family of exploration strategies* to sample elements from this region that are also predicted as positive. Using the classifiers from previous iterations, we ensure a high utility over the "exploit" region, and by positively classifying certain samples from the "explore" regions, we collect outcome information about individuals and groups which would have otherwise been ignored. Fairness mechanisms can be incorporated in both the "exploit" and "explore" parts of our algorithm to ensure performance parity for all groups defined by given protected attributes (Section 3). An important aspect of our framework is the necessity to have high utility in every iteration. This is motivated by applications, e.g. lending settings, where high costs of erroneous decisions implicitly constrain the decision-maker to have a small number of false positives (FRED; FRS). To that end, our framework includes false discovery rate constraints guaranteeing that the expected number of false positives among the samples classified positively is small. Theoretically, we show that our approach always satisfies the specified bound on the false discovery rate (Theorem 4.1). Furthermore, for all groups, the prediction utility is at least as high as the previous iteration (Theorem 4.2); i.e., performance improvement through data collection. Finally, we show that, due to exploration, the predictions of our approach converge to the predictions of an "optimal classifier" (Theorem 4.3). Empirical analysis on the Adult Income and German Credit datasets further demonstrates that our proposed framework results in improved performance for all relevant groups as more data is collected (Section 5). The loss in the cumulative utility and utility per iteration due to additional exploration is minimal and the performance disparities across protected attribute groups are reduced.

Related Works

Certain recent works tackle the problem of partial feedback by proposing solutions that either collect additional data or modify the learning process to effectively utilize available outcome-labeled data. Most of these works, however, can be impractical for relevant real-world applications. For outcome exploration, Bechavod et al. (2019) propose a strategy that uses initial iterations for exploration by positively classifying all incoming samples and, in the following iterations, exploits the collected data to learn a classifier. Only exploring during the initial iterations, however, leads to a large number of false positives and low utility in these iterations. For instance, in the loan setting, a bank would be unlikely to adopt a strategy that gathers information at the cost of huge losses in certain years. In contrast, our framework performs both exploration and exploitation at every iteration, limiting the number of false positives per iteration. Wei (2021) formulates a dynamic programming framework to find a threshold-based classifier that balances exploration and exploitation. Kilbertus et al. (2020) similarly propose stochastic decision-making policies that assign a non-zero likelihood of selection to every point in the domain. Both these works adjust the learned classification policy to implicitly explore additional samples. Our framework instead employs explicit data collection strategies that are more effective in improving the rate of learning (as we demonstrate in our empirical analysis in Section 5). Yang et al. (2022) forward a bandit-type approach that uses bounded exploration to gather additional outcome data every iteration. However, their method requires non-trivial parametric assumptions on the feature distribution. Rateike et al. (2022) develop an online process that first learns an unbiased representation of the data and then trains an online classifier over the learned representation space. Similar to the papers mentioned above, this approach also does not employ any constraint on false positives which can lead to low utility in certain iterations when sufficient information for learning is unavailable (see Section 5 for empirical comparison against these methods). Data collection frameworks proposed in the above works aim to *eventually* collect a sufficient amount of data through exploration so that long-term prediction utility is high. However, as we discuss in the following sections, this approach often comes at the cost of low short-term or iteration-wise utility and, hence, can be inappropriate for real-world applications.

Discussions on the comparison of our work to other relevant approaches from the fields of active learning, fair classification, and classification using selective labels are presented in Appendix B.

2. Model, Stakeholders, and Classification

Let $D := \mathcal{X} \times \{0,1\} \times [p]$ be a finite-sized domain, where $\mathcal{X}$ is the set of features, $\{0,1\}$ is the set of labels, and $[p]$ is the set of protected attributes. (Extensions to continuous domains and multi-class settings are discussed in Section 4 and Appendix A respectively.) Let μ be the true data distribution over D and $\mathcal{F} \subseteq \{0,1\}^{\mathcal{X}\times[p]}$ be a hypothesis class of binary classifiers. For any distribution η, we use $\Pr_\eta[\cdot]$ and $\mathbb{E}_\eta[\cdot]$ to denote $\Pr_{(X,Y,Z)\sim\eta}[\cdot]$ and $\mathbb{E}_{(X,Y,Z)\sim\eta}[\cdot]$ respectively. The goal is to find a classifier with the highest *utility*, where the utility definition is context-specific, e.g. it could denote predictive accuracy or *revenue*. Hence, we consider a family of utility metrics.

Definition 2.1 (Utility Metrics). Given tuple $\gamma := (\gamma_{00}, \gamma_{01}, \gamma_{10}, \gamma_{11})$, the utility of $f \in \mathcal{F}$ w.r.t. μ is $\mathrm{Util}_\mu(f,\gamma) := \sum_{i,j\in\{0,1\}} \gamma_{ij} \cdot \Pr_\mu[f(X,Z)=i, Y=j]$.

For a given γ, the goal is to solve $\max_{f\in\mathcal{F}} \mathrm{Util}_\mu(f,\gamma)$. By using different coefficients for different kinds of predictions we can capture a wide variety of utility metrics (Elkan, 2001). For $\gamma_{\mathrm{acc}} = (1,0,0,1)$, $\mathrm{Util}_\mu(\cdot,\gamma_{\mathrm{acc}})$ is proportional to standard predictive accuracy and for $\gamma_{\mathrm{pos}} = (0,1,0,1)$, $\mathrm{Util}_\mu(\cdot,\gamma_{\mathrm{pos}})$ is the fraction of positive predictions. A metric relevant to our setting is *revenue*. It is the weighted sum of false positive and true positive predictions: given $c_1, c_2 > 0$, let $\gamma_{\mathrm{rev}} := (0,-c_1,0,c_2)$, then $\mathrm{revenue}_{c_1,c_2,\mu}(f) := \mathrm{Util}_\mu(f,\gamma_{\mathrm{rev}}) \cdot$ (number of samples). Here, c_1 represents the absolute value of loss incurred for making a false positive error and c_2 represents profit acquired for a true positive prediction.

Group-specific performance and classifier *fairness* can be measured with conditional utility: for group z, define $\mathrm{Util}_\mu(f,\gamma,z) := \mathrm{Util}_{\mu|Z=z}(f,\gamma)$. E.g., for $\gamma_{\mathrm{tpr}} = (0,0,0,1/\Pr[Y{=}1 \mid Z{=}z])$, $\mathrm{Util}_\mu(f,\gamma_{\mathrm{tpr}},z)$ denotes the true positive rate (TPR) for group z. Performance disparity of f across groups $z_0, z_1 \in [p]$ can then be quantified as $|\mathrm{Util}_\mu(f,\gamma,z_0) - \mathrm{Util}_\mu(f,\gamma,z_1)|$, i.e., absolute difference between group-wise utilities. With $\gamma = \gamma_{\mathrm{pos}}$, this denotes acceptance rate disparity (or *statistical rate* (Celis et al., 2019a)) and with $\gamma = \gamma_{\mathrm{tpr}}$, this denotes TPR disparity.

Partial feedback, false discovery rate, and optimal offline classifier. We consider the *partial feedback* setting, where true outcome labels are only observed for samples that were positively classified in the past. While the usual goal of classification is to ensure high predictive accuracy, in the partial feedback setting, there is an additional goal to gather information about unobserved samples. A trivial approach to data collection is to positively classify all samples and then observe the true outcomes. While this would lead to rich data collection, it will also have poor classifier utility. Applications involving high-stakes decisions, e.g. credit lending, usually attempt to make as few false positive predictions as possible. This is because the losses due to false positives are often much larger than profits from true positives (in the $\mathrm{revenue}_{c_1,c_2}(\cdot)$ metric defined above, this is characterized by $c_1 > c_2$). For such applications, it is necessary to limit the number of false positive errors made which can be encoded using the *false discovery rate*.

Definition 2.2 (False-discovery Rate Constraint). For any $\alpha \in (0,1]$, $f \in \mathcal{F}$ is said to satisfy α-false-discovery rate constraint (or α-FDR) if $\Pr_\mu[Y=0 \mid f(X,Z)=1] \leq \alpha$.

FDR captures the fraction of false positives among the samples classified positively. When losses associated with false positives are larger in magnitude than the profits associated with true positives, having a high FDR can lead to potentially negative utility. Hence, using an appropriate *non-trivial* FDR constraint in our framework can ensure that utility per iteration is lower bound by a positive amount. For a given α and γ, the goal of classification with partial feedback is to converge to the optimal offline classifier f^α_{opt}, where f^α_{opt} is the classifier with the maximum utility w.r.t. true distribution μ subject to α-FDR constraint:

$$f^\alpha_{\mathrm{opt}} := \operatorname*{argmax}_{f\in\mathcal{F}} \mathrm{Util}(f,\gamma), \text{ s.t., } f \text{ satisfies } \alpha\text{-FDR}. \quad (1)$$

Stakeholders and iterative model. Our setting is iterative: At each iteration $t \in \{1,2,\dots\}$, an institution needs to make predictions about a (new) set of $n \in \mathbb{N}$ individuals. E.g., suppose each year a bank must make predictions for a fresh set of loan applicants. Before the first iteration, the institution had a decision-making process in place that it used to make past decisions. This could either be just human decision-makers or a classifier f_0. We assume that the labeled samples L_0 (i.e., samples predicted positively in the past) and unlabeled samples U_0 from the past (i.e., samples predicted negatively in the past) are available. If the past predictions were made by humans, then one can

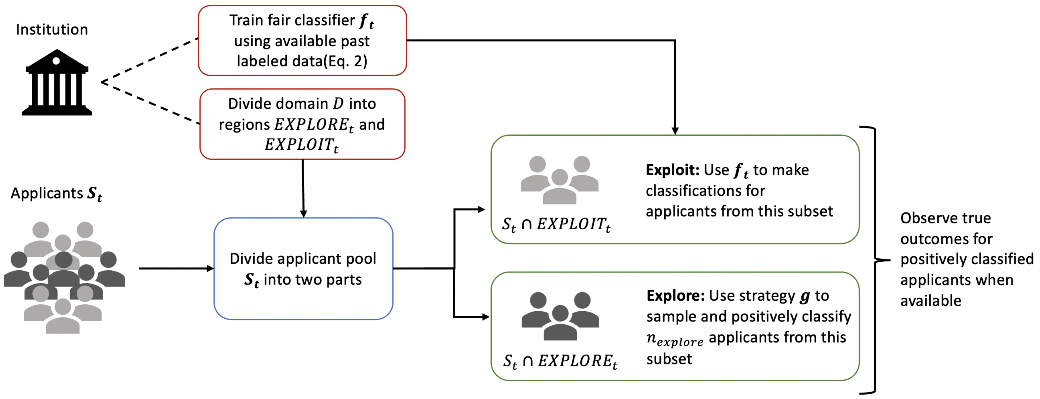
Institution
Train fair classifier f_t using available past labeled data(Eq. 2)
Divide domain D into regions $EXPLORE_t$ and $EXPLOIT_t$
Applicants S_t
Divide applicant pool S_t into two parts
$S_t \cap EXPLOIT_t$
Exploit: Use f_t to make classifications for applicants from this subset
$S_t \cap EXPLORE_t$
Explore: Use strategy g to sample and positively classify $n_{explore}$ applicants from this subset
Observe true outcomes for positively classified applicants when available

Figure 1. Pipeline of the process undertaken at time-step t to classify unlabeled samples S_t. The institution learns a classifier f_t using past labeled data. It also creates exploitation-exploration partitions to decide which elements in S_t will be classified using f_t and which elements will compose the exploration set, over which the exploration strategy g will be employed.

train a classifier f_0 to simulate human decision-making (by simulating the partition between L_0 and U_0). Importantly, we make the following minimal assumptions on f_0.

Assumption 2.3. *We assume $f_0 \in \mathcal{F}$ is (α, λ)-feasible: a classifier f is (α, λ)-feasible if (1) f satisfies the α-FDR constraint; and (2) there exists a constant $\lambda \in [0, 1]$ such that* $\Pr_\mu[f(X, Z) = 1] \geq \lambda$.

We require Assumption 2.3 (1) to ensure that α-FDR is satisfied in the first few iterations (when our predictions are similar to f_0) and Assumption 2.3 (2) to prove concentration bounds on the FDR of f_0.

The features-label pairs at the t-th iteration correspond to a set S_t of n i.i.d. samples from μ. However, the institution only has access to features $X_t := S_t|_{\mathcal{X}}$ and not the labels. After making predictions $\widehat{y}_x$ for each $x \in X_t$, the institution observes the labels of all positively classified samples, i.e., the institution observes $\{y \mid (x, y) \in S_t \text{ and } \widehat{y}_x = 1\}$. This process "partitions" S_t into a labeled set $L_t := \{(x, y) \mid (x, y) \in S_t \text{ and } \widehat{y}_x = 1\}$ and an unlabeled set $U_t := \{x \mid (x, y) \in S_t \text{ and } \widehat{y}_x = 0\}$. Sets $(L_i, U_i)_{i=1}^t$ can be used for prediction in future iterations. Note that the predictions $\widehat{y}$ should satisfy the α-FDR constraint so that the utility per iteration is high. Prior works do not satisfy this constraint and, hence, often have low iteration-wise utility (as observed in Section 5).

3. Our Framework

As mentioned earlier, our approach for simultaneous data collection and prediction is designed to handle the partial feedback setting. To do so, at each iteration t, we partition the domain D into two regions: $\textsc{Exploit}_t$ (initialized to be empty before the first iteration) and $\textsc{Explore}_t$ (defined as $D \setminus \textsc{Exploit}_t$). Region $\textsc{Exploit}_t$ contains all points for which outcome labels have been observed *sufficiently* many times in the past: concretely, each point (x, z) has a weight $w_t(x, z)$–which is proportional to the number of times its outcome has been observed in the previous iterations–and (x, z) is included in $\textsc{Exploit}_t$ if $w_t(x, z) > \tau$ (where τ is a fixed pre-specified threshold). For any (x, z), $w_t(x, z)$ never decreases from one iteration to the next, and hence, points are only added to the exploitation region and removed from the exploration region. Figure 1 demonstrates our workflow. The pseudocode of our approach is presented in Algorithm 1 and we next describe its various components.[1]

Given FDR parameter α as input, fix any $\alpha_{\text{exploit}} \in (0, \alpha)$. At iteration t, Algorithm 1 receives an unlabeled dataset S_t.

Learning (Step 3). First, Algorithm 1 learns a classifier f_t that maximizes the utility over $\textsc{Exploit}_t$ subject to satisfying α_{exploit}-FDR constraint and making sufficiently many positive predictions. One difficulty in learning f_t is that the empirical distribution over $\textsc{Exploit}_t$ may not be an unbiased estimate of the true distribution μ (as certain samples are over-represented due to past-decisions). We correct this by optimizing the utility with respect to a re-weighted distribution η_w which ensures unbiasedness with respect to μ. Intuitively, η_w is a product of three terms: an indicator ensuring the support of η_w is $\textsc{Exploit}_t$, (b) $\mu(x, z)$, and (c) $\Pr_\mu[Y{=}y|X{=}x, Z{=}z]$. If these terms are known exactly, then one can show that, by chain rule of probability, η_w is the same as μ on $\textsc{Exploit}_t$. Algorithm 1 uses estimates of these terms, which we show is good enough in our theoretical analysis (e.g., Eq. 4 generalization bound). This optimization can be solved using standard cost-sensitive classification

[1] Note that occurrences of D in Algorithm 1 can be replaced by S_t if the domain D is extremely large in practice.

methods (Appendix D).

Exploitation (Step 4-5). Next, Algorithm 1 uses f_t to predict the labels for samples in $S_t \cap \textsc{Exploit}_t$. Further, by design, if n_{exploit} samples are positively predicted in this step, then there are at most $n_{\text{exploit}} \cdot \alpha_{\text{exploit}}$ false positives.

Exploration (Steps 6-8). The exploration region consists of samples for which sufficient outcome information is not available. To determine which samples from $\textsc{Explore}_t$ region are positively predicted, we use function g, which we call the *exploration strategy*. Concretely, Algorithm 1 draws a "certain" number of elements from $\textsc{Explore}_t$, with sampling probability of any point (x, z) being proportional to $g(x, z; f_t)$ and predicts a positive label for these elements.

Observation and region update (Steps 9-12). Finally, Algorithm 1 observes the true outcome labels of positively-predicted samples. In practice, this observation step entails checking if each positively classified individual satisfies certain requirements, e.g., whether a loan is paid back within two years (see Section 6 for more discussion on this point). Before the next iteration, we also update the exploitation region to include points (x, z) whose weight $w_{t+1}(x, z) = \sum_{1 \le i \le t} g(x, z; f_i)$ now exceeds τ (observe that $w_{t+1}(x, z)$ is proportional to the number of times the label of (x, z) has been observed in the first t iterations).

Two aspects of the exploration step that need further description are (a) the number of samples, n_{explore}, from $\textsc{Explore}_t$ region that are positively classified, and (b) the exploration strategy g. Regarding (a), Algorithm 1 sets $n_{\text{explore}}{=}n_{\text{exploit}} \cdot {}^{(\alpha - \alpha_{\text{exploit}})}/_{1-\alpha}$. This choice allows us to control the number of false positives and satisfy α-FDR. Roughly, even if all n_{explore} samples from the explore region are false positives, the combined number of false positives from the exploration and exploitation steps is at most $\alpha \cdot n_{\text{exploit}} \cdot {}^{(1-\alpha_{\text{exploit}})}/_{1-\alpha}$, which ensures α-FDR since the total positive predictions is $n_{\text{exploit}} \cdot {}^{(1-\alpha_{\text{exploit}})}/_{1-\alpha}$. See Theorem 4.1 for formal proof of this feasibility claim.

Regarding (b), Algorithm 1's only requirement for g is that it take positive values so that all points have a positive probability of being observed. When no information is available about the samples in the $\textsc{Explore}_t$, the obvious choice for g is the uniform distribution. However, that is rarely the case in practice: while $\textsc{Exploit}_t$ and $\textsc{Explore}_t$ could differ in group composition and will likely belong to different underlying feature-label distributions, there can be some similarities across different groups that would allow classifiers trained on $\textsc{Exploit}_t$ to be partially predictive on $\textsc{Explore}_t$. E.g., say in a loan application setting, suppose two applicants from different groups have very high credit scores. Even if $\textsc{Exploit}_t$ contains data from only one group, information about the high credit score applicant in one group can be used to judge that high credit score applicants from all groups have low default risk. In such cases, classifier f_t can be utilized for exploration as well by, say, choosing $g^{\text{clf}}(x, z; f_t) \propto \beta {+} (1{-}\beta) f_t(x, z)$. Parameter $\beta \in [0, 1]$ will depend on the expected accuracy of classifier f_t over samples in $\textsc{Explore}_t$.

Algorithm 1 Data Collection and Prediction Framework

Input: Hypothesis class $\mathcal{F}$ with $f_0 \in \mathcal{F}$, $\alpha > 0$, labeled dataset L_0, and unlabeled data stream $S_0, S_1, \ldots, S_T$. Constants $\alpha_{\text{exploit}} \in (0, \alpha)$, $\varepsilon, \tau, \lambda \in (0, 1]$, and exploration strategy g.

```
1:  Initialize $\textsc{Exploit}_1 = \emptyset$ and $\textsc{Explore}_1 = D$
2:  for $t = 1, 2, \ldots, T$ do
        Learn:
3:      If $t = 1$, set $f_t = f_0$ (since $\textsc{Exploit}_1 = \emptyset$), otherwise
            $f_t = \arg\max_{h \in \mathcal{F}} \ \text{Util}_{\eta_w}(h, \gamma),$    (2)
            s.t., $\Pr_{\eta_w}[h = 1] \ge \lambda - \varepsilon,$
                  $\Pr_{\eta_w}[h \ne y \mid h = 1] \le \alpha_{\text{exploit}} + \varepsilon.$
        Where $\eta_w(x, y, z)$ is a density proportional to the product of three terms: (1) the indicator $\mathbb{I}[(x, z) \in \textsc{Exploit}_t]$, (2) the probability that $(x, z)$ is in $\bigcup_{i=0}^{t} S_i$ and (3) the probability that $(x, z)$ is in $\bigcup_{i=0}^{t-1} L_i$.
        Exploit:
4:      $\widehat{L}_t := \{(x, z) \in S_t \cap \textsc{Exploit}_t \mid f_t(x, z) = 1\}$
5:      $n_{\text{exploit}} = |\widehat{L}_t|$
        Explore:
6:      Set $n_{\text{explore}} = (\alpha - \alpha_{\text{exploit}} - \varepsilon) \cdot n_{\text{exploit}} / (1 - \alpha)$
7:      $\forall\ (x, z) \in S_t \cap \textsc{Explore}_t$, set $p(x, z) \propto g(x, z; f_t)$
8:      Sample $n_{\text{explore}}$ points from $S_t \cap \textsc{Explore}_t$ using distribution $p$ and add sampled points to $\widehat{L}_t$
        Observation and Region Update:
9:      For each $(x, z) \in \widehat{L}_t$, observe label $y$. Create $L_t = \{(x, z, y) | (x, z) \in \widehat{L}_t, y$ is $(x, z)$'s observed outcome$\}$
10:     Initialize $\textsc{Exploit}_{t+1} = \textsc{Exploit}_t$
11:     For each $(x, z) \in \textsc{Explore}_{t-1}$, such that $w_{t+1}(x, z) = \sum_{1 \le i \le t} g(x, z; f_i) > \tau$, add $(x, z)$ to $\textsc{Exploit}_{t+1}$
12:     Set $\textsc{Explore}_{t+1} = D \setminus \textsc{Exploit}_{t+1}$
13: end for
```

Fairness in exploration. While exploration strategy $g = g^{\text{clf}}$ can allow for improved exploration utility, it can exacerbate social biases if f_t is biased towards favoring certain groups. To reduce performance disparity across groups, it is therefore important to explore the outcomes of individuals from marginalized groups at an increased rate. This can be accomplished by choosing g in a manner that takes into account the proportions of different groups. E.g., $g^{\text{fair}}(x, z; f_t) {\propto} g^{\text{clf}}(x, z; f_t) \cdot \Pr_\mu[Z = z | (X, Z) \in \textsc{Explore}_t]$ would use classifier out-put to improve utility while ensuring that every group's selection rate is close their proportion in $\textsc{Explore}_t$. Hence, groups that are under-represented in $\textsc{Exploit}_t$, compared to their population proportion, will be explored at a higher rate. See Appendix A for other choices and discussion of g.

Fairness in exploitation. A common approach to mitigate biases in classification is to use constraints during learning that require performance disparity across groups to be small (Celis et al., 2019a). Our framework can incorporate such fairness mechanisms by constraining the classifier trained in Step 3 using any common fairness metric. Fairness constraints in exploitation may be necessary when Algorithm 1 is implemented using a biased data source (which is the case in Section 5 simulations). In such cases, despite re-weighting, the distribution used for training in Step 3 of Algorithm 1 can still misrepresent marginalized groups. However, just using fairness constraints during exploitation is not sufficient as it only ensures fairness over $\textsc{Exploit}_t$ (which may not represent the entire domain). Hence, it should be used along with exploration fairness. As observed in Section 5, group disparity is smallest when fairness is incorporated in both exploit steps and explore steps.

4. Theoretical Results

We next present the theoretical guarantees of our framework; all proofs are provided in Appendix C. The sample complexities in our results depend on "how uniform g is." One way to capture this is via the following parameter:

$$\sigma := (\min_{(x,z)\in D} g(x,z))/(\textstyle\sum_{(x,z)\in D} g(x,z)). \quad (3)$$

σ is non-zero as $g > 0$. It is minimized when g approaches 0 at some $(x,z) \in D$, and it is maximized when g is the uniform distribution over D. Our first result shows that Algorithm 1 satisfies the specified FDR constraint.

Theorem 4.1 (Feasibility w.r.t. FDR constraint). *Suppose f_0 is (α,λ)-feasible (Assumption 2.3). For any $\varepsilon,\delta,\tau \in (0,1]$, Algorithm 1 satisfies the following at every iteration t: If $n \geq |D| \cdot \mathrm{poly}\left(1/\lambda,\ 1/\tau,\ 1/\min\{\varepsilon,\alpha-\varepsilon\}\right) \cdot \log\left(|D|/\sigma\delta\right)$, then the predictions made in the t-th iteration satisfy the α-FDR constraint with probability at least $1-\delta$ w.r.t. the randomness in $S_1, S_2, \ldots, S_t$ and Algorithm 1.*

Theorem 4.1 holds for any exploration strategy g which takes positive values. In line with other works using exploration for data collection (e.g., Wei (2021)), Theorem 4.1 requires n to be linear in $|D|$. This dependence can be improved by making additional assumptions on μ: for instance, if μ is "smooth," then one can reduce the sample complexity from $|D|$ to C, where C is the minimum number clusters that achieve a "high prediction accuracy" (see Theorem 19.3 of Shalev-Shwartz and Ben-David (2014) and Appendix A).

Our next result shows that the group-wise utility of the classifiers learned by our framework increases at every iteration. The results holds for hypothesis classes $\mathcal{F}$ where each hypothesis $f \in \mathcal{F}$ is a tuple of p "classifiers" $f = (f_1, f_2, \ldots, f_p)$, one for each group. Here, each classifier f_z belongs to some *base hypothesis class* $\mathcal{B} \subseteq \{0,1\}^{\mathcal{X}}$ (e.g., set of linear classifiers) and we say $\mathcal{F}$ is *derived* from $\mathcal{B}$.

Theorem 4.2 (Fairness: Improvement in group-wise utility). *Suppose f_0 is (α,λ)-feasible (Assumption 2.3) and $\mathcal{F}$ is derived from $\mathcal{B} \subseteq \{0,1\}^{\mathcal{X}}$. For any $\varepsilon,\delta,\tau \in (0,1]$ and tuple γ, Algorithm 1 satisfies the following at every iteration t and $z \in [p]$: If $n \geq |D| \cdot \mathrm{poly}\left(1/\lambda, 1/\tau, 1/\min\{\varepsilon,\alpha-\varepsilon\}\right) \cdot \log\left(|D|/\sigma\delta\right)$, then with probability at least $1-\delta$,*

$$Util_{\mu,t}(f_t,\gamma,z) \geq \max_{0\leq i\leq t-1} Util_{\mu,t}(f_i,\gamma,z) - \varepsilon.$$

Where $Util_{\mu,t}(f,z)$ is the utility of f over draws $(X,Y,Z) \sim \mu$ conditioned on $(X,Z) \in \textsc{Exploit}_t$ and $Z = z$. The randomness at the t-th iteration is w.r.t. the randomness in $S_1, \ldots, S_t$ and Algorithm 1.

Theorem 4.2 holds for any Util and shows that, with high probability, f_t achieves a higher utility for each group than any previously learned classifier. At the t-th iteration, the utility in Theorem 4.2 is measured w.r.t. $\textsc{Exploit}_t$ (since f_t is only used to make predictions for samples in $\textsc{Exploit}_t$).

Our final theoretical result shows that f_t's utility converges to f_{opt}'s utilitys as $t \to \infty$ since, for any t, f_t is "accurate" on $\textsc{Exploit}_t$ and, as $t \to \infty$, $\textsc{Exploit}_t$ converges to D. Intuitively, this is true because Algorithm 1 observes the true labels of all samples with a positive probability. Convergence-rate for group z depends on τ, $\alpha_{\mathrm{explore}}$, and $\sigma(z)$. Where $\sigma(z) := \min_{x\in\mathcal{X}} g(x,z)/(\sum_{(x,z)\in D} g(x,z))$ is the fraction of mass that function g assigns to group z.

Theorem 4.3 (Group-wise convergence to $f_{\mathrm{opt}}^{\alpha}$). *Suppose f_0 is $(\alpha-\varepsilon,\lambda)$-feasible. For any $\alpha,\varepsilon,\delta,\tau \in (0,1]$ and $\alpha_{\mathrm{exploit}} = \alpha-\varepsilon$, $\alpha_{\mathrm{explore}} = \varepsilon$, Algorithm 1 satisfies the following: if $t \geq 1/\sigma(z)$ and $n \geq |D| \cdot \mathrm{poly}\left(1/\lambda, 1/\tau, 1/\min\{\varepsilon,\alpha-\varepsilon\}\right) \cdot \log\left(|D|/\sigma\delta\right)$ then with probability at least $1-\delta$, the utility of classifier f_t learned by the framework in t-th iteration is at least as large as the utility of $f_{\mathrm{opt}}^{\alpha}$ on samples in the z-th group drawn from μ, i.e.,*

$$Util_{\mu}\left(f_t,\gamma,z\right) \geq Util_{\mu}\left(f_{\mathrm{opt}}^{\alpha},\gamma,z\right) - \varepsilon.$$

Where the randomness at the t-th iteration is w.r.t. the randomness in $S_1, S_2, \ldots, S_t$ and Algorithm 1.

The convergence rate for the z-th group increases with $\sigma(z)$ and, hence, choosing g that explores samples in z-th group with higher probability improves the convergence rate on the z-th group. This may be desirable in some contexts to address historical biases (see Section 3 and Appendix A). Finally, the following convergence bounds are a corollary of Theorem 4.3: if $n \geq |D| \cdot \mathrm{poly}\left(1/\lambda, 1/\tau, 1/\min\{\varepsilon,\alpha-\varepsilon\}\right) \cdot$

Table 1. Comparison of all methods on the Adult (race) and German (gender) datasets. We report the avg. revenue per iteration (standard error in brackets), avg. FDR, and avg. acceptance rate disparity (statistical rate). Parameter details are provided in Figure 2, 3 captions.

	Adult - Protected attribute: Race				German - Protected attribute: Gender			
Method	Revenue (in thousands)	FDR	Stat. Rate	TPR Disparity	Revenue (in thousands)	FDR	Stat. Rate	TPR Disparity
Alg. 1 - no fairness constraint	71.5 (12.6)	.15 (.02)	.02 (.02)	.08 (.05)	6.4 (4.2)	.11 (.07)	.07 (.06)	.08 (.07)
Alg. 1 - only exploit fairness	73.0 (12.0)	.15 (.02)	.02 (.01)	.08 (.05)	8.3 (4.1)	.11 (.05)	.04 (.04)	.06 (.05)
Alg. 1 - only explore fairness	74.7 (12.0)	.15 (.02)	.03 (.02)	.07 (.06)	9.4 (4.3)	.11 (.05)	.07 (.05)	.08 (.07)
Alg. 1 - both fairness constraints	74.1 (12.2)	.15 (.02)	.01 (.01)	.06 (.04)	8.8 (4.1)	.11 (.05)	.05 (.04)	.06 (.06)
Baseline - OPT-OFFLINE	75.7 (9.9)	.14 (.02)	.03 (.02)	.12 (.08)	9.7 (2.7)	.15 (.04)	.15 (.04)	.15 (.05)
Baseline - FAIR-CLF	71.1 (1.4)	.14 (.02)	.03 (.02)	.11 (.08)	8.7 (3.9)	.13 (.05)	.05 (.05)	.06 (.06)
KILBERTUS ET AL.	44.9 (14.1)	.12 (.03)	.03 (.02)	.09 (.07)	9.7 (5.4)	.24 (.03)	.11 (.09)	.09 (.08)
YANG ET AL.	-14.7 (16.4)	.35 (.07)	.12 (.04)	.12 (.06)	-1.7 (9.7)	.29 (.04)	.08 (.14)	.06 (.13)
RATEIKE ET AL.	-17.2 (8.1)	.12 (.01)	.02 (.01)	.02 (.01)	2.2 (0.8)	.25 (.01)	.04 (.02)	.05 (.02)

$\log\left({}^{|D|}/_{\sigma\delta}\right)$ and $t \geq {}^{1}/_{\sigma}$, then with probability at least $1-\delta$, $\text{Util}_\mu\left(f_t,\gamma\right) \geq \text{Util}_\mu\left(f^\alpha_{\text{opt}},\gamma\right) - \varepsilon$. I.e., f_t's utility is at least as large as that of f^α_{opt} on the samples drawn from μ.

Key proof technique. The technical core of our analysis is a generalization bound (Lemma C.1) showing that the reweighted distribution η_w in Step 3 of Algorithm 1 is a good approximation of μ on $\textsc{Exploit}_t$. We show that if $n \geq |D| \cdot \text{poly}\left({}^{1}/_{\lambda}, {}^{1}/_{\tau}, {}^{1}/_{\min\{\varepsilon,\alpha-\varepsilon\}}\right) \cdot \log\left({}^{|D|}/_{\sigma\delta}\right)$, then at any iteration t and for any bounded function $h\colon \{0,1\} \times \{0,1\} \times [p] \to [0,1]$ the following holds: with probability at least $1-\delta$, for all $f \in \mathcal{F}$

$$\left| \begin{array}{l} \mathbb{E}_{\eta_w|(X,Z)\in\textsc{Exploit}_t}\left[h(f(X,Z),Y,Z)\right] \\ -\ \mathbb{E}_{\mu|(X,Z)\in\textsc{Exploit}_t}\left[h(f(X,Z),Y,Z)\right] \end{array} \right| \leq O(\varepsilon). \quad (4)$$

At any t, η_w is a product of three terms (see Step 3 of Algorithm 1): (a) an indicator ensuring the support of η_w is $\textsc{Exploit}_t$, (b) an estimate of $\mu(x,z)$, and (c) an estimate of $\Pr_\mu[Y=y|(X,Z)=(x,z)]$. If the estimates in (b) and (c) are exact, then Equation (4) follows. We prove that, with probability $\geq 1-\delta$, both estimates are nearly-correct for all $(x,z) \in \textsc{Exploit}_t$ with $\mu(x,z) > {}^{\varepsilon}/_{|D|}$. This implies that with probability $\geq 1-\delta$, $\eta_w(x,y,z) \in (1 \pm O(\varepsilon))\mu(x,y,z)$ for any $(x,z) \in \textsc{Exploit}_t \setminus R$ and $y \in \{0,1\}$, where $R \coloneqq \{(x,z) \mid \mu(x,z) \leq {}^{\varepsilon}/_{|D|}\}$. That is, η_w is a good approximation of μ on $\textsc{Exploit}_t \setminus R$ – Equation (4) follows as $\mu(R) \leq \varepsilon$. The proof of estimate (b)'s accuracy is via the Chernoff-bound. For (c), if we show that $\bigcup_{i=0}^{t-1} L_i$ has at least $k_0 \coloneqq \text{poly}\left({}^{1}/_{\varepsilon}\right) \cdot \log\left({}^{|D|}/_{\delta}\right)$ copies of each $(x,z) \in \textsc{Exploit}_t \setminus R$, then bound on estimate (c) follows. Fix any $(x,z) \in \textsc{Exploit}_t \setminus R$. Let N_i be the number of copies of (x,z) in L_i. If $N_1, N_2, \ldots$ are independent, then existing concentration inequalities imply the claim. The main challenge is that independence may not hold as Algorithm 1's choices depend on past observations. To overcome this, we show that $N_1, \ldots, N_T$ are mutually independent, where T is the last iteration when (x,z) is in the exploration region (as before, observed labels for (x,z) do not affect η_w). Appendix C provides a detailed proof.

5. Empirical Results

5.1. Adult Income Dataset

We first evaluate our framework over the new Adult Income dataset which contains demographic and financial data of around 251k individuals from California (Ding et al., 2021). The task is to predict whether an individual's annual income is above \$50k. We use race and gender as protected attributes. The dataset is almost evenly divided w.r.t. gender, and for race, we limit the dataset to White (93%) and Black/African-American (7%) individuals. See Appendix E for feature and pre-processing details. We present results for race in this section and results for gender are deferred to Appendix E. We test our algorithm over 40 iterations and the dataset is randomly split into 40 equal parts. The first part, denoted by S_0, is used to construct the initial dataset, and the i-th part is the input for the i-th iteration. We create an initial dataset using S_0 that simulates real-world biased data settings. To do so, S_0 is divided into a labeled subset L_0 and an unlabeled subset U_0, with L_0 containing 90% samples with class label 1 and only 10% samples with class label 0, and $U_0 {=} S_0 \setminus L_0$. L_0 represents decisions made by prior biased mechanisms (e.g., biased human decision-makers) that primarily accepted samples for which accurate decisions could be made. An initial logistic classifier (f_0) is trained to simulate these prior decisions. Assigning samples in L_0 with a dummy label of 1 and samples in U_0 with a dummy label of 0, we train f_0 to simulate past decision boundaries.

We use constrained logistic regression with adjusted thresholds to compute f_t at any iteration t. Utility is measured using the $\text{revenue}_{c_1,c_2}(\cdot)$ metric with $c_1 = 500$ and $c_2 = 200$. The FDR constraint parameter $\alpha = 0.15$. Performance for different α and implementation details for the optimization program are provided in Appendix E. We perform 50 repetitions using a random dataset split in each repetition.[2]

[2] Link to code: http://github.com/vijaykeswani/Fair-Classification-with-Partial-Feedback/

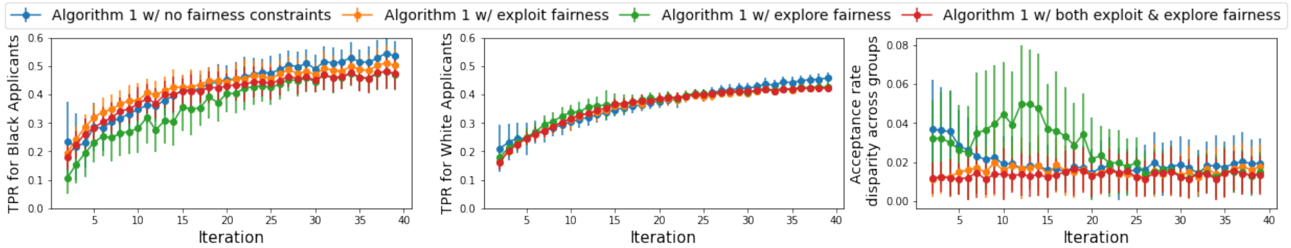

Figure 2. Iteration-wise performance of Algorithm 1 (with explore and/or exploit fairness) on Adult (protected attribute is race). Parameters $\alpha = 0.15$ with $\alpha_{\text{exploit}} = 0.075 \cdot t^{0.2}$ and $\alpha_{\text{explore}} = \alpha - \alpha_{\text{exploit}}$, $\tau = 0.5, \lambda = 0, \varepsilon = 10^{-3}$.

Fairness constraints. As fairness can be incorporated in both the exploration and exploitation components of Algorithm 1, we obtain four variants of our algorithm: *(a) no fairness constraints, (b) only exploit fairness, (c) only explore fairness*, and *(d) both exploit and explore fairness constraints*. For exploit fairness, we use the statistical rate constraint; i.e., constrain the absolute difference between acceptance rates across groups (see the last paragraph in Section 3). When explore fairness is not used – variants (a), (b) – the exploration function g is set to be g^{clf} and when explore fairness is used – variants (c), (d) – g is set to be g^{fair}. Functions g^{clf}, g^{fair} are described in Section 3; see Appendix E for details of sampling using these functions.

Baselines. We compare our approach to the following baselines: (a) KILBERTUS ET AL (Kilbertus et al., 2020), which uses stochastic classifiers to assign a non-zero exploration probability to every sample; (b) YANG ET AL (Yang et al., 2022), which employs a bandit-type approach, first determining the likelihoods using a logistic model and then adjusting classifier thresholds to incorporate gathered information; (c) RATEIKE ET AL (Rateike et al., 2022), which learns an unbiased representation of the data using which an online decision-making model is trained; (d) OPT-OFFLINE, i.e. the ideal (*unattainable* in partial feedback setting) classifier trained using i.i.d. samples from μ; (e) FAIR-CLF, which implements a classifier, with statistical parity and FDR constraints, that is trained every iteration using the available labeled data. Implementation details are provided in Appendix E. We report the mean and standard error of revenue, FDR, statistical rate, and TPR disparity across protected attributes. Iteration-wise (i.e., for each t, evaluate f_t over S_{t+1} using above metrics) and cumulative performances are reported to determine short-term and long-term utilities.

Results. Table 1 presents the cumulative performance of our algorithms for the Adult dataset. The table shows that all variants of Algorithm 1 achieve high average cumulative revenue while satisfying the given FDR constraint. Fairness constraints also have an impact on revenue and fairness. Using either explore or exploit fairness leads to an increase in cumulative revenue. This is because both constraints increase the selection of qualified minority group individuals, which leads to improved revenue. In terms of fairness, Algorithm 1 achieves a low statistical rate for all variants. Additional fairness is not necessary to achieve a small statistical rate here because the minority group form only 10% of the dataset. Hence, any amount of non-trivial exploration can improve the selection rate for this group. However, using both explore and exploit fairness leads to the lowest average TPR disparity, showing that additional fairness can be useful in gathering accurate information.

Figure 2 presents the iteration-wise performance of our algorithms. The first two plots show that TPR increases with increasing iterations for all groups. TPR is also larger in the initial iterations when using exploit fairness, but is overtaken by or is similar to the TPR of Algorithm 1 with no fairness constraints in the later iterations. Hence, fairness constraints accelerate data collection in the initial iterations, but once sufficient data is available, it seems to have a similar TPR as the variant with no constraints. The third plot in Figure 2 also shows that using exploit fairness results in the smallest statistical rate. With gender as the protected attribute, fairness constraints have a larger impact in reducing these outcome disparities (see results in Appendix E).

Comparison to baselines. From Table 1, we can see that all variants of Algorithm 1 have slightly smaller revenue than OPT-OFFLINE baseline. This is expected since OPT-OFFLINE is trained using samples from the dataset distribution, which is unavailable in the online setting we operate in. When using fairness constraints, our algorithms also achieve lower statistical rates and TPR disparities than OPT-OFFLINE, addressing the biases in the underlying dataset. Algorithm 1 also outperforms the KILBERTUS ET AL and YANG ET AL in terms of revenue, statistical rate, and TPR disparity. In particular, the revenue for these methods is low because of the high number of false positive errors. For RATEIKE ET AL, FDR, statistical rate, and TPR disparity are small, however, the cumulative revenue achieved is also quite low. This is because their algorithm results in a large number of false positives in initial iterations and, hence, extremely low revenue in those iterations. By using FDR constraints, we ensure that the number of false positive errors is small in every iteration. Algorithm 1 also has better revenue than the FAIR-CLF baseline, which, due to lack of any explicit exploration, gathers outcome information at a slower rate

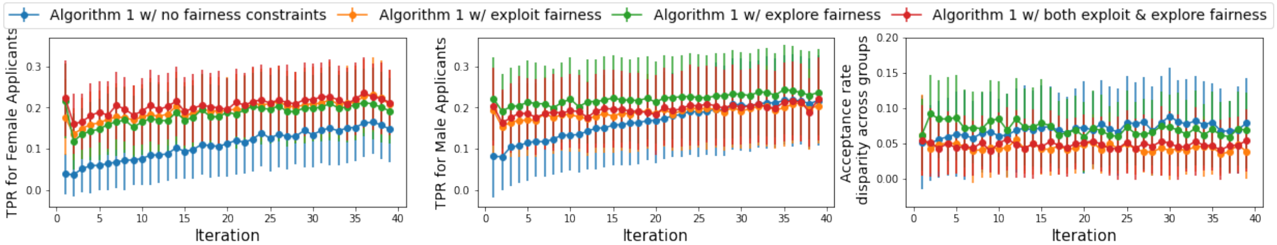

Figure 3. Iteration-wise performance of Algorithm 1 (with explore and/or exploit fairness) on German (protected attribute is gender). Parameters $\alpha = 0.15$ with $\alpha_{\text{exploit}} = 0.075 \cdot t^{0.2}$ and $\alpha_{\text{explore}} = \alpha - \alpha_{\text{exploit}}$, $\tau = 0.5, \lambda = 0, \varepsilon = 10^{-3}$.

than our framework. For additional assessments of our approach, Appendix E presents iteration-wise comparison to OPT-OFFLINE, FAIR-CLF, and KILBERTUS ET AL.

5.2. German Credit Dataset

Next, we evaluate the performance over the German Credit dataset (Hofmann, 1994) which contains entries of individuals who have taken credit and the task is to decide whether credit risk associated with them is *good* or *bad*. We use gender as the protected attribute (69% men, 31% women). Features and pre-processing details are provided in Appendix E. The original dataset contains only 1000 entries. Since this is not sufficient for testing our framework, we sample with replacement 500 entries from the dataset to serve as input S_t for iteration t. Other implementation details are similar to Section 5.1. The utility is measured using revenue$_{c_1,c_2}(\cdot)$ with $c_1 = -500$ and $c_2 = 200$.

Results. Table 1 presents cumulative performance. Algorithm 1 with only explore fairness and Algorithm 1 with both explore and exploit fairness constraints achieve higher average revenue than other variants. Algorithm 1 with only exploit fairness constraints also achieves lowest statistical rate and TPR disparity but the fairness of all variants is within one standard deviation of each other. Hence, using fairness constraints ensures both low disparity and high revenue. Figure 3 presents iteration-wise performance and shows that average TPR increases as we gather more information. Using fairness constraints also leads to high TPR for both groups implying that they accelerate data collection. Statistical rate is smallest when exploit fairness is used.

Comparison to baselines. Table 1 further shows that cumulative revenue using Algorithm 1 is smaller than that of OPT-OFFLINE and KILBERTUS ET AL baselines. However, these algorithms perform worse in terms of fairness and FDR and lead to relatively larger statistical rates and TPR disparities. Algorithm 1 with explore fairness or with both fairness constraints also achieve higher revenue and similar fairness as FAIR-CLF and RATEIKE ET AL. Overall, the differences between our algorithm and baselines are relatively smaller for this dataset (compared to Adult) since the dataset size is much smaller here. Yet, we still see certain improvements due to exploration.

6. Discussion and Conclusion

We provide a framework for data collection and learning that obtains high prediction utility in every iteration and gathers outcome data for previously unobserved samples. Our framework ensures that false positive errors are bounded and employs fairness mechanisms for improved exploration of under-represented groups. We next discuss certain practical advantages. An expanded discussion on exploration strategies, distribution and outcome shifts, multi-class classification, and limitations is presented in Appendix A.

Short-term gains and long-term utility. An important advantage of our framework is that it uses both exploitation and exploration strategies in every iteration. Using available data to make accurate predictions over the exploit regions, we ensure that the utility is high in every iteration. And by randomly selecting samples in the explore region, we gather data that improves prediction utility in later iterations. We also provide exploration strategies that can lead to higher utility over the explore region by using the available classifier (e.g. $g = g^{\text{fair}}$ leads to better utility over the Adult dataset). With appropriate choices, our framework shows gains in both short-term and long-term utilities.

Fairness. As discussed earlier, fairness can be incorporated by training classifier f_t to satisfy fairness constraints or by using exploration strategies that encode fairness notions. In practice, a combination of both could be the most effective way of tackling biases. Fairness constraints when learning f_t can also encode an implicit form of exploration, since they may encourage increased selection of minority individuals to satisfy the constraints. However, as noted in prior work (Kallus et al., 2020), implicit exploration using fairness constraints may not be sufficient for collecting outcome data about minority groups. In Section 5, we observe that complementing fairness constraints in exploitation with fairness strategies in exploration is the most effective solution to reduce disparities. Hence, standard fair classification performance can be improved by using additional exploration.

Acknowledgements

This project is supported in part by NSF Award IIS-2045951.

Impact Statement

Our exploration mechanism focuses on data collection for subpopulations that are under-represented in the available labeled data. The explicit focus on exploration is crucial since the under-represented subpopulations are usually not random subsets of the domain, but rather are composed of individuals from marginalized groups that have historically been denied equal opportunities. Our work allows for improved data collection from these groups. Note that, exploration and data collection can still lead to errors that hurt outcomes for some individuals who are incorrectly positively classified (e.g., negative impact on credit score if unable to pay back loan). Data collection should ideally not come at the cost of individual-level harms and corresponding steps to mitigate this harm should taken when implementing our framework in practice (e.g., through strict FDR constraints in our framework.)

References

Jacob D Abernethy, Pranjal Awasthi, Matthäus Kleindessner, Jamie Morgenstern, Chris Russell, and Jie Zhang. Active sampling for min-max fairness. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 53–65. PMLR, 17–23 Jul 2022.

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna M. Wallach. A Reductions Approach to Fair Classification. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 60–69. PMLR, 2018.

Hadis Anahideh, Abolfazl Asudeh, and Saravanan Thirumuruganathan. Fair active learning. *Expert Systems with Applications*, 199:116981, 2022. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2022.116981. URL https://www.sciencedirect.com/science/article/pii/S0957417422004055.

David Arnold, Will Dobbie, and Crystal S Yang. Racial bias in bail decisions. *The Quarterly Journal of Economics*, 133(4):1885–1932, 2018.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning: Limitations and Opportunities*. MIT Press, 2023.

Yahav Bechavod, Katrina Ligett, Aaron Roth, Bo Waggoner, and Zhiwei Steven Wu. *Equal Opportunity in Online Classification with Partial Feedback*. Curran Associates Inc., Red Hook, NY, USA, 2019.

P Jeffrey Brantingham. The logic of data bias and its impact on place-based predictive policing. *Ohio St. J. Crim. L.*, 15:473, 2017.

William Cai, Ro Encarnacion, Bobbie Chern, Sam Corbett-Davies, Miranda Bogen, Stevie Bergman, and Sharad Goel. Adaptive sampling strategies to construct equitable training datasets. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 1467–1478, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393522. doi: 10.1145/3531146.3533203. URL https://doi.org/10.1145/3531146.3533203.

L. Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K. Vishnoi. Classification with Fairness Constraints: A Meta-Algorithm with Provable Guarantees. In *FAT*, pages 319–328. ACM, 2019a.

L. Elisa Celis, Sayash Kapoor, Farnood Salehi, and Nisheeth K. Vishnoi. Controlling Polarization in Personalization: An Algorithmic Framework. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, pages 160–169, New York, NY, USA, 2019b. ACM. ISBN 978-1-4503-6125-5. doi: 10.1145/3287560.3287601. URL http://doi.acm.org/10.1145/3287560.3287601.

L. Elisa Celis, Anay Mehrotra, and Nisheeth Vishnoi. Fair classification with adversarial perturbations. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 8158–8171. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/44e207aecc63505eb828d442de03f2e9-Paper.pdf.

Jiahao Chen, Nathan Kallus, Xiaojie Mao, Geoffry Svacha, and Madeleine Udell. Fairness under unawareness: Assessing disparity when protected class is unobserved. In *FAT*, pages 339–348. ACM, 2019.

Amanda Coston, Ashesh Rambachan, and Alexandra Chouldechova. Characterizing fairness over the set of good models under selective labels. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 2144–2155. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/coston21a.html.

Maria De-Arteaga, Artur Dubrawski, and Alexandra Chouldechova. Learning under selective labels in the presence of expert consistency. *CoRR*, abs/1807.00905, 2018. URL http://arxiv.org/abs/1807.00905.

Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. *Advances in neural information processing systems*, 34:6478–6490, 2021.

Charles Elkan. The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence*, volume 17, pages 973–978. Lawrence Erlbaum Associates Ltd, 2001.

Danielle Ensign, Sorelle A. Friedler, Scott Neville, Carlos Scheidegger, and Suresh Venkatasubramanian. Runaway feedback loops in predictive policing. In *FAT*, volume 81 of *Proceedings of Machine Learning Research*, pages 160–171. PMLR, 2018.

Hortense Fong, Vineet Kumar, Anay Mehrotra, and Nisheeth K. Vishnoi. Fairness for auc via feature augmentation. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 610, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393522. doi: 10.1145/3531146.3533126. URL https://doi.org/10.1145/3531146.3533126.

FRED. Delinquency Rate on All Loans, All Commercial Banks. https://fred.stlouisfed.org/series/DRALACBN.

FRS. Board of governors of the federal reserve system, Charge-Off and Delinquency Rates on Loans and Leases at Commercial Banks. https://www.federalreserve.gov/releases/chargeoff/.

Thomas Gramespacher and Jan-Alexander Posth. Employing explainable ai to optimize the return target function of a loan portfolio. *Frontiers in Artificial Intelligence*, 4: 693022, 2021.

Hans Hofmann. German credit dataset, 1994. https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data).

Nathan Kallus and Angela Zhou. Residual unfairness in fair machine learning from prejudiced data. In *International Conference on Machine Learning*, pages 2439–2448. PMLR, 2018.

Nathan Kallus, Xiaojie Mao, and Angela Zhou. Assessing algorithmic fairness with unobserved protected class using data combination. In *FAT**, page 110. ACM, 2020.

Feiyang Kang, Hoang Anh Just, Anit Kumar Sahu, and Ruoxi Jia. Performance scaling via optimal transport: Enabling data selection from partially revealed sources. In A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 61341–61363. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/c142c14699223f7417cad706fd6f652e-Paper-Conference.pdf.

Vijay Keswani and L. Elisa Celis. Addressing strategic manipulation disparities in fair classification. In *Proceedings of the 3rd ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization*, EAAMO '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400703812. doi: 10.1145/3617694.3623252. URL https://doi.org/10.1145/3617694.3623252.

Niki Kilbertus, Manuel Gomez Rodriguez, Bernhard Schölkopf, Krikamol Muandet, and Isabel Valera. Fair decisions despite imperfect predictions. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 277–287. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/kilbertus20a.html.

Jon Kleinberg, Himabindu Lakkaraju, Jure Leskovec, Jens Ludwig, and Sendhil Mullainathan. Human decisions and machine predictions. *The quarterly journal of economics*, 133(1):237–293, 2018.

Himabindu Lakkaraju, Jon Kleinberg, Jure Leskovec, Jens Ludwig, and Sendhil Mullainathan. The Selective Labels Problem: Evaluating Algorithmic Predictions in the Presence of Unobservables. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 275–284, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450348874. doi: 10.1145/3097983.3098066. URL https://doi.org/10.1145/3097983.3098066.

Yunyi Li, Maria De-Arteaga, and Maytal Saar-Tsechansky. When more data lead us astray: Active data acquisition in the presence of label bias. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 10(1):133–146, Oct. 2022. doi: 10.1609/hcomp.v10i1.21994. URL https://ojs.aaai.org/index.php/HCOMP/article/view/21994.

Zhiyong Li, Xinyi Hu, Ke Li, Fanyin Zhou, and Feng Shen. Inferring the outcomes of rejected loans: an application of semisupervised clustering. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 183(2): 631–654, 2020.

Emmanuel Martinez and Lauren Kirchner. The secret bias hidden in mortgage-approval algorithms, August 2021. https://themarkup.org/denied/2021/08/25/the-secret-bias-hidden-in-mortgage-approval-algorithms.

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

Aldo Pacchiano, Shaun Singh, Edward Chou, Alex Berg, and Jakob Foerster. Neural pseudo-label optimism for the bank loan problem. *Advances in Neural Information Processing Systems*, 34:6580–6593, 2021.

Vishakha Patil, Ganesh Ghalme, Vineet Nair, and Y. Narahari. Achieving fairness in the stochastic multi-armed bandit problem. *J. Mach. Learn. Res.*, 22(1), jan 2021. ISSN 1532-4435.

Miriam Rateike, Ayan Majumdar, Olga Mineeva, Krishna P Gummadi, and Isabel Valera. Don't throw it away! the utility of unlabeled data in fair decision making. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 1421–1433, 2022.

Nithya Sambasivan, Shivani Kapania, Hannah Highfill, Diana Akrong, Praveen Paritosh, and Lora M Aroyo. "everyone wants to do the model work, not the data work": Data cascades in high-stakes ai. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380966. doi: 10.1145/3411764.3445518. URL https://doi.org/10.1145/3411764.3445518.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Amr Sharaf, Hal Daume III, and Renkun Ni. Promoting fairness in learned models by learning to active learn under parity constraints. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 2149–2156, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393522. doi: 10.1145/3531146.3534632. URL https://doi.org/10.1145/3531146.3534632.

Jie Shen, Nan Cui, and Jing Wang. Metric-fair active learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 19809–19826. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/shen22b.html.

Ramya Srinivasan and Ajay Chander. Biases in AI systems. *Communications of the ACM*, 64(8):44–49, 2021.

Serena Wang, Wenshuo Guo, Harikrishna Narasimhan, Andrew Cotter, Maya R. Gupta, and Michael I. Jordan. Robust optimization for fairness with noisy protected groups. In *NeurIPS*, 2020.

Dennis Wei. Decision-making under selective labels: Optimal finite-domain policies and beyond. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 11035–11046. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/wei21a.html.

Min Wen, Osbert Bastani, and Ufuk Topcu. Algorithms for fairness in sequential decision making. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1144–1152. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/wen21a.html.

Yifan Yang, Yang Liu, and Parinaz Naghizadeh. Adaptive data debiasing through bounded exploration. In *Advances in Neural Information Processing Systems*, 2022.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *AISTATS*, volume 54 of *Proceedings of Machine Learning Research*, pages 962–970. PMLR, 2017a.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *WWW 2017*, pages 1171–1180, 2017b.

Supplementary Material

A. Additional Discussion

In this section, we further discuss exploration strategies, distribution and outcome shifts, multi-class classification, limitations of our analysis, and broader impact.

Outcome observations. Our framework assumes that for samples predicted positively at any iteration t, the true outcomes will be observed before time $t + 1$. However, in real-world applications, outcomes for different samples might be observed at different time intervals. For instance, loan default is defined as whether a loan was repaid or not within a certain number of years. However, different people could repay at different times. We make the assumption that all outcomes are observed before the next iteration for ease of analysis. Moreover, we also assume that the data observed is accurate whereas, in practice, data inevitably has recording errors that can have adverse effects (Chen et al., 2019; Wang et al., 2020; Celis et al., 2021; Keswani and Celis, 2023). Future work can further explore methods to model settings where samples are observed at different time periods and possibly contains errors.

Other fair exploration strategies. We discussed a few exploration strategies in Section 3 that either assume no information (uniform sampling), strategies that use the classifier ($g^{\mathrm{clf}}(x, z) \propto \beta + (1 - \beta)f_t(x, z)$), and fair exploration strategies that use the classifier ($g^{\mathrm{fair}}(x, z) \propto (\beta + (1 - \beta)f_t(x, z)) \cdot \Pr_\mu [Z = z \mid (X, Z) \in \textsc{Explore}_t]$). There are many other possible choices of exploration strategies and we discuss a few other fair strategies here.

If the classifier has very low accuracy or large biases over the $\textsc{Explore}$ region, it would not make sense to use it in the exploration function. In such cases, if we still want to sample minority group individuals at a higher rate, then using the uniform sampling function would sample groups proportional to their representation in the $\textsc{Explore}_t$ region. Hence, if these groups are under-represented in the $\textsc{Exploit}_t$ region but over-represented in the $\textsc{Explore}_t$ region, they would be selected at a higher rate. Alternately, if the underlying population is demographically-imbalanced and the institution wants to select an equal number of individuals from all groups for exploration, then it can use the exploration function $g(x, z) \propto 1/ \Pr_\mu [Z = z \mid (X, Z) \in S_t]$. If the classifier does have reasonable accuracy for some $\textsc{Explore}_t$ samples and the institution wants to select an almost equal number of samples from all groups, then it can use the exploration function $g(x, z) \propto (\beta + (1 - \beta)f_t(x, z)) \cdot 1/ \Pr_\mu [Z = z \mid (X, Z) \in S_t]$.

Depending on the context and application and question, different choices for g would be relevant and useful. Nevertheless, in all settings, it is important to select a function that takes positive values and ensure all demographic groups are appropriately represented among the explored samples.

Multi-class classification. We have mainly considered binary classification settings so far. However, our proposed algorithm can be used for multi-class classification as well. Suppose the set of all class labels is $\mathcal{Y}$ and outcomes are observed only if the prediction belongs to a subset $\mathcal{Y}' \subset \mathcal{Y}$. In this case, at iteration t, Step 3 of Algorithm 1 can learn a multi-class classifier over the labeled weighted data. The remaining steps, i.e., determining the EXPLORE and EXPLOIT partitions and exploration remains unchanged. This is because computing weights for each sample (Step 6 in Algorithm 1) would once again involve determining whether at least τ fraction of previous classifiers would have predicted a label in set $\mathcal{Y}'$ - if yes, this sample is put in EXPLOIT region, otherwise put it in EXPLORE region. With these simple modifications, Algorithm 1 can be used for multi-class classification settings.

Choice of hypothesis class. The choice of hypothesis class for Algorithm 1 is important. It encodes context-specific knowledge, such as, whether individuals in different groups have similar or different distributions of features and labels. On the one hand, if the distribution of features and labels is similar across groups, then we may want classifiers to not use protected attributes for predictions. On the other hand, if the distribution of features and labels is different across groups, then we may want the classifiers to use the protected attributes for prediction. Depending on the application and available data, this choice can be made appropriately.

Sample complexity of theoretical results. Our theoretical results (Theorems 4.1, 4.2, 4.3) do not make assumptions on the underlying distribution μ, as additional assumptions may reduce the practical usefulness of the framework. This results in linear dependence on domain size $|D|$, which can be large. The difficulty in proving such bounds is that they rely on training data being sampled i.i.d. from the underlying distribution μ. However, in any non-trivial data collection framework, the specific samples collected are bound to depend on the observations made in the previous iterations. This dependence on past observations violates the independence assumption.

That said, if the distribution μ is known to satisfy additional properties, then the number of samples required by the theoretical results may be reduced. For instance, suppose the underlying distribution is "smooth," as assumed by generalization guarantees of clustering algorithms (see Chapter 19 of (Shalev-Shwartz and Ben-David, 2014). In this case, one can first cluster the samples in the domain D to obtain a set of clusters C such that almost all samples in each cluster have the same label, and then use our framework with C as the underlying domain (see Theorem 19.3 (Shalev-Shwartz and Ben-David, 2014)). In this case, the dependence on $|D|$ improves to $|C|$, which is always smaller and the ratio $\frac{|C|}{|D|}$ depends on the desired accuracy and the "smoothness" of μ.

Limitations of empirical analysis. In our empirical analysis, we assume that the applicants arriving at every iteration are sampled i.i.d. from the dataset distribution. However, the distributions for the Adult and German datasets are likely to be different than the true population distributions. Since we do not have access to the true distribution and only have access to the given dataset, we are only able to simulate the setting where the initial data available to the framework L_0 is different than the dataset distribution (i.e., treating dataset distribution as true distribution). Future simulations on settings where true underlying distribution is available will be useful to assess the complete impact of exploration.

Distribution shifts. Algorithm 1 assumes that features arriving at the beginning of every iteration, $S_1, S_2, \ldots, S_t$ sampled from the true distribution μ. In practice, this may not necessarily be true. Sample distributions can be different at different iterations and even the underlying distribution μ can change over time. Keeping α_{explore} in Algorithm 1 to be non-zero for all iterations can ensure that a certain number of samples are explored every iteration. Future work can also study and analyze other exploration strategies to find the most effective one for this setting.

Broader impact. Our framework provides a method to collect outcome information about relatively unexplored subpopulations so that classifiers trained using observed data are accurate over the entire population and not just for subpopulations for whom data is available. It is important to note that in our framework fairness is primarily ensured with respect to predefined protected attributes. Our framework might not ensure performance parity for groups that are not explicitly defined as "protected" and, hence, it would be crucial to first identify all group attributes with respect to which observation disparities exist.

Secondly, exploration in settings like credit lending comes with risks. Providing a loan to a person who cannot pay it back can be harmful to the person's financial status and severely affect their credit score. We try to minimize this possibility by incorporating the decision of classifiers learned on exploit region during exploration, but the stochasticity of exploration can still make this scenario possible. While exploration stochasticity can be beneficial to the institution in terms of collecting data about the entire population, it would also be useful to scout additional exploration strategies that minimize risk for individuals and this can be a fruitful direction for future work on this topic.

B. Other Related Work

Approaches from related fields like active learning and fair classification can be employed in the partial feedback setting, but suffer from some drawbacks that we discuss in this section.

Fairness classification. To reduce performance disparity across demographic groups, fair classification (Zafar et al., 2017a;b; Agarwal et al., 2018; Celis et al., 2019a) and multi-armed bandit (Celis et al., 2019b; Patil et al., 2021) approaches can also potentially be used. However, fairness constraints are informative of outcome bias only when evaluated over datasets whose empirical distribution is close to the underlying population distribution. Since partial feedback results in distribution shifts, the above assumption may not hold, and learning using only fairness constraints would not guarantee low disparity and high utility (Kallus and Zhou, 2018). Our framework allows for supplementing fairness constraints with additional exploration to address this issue.

Active learning. Active learning-based data collection approaches assume that labels can be queried for each sample at a *fixed cost* (Cai et al., 2022; Li et al., 2022; Abernethy et al., 2022; Shen et al., 2022; Anahideh et al., 2022; Sharaf et al., 2022). These approaches, once again, focus on achieving high long-term utility by addressing gaps in the available outcome (empirical) distribution. The main difference between our work and prior active learning approaches is the presence of the FDR constraint which ensures that iteration-wise utility is also continuously high despite exploration. Additionally, unlike the standard setup for active learning, our work assumes that labeling costs depend on the outcome and the classifier. For instance, a false positive can be significantly more costly to the institution than a false negative (as in the case of loan defaults).

The adaptive exploration approach of Yang et al. (2022) falls within the category of active learning and Section 5 demonstrates the drawbacks of this approach - empirical analysis shows that the algorithm of Yang et al. (2022) achieves significantly lower cumulative utility and higher statistical rate disparity than our framework.

Classification using selective labels. For judicial bail decisions, Kleinberg et al. (2018) train a model using past judges' bail decisions with true outcomes only available for released defendants. Unlike our work, they do not incorporate outcome information from positively classified samples into classifier training. De-Arteaga et al. (2018) suggest a similar approach for data augmentation that exploits the human decision-makers' decisions in regions where they are accurate and trains a classifier for regions where they are not. Their approach also has additional drawbacks: it assumes that all decision-makers have similar behavior, which is not true in our setting where the decision-makers are classifiers that can be different across different iterations. Extrapolation (Coston et al., 2021) or reweighting (Li et al., 2020) methods for partial feedback settings have also been proposed to impute the labels for unlabeled samples. Imputation of this form, however, assumes that all samples are appropriately represented in the labeled data distribution. This assumption does not hold in settings where historical biases have led to the under-representation of certain groups in the labeled data. Algorithms for ensuring fairness in sequential decision-making tasks – modeled as Markov decision processes (MDPs) – are also relevant in this setting (Wen et al., 2021). While these may be used for decision-making, their practical use may be limited by the assumption that all involved individuals' actions follow a known MDP.

Data augmentation. Beyond the data collection approaches discussed in Section 1, certain related methods often turn towards other sources to obtain information about outcome data of under-explored populations. This includes methods of data collection using human annotators (Li et al., 2022) or augmenting available data using third-party signals (Fong et al., 2022); however, these approaches are often impractical as they can only provide proxies for the true outcomes, which themselves can encode social biases (Srinivasan and Chander, 2021). One recent approach also studies how to distribute a specified data-collection budget among different data sources (Kang et al., 2023). In this case, the data collection takes place *before* the prediction phase and does not rely on the predictions being made. The main challenge of partial feedback setting, however, is that data collection and learning are necessarily intertwined. For instance, in the credit-lending application, banks require data on past loan repayment rates which themselves are determined by the predictions made by the bank (i.e., the loans it gives out). Prior approaches that do not account for this causal relation between prediction and data collection are bound to be either ineffective in improving prediction performance and/or wasteful in data collection.

C. Proofs

C.1. Proof of Theorem 4.1

In this section, we prove Theorem 4.1. For ease of reference, we restate Theorem 4.1 below.

Theorem 4.1 (Feasibility w.r.t. FDR constraint). *Suppose f_0 is (α, λ)-feasible (Assumption 2.3). For any $\varepsilon, \delta, \tau \in (0,1]$, Algorithm 1 satisfies the following at every iteration t: If $n \geq |D| \cdot \mathrm{poly}\left({}^{1}\!/\!_{\lambda},\, {}^{1}\!/\!_{\tau},\, {}^{1}\!/\!_{\min\{\varepsilon,\alpha-\varepsilon\}}\right) \cdot \log\left({}^{|D|}\!/\!_{\sigma\delta}\right)$, then the predictions made in the t-th iteration satisfy the α-FDR constraint with probability at least $1-\delta$ w.r.t. the randomness in $S_1, S_2, \ldots, S_t$ and Algorithm 1.*

Recall that we assume that Assumption 2.3 holds with constants $\alpha, \lambda \in (0,1]$. This implies that the classifier $f_0 \in \mathcal{F}$ satisfies the α-FDR constraint and has a selection rate of at least λ, i.e.,

$$\frac{\Pr_{\mu}\left[f_0(X,Z)=1 \text{ and } Y=0\right]}{\Pr_{\mu}\left[f_0(X,Z)=1\right]} \leq \alpha \quad \text{and} \quad \Pr_{\mu}\left[f_0(X,Z)=1\right] \geq \lambda. \tag{5}$$

Where we use $\Pr_{\mu}[\cdot]$ to denote $\Pr_{(X,Y,Z)\sim\mu}[\cdot]$ and $\mathbb{E}_{\mu}[\cdot]$ to denote $\mathbb{E}_{(X,Y,Z)\sim\mu}[\cdot]$.

To prove Theorem 4.1, given any $\varepsilon, \delta, \tau \in (0,1]$, we need to show that if n is sufficiently large, then at each iteration $t \in \{1, 2, \ldots\}$, with probability at least $1-\delta$, the data collection and prediction framework in Algorithm 1, run with the parameters α_{exploit} and α_{explore}, makes predictions that satisfy the $(\alpha_{\text{exploit}} + \alpha_{\text{explore}} + O(\varepsilon))$-FDR constraint, i.e.,

$$\frac{\sum_{(x,z)\in S_t} \mathbb{I}\left[\widehat{y}_{(x,z)} = 1 \text{ and } y = 0\right]}{\sum_{(x,z)\in S_t} \mathbb{I}\left[\widehat{y}_{(x,z)} = 1\right]} \leq \alpha_{\text{exploit}} + \alpha_{\text{explore}} + O\left(\varepsilon\right). \tag{6}$$

In particular, we will show that the following lower bound on n is sufficient

$$n \geq \frac{12\,|D|}{\tau\varepsilon^3} \cdot \frac{1}{\alpha_{\text{explore}}\lambda} \cdot \log\frac{|D|}{\delta\sigma}. \tag{7}$$

The main step in the proof is to establish the following concentration inequality.

Lemma C.1. *For any bounded function $h\colon \{0,1\}\times\{0,1\}\times[p]\to[0,1]$, any number $t\in\{1,2,\dots\}$, and any constants $\varepsilon,\tau,\delta\in(0,1]$, given $n \geq |D|\cdot \mathrm{poly}\left(1/\tau, 1/\varepsilon, 1/\lambda, 1/\alpha_{\text{explore}}\right)\cdot\log\left(|D|/(\delta\sigma)\right)$ at the t-th iteration η_w is such that the following holds*

$$\Pr\left[\forall_{f\in\mathcal{F}},\quad \left|\begin{array}{c} \mathbb{E}_{\eta_w}\left[h(f(X,Z),Y,Z)\mid (X,Z)\in \textsc{Exploit}_t\right] \\ -\ \ \mathbb{E}_{\mu}\left[h(f(X,Z),Y,Z)\mid (X,Z)\in \textsc{Exploit}_t\right]\end{array}\right| \leq \varepsilon\right] \geq 1-\delta.$$

Where η_w is the distribution defined in Step 3 of Algorithm 1, $\textsc{Exploit}_t$ is the exploitation region at the t-th iteration. The expectations and probabilities are over the randomness in the draw of $S_0, S_1, \dots, S_t$ and the randomness in Algorithm 1.

Before presenting the proof of Theorem 4.1 we require the definition of the Vapnik–Chervonenkis (VC) dimension.

Definition C.2. Given a finite set A, define the collection of subsets $\mathcal{F}_A := \{\{a\in A \mid f(a)=1\} \mid f\in\mathcal{F}\}$. We say that $\mathcal{F}$ shatters a set B if $|\mathcal{F}_B| = 2^{|B|}$. The VC dimension of $\mathcal{F}$, $\mathrm{VC}(\mathcal{F})\in\mathbb{N}$, is the largest integer such that there exists a set C of size $\mathrm{VC}(\mathcal{F})$ that is shattered by $\mathcal{F}$.

Proof of Theorem 4.1 assuming Lemma C.1. Fix any iteration $t\in\{1,2,\dots\}$. Assume $\varepsilon \leq \lambda/3$.[3] Consider the classifier f_t in Algorithm 1. Recall that Algorithm 1 uses f_t for making predictions in the exploitation region $\textsc{Exploit}_t$ and makes at most ϕ positive predictions in the exploration region $\textsc{Explore}_t$, where

$$\phi := \left(\sum_{(x,z)\in S_t\cap\textsc{Exploit}_t} \mathbb{I}\left[f_t(x,z)=1\right]\right)\cdot\alpha_{\text{explore}}.$$

Due to this, to establish Equation (6), it suffices to prove the following

$$\frac{\sum_{(x,z)\in S_t\cap\textsc{Exploit}_t} \mathbb{I}\left[f_t(x,z) = 1 \text{ and } y = 0\right]}{\sum_{(x,z)\in S_t\cap\textsc{Exploit}_t} \mathbb{I}\left[f_t(x,z) = 1\right]} \leq \alpha_{\text{exploit}} + \frac{10\varepsilon}{\lambda}. \tag{8}$$

First, we will express the above ration as a ratio of expectations over μ using a standard generalization bound. Then, we will use Lemma C.1 to complete the proof. Note that, we can bound the VC-dimension of $\mathcal{F}$ by $|D|$: $\mathrm{VC}(\mathcal{F})\leq|D|$ (Shalev-Shwartz and Ben-David, 2014). Hence, using the standard generalization inequality in Section 28.1 (Shalev-Shwartz and Ben-David, 2014) and the lower bound on n, we have the following bound: for any distribution ζ over D

$$\Pr\left[\forall_{f\in\mathcal{F}},\quad \left|\mathop{\mathbb{E}}_{S\sim\zeta^n}\left[h(f(X,Z),Y,Z)\right] - \mathop{\mathbb{E}}_{\zeta}\left[h(f(X,Z),Y,Z)\right]\right| \leq \varepsilon\right] \geq 1-\delta.$$

Setting ζ to be the distribution μ restricted to the $\textsc{Exploit}_t$ and observing that S_t has n samples drawn i.i.d. from μ, we deduce the following from the above inequality

$$\Pr\left[\forall_{f\in\mathcal{F}},\quad \left|\begin{array}{c} \mathbb{E}_{S_t}\left[h(f(X,Z),Y,Z)\mid (X,Z)\in \textsc{Exploit}_t\right] \\ -\ \ \mathbb{E}_{\mu}\left[h(f(X,Z),Y,Z)\mid (X,Z)\in \textsc{Exploit}_t\right]\end{array}\right| \leq \varepsilon\right] \geq 1-\delta. \tag{9}$$

[3] If $\varepsilon > \lambda/3$, we can set $\varepsilon = \lambda/3$. This only improves the guarantee we prove, and the lower bound on n is not violated as it depends on $\min\{\lambda/3, \varepsilon\}$.

Combining Lemma C.1, Equation (9) using the triangle inequality, and the union bound, we get the following bound

$$\Pr\left[\forall_{f\in\mathcal{F}},\quad \left|\begin{array}{c}\mathbb{E}_{\eta_w}\left[h(f(X,Z),Y,Z)\mid (X,Z)\in\textsc{Exploit}_t\right]\\ -\ \mathbb{E}_{S_t}\left[h(f(X,Z),Y,Z)\mid (X,Z)\in\textsc{Exploit}_t\right]\end{array}\right|\leq 2\varepsilon\right]\geq 1-2\delta. \tag{10}$$

Let $\mathscr{E}$ be the event that the following holds

$$\forall_{f\in\mathcal{F}},\quad \left|\begin{array}{c}\mathbb{E}_{\eta_w}\left[h(f(X,Z),Y,Z)\mid (X,Z)\in\textsc{Exploit}_t\right]\\ -\ \mathbb{E}_{S_t}\left[h(f(X,Z),Y,Z)\mid (X,Z)\in\textsc{Exploit}_t\right]\end{array}\right|\leq 2\varepsilon$$

The Equation (10) implies that $\Pr[\mathscr{E}]\geq 1-2\delta$. Conditioned on $\mathscr{E}$, selecting

$$h(y_1,y_2,z)=\mathbb{I}[y_1=1\text{ and }y_2=0],$$

Equation (9) implies that

$$\left|\begin{array}{l}\mathbb{E}_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\mid (X,Z)\in\textsc{Exploit}_t\right]\\ -\quad\frac{\sum_{(x,z)\in S_t\cap\textsc{Exploit}_t}\mathbb{I}[f_t(x,z)=1\text{ and }y=0]}{|S_t\cap\textsc{Exploit}_t|}\end{array}\right|\leq 2\varepsilon. \tag{11}$$

Similarly, conditioned on $\mathscr{E}$, for

$$h(y_1,y_2,z)=\mathbb{I}[y_1=1],$$

Equation (9) implies that

$$\left|\mathop{\mathbb{E}}_{\eta_w}\left[f_t(X,Z)=1\mid (X,Z)\in\textsc{Exploit}_t\right]-\frac{\sum_{(x,z)\in S_t\cap\textsc{Exploit}_t}\mathbb{I}\left[f_t(x,z)=1\right]}{|S_t\cap\textsc{Exploit}_t|}\right|\leq 2\varepsilon. \tag{12}$$

Since f_t is feasible for Program (2) in Step 3 of Algorithm 1, it follows that f_t satisfies the following constraints

$$\frac{\Pr_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\right]}{\Pr_{\eta_w}\left[f_t(X,Z)=1\right]}\leq\alpha_{\text{exploit}}+\varepsilon\quad\text{and}\quad\mathop{\Pr}_{\eta_w}\left[f_t(X,Z)=1\right]\geq\lambda-\varepsilon. \tag{13}$$

Now, we are ready to prove Equation (8).

$$\begin{aligned}\frac{\sum_{(x,z)\in S_t\cap\textsc{Exploit}_t}\mathbb{I}\left[f_t(x,z)=1\text{ and }\mathcal{O}(x,z)=0\right]}{\sum_{(x,z)\in S_t\cap\textsc{Exploit}_t}\mathbb{I}\left[f_t(x,z)=1\right]} &\overset{(11),\,(12)}{\leq} \frac{\Pr_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\right]+2\varepsilon}{\Pr_{\eta_w}\left[f_t(X,Z)=1\right]-2\varepsilon}\\ &= \frac{\Pr_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\right]+2\varepsilon}{\Pr_{\eta_w}\left[f_t(X,Z)=1\right]}\cdot\frac{1}{1-\frac{2\varepsilon}{\Pr_{\eta_w}[f_t(X,Z)=1]}}\\ &\overset{(13)}{\leq} \frac{\Pr_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\right]+2\varepsilon}{\Pr_{\eta_w}\left[f_t(X,Z)=1\right]}\cdot\frac{1}{1-\frac{2\varepsilon}{\lambda-2\varepsilon}}\\ &\leq \frac{\Pr_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\right]+\varepsilon}{\Pr_{\eta_w}\left[f_t(X,Z)=1\right]}\cdot\left(1+\frac{4\varepsilon}{\lambda}\right).\end{aligned}$$

$$\left(\text{Using that }\left(1-\tfrac{x}{1-2x}\right)^{-1}\leq 1+4x\text{ for all }x\in[0,\tfrac{1}{4}]\text{ and }\varepsilon\leq\tfrac{\lambda}{4}\right) \tag{14}$$

An upper bound on the RHS of the above equation is as follows

$$\begin{aligned}\frac{\Pr_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\right]+2\varepsilon}{\Pr_{\eta_w}\left[f_t(X,Z)=1\right]}\cdot\left(1+\frac{4\varepsilon}{\lambda}\right) &\overset{(13)}{\leq} \frac{\Pr_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\right]}{\Pr_{\eta_w}\left[f_t(X,Z)=1\right]}\cdot\left(1+\frac{4\varepsilon}{\lambda}\right)+\frac{2\varepsilon}{\lambda-\varepsilon}\cdot\left(1+\frac{4\varepsilon}{\lambda}\right)\\ &\leq \frac{\Pr_{\eta_w}\left[f_t(X,Z)=1\text{ and }Y=0\right]}{\Pr_{\eta_w}\left[f_t(X,Z)=1\right]}\cdot\left(1+\frac{4\varepsilon}{\lambda}\right)+\frac{4\varepsilon}{\lambda}\cdot\left(1+\frac{4\varepsilon}{\lambda}\right)\\ &\qquad\left(\text{Using that }\tfrac{x}{1-x}\leq 2x\text{ for all }x\in[0,\tfrac{1}{4}]\text{ and }\varepsilon\leq\tfrac{\lambda}{4}\right)\\ &\overset{(13)}{\leq} (\alpha_{\text{exploit}}+\varepsilon)\cdot\left(1+\frac{4\varepsilon}{\lambda}\right)+\frac{2\varepsilon}{\lambda}\cdot\left(1+\frac{4\varepsilon}{\lambda}\right)\\ &\leq \alpha_{\text{exploit}}+\frac{14\varepsilon}{\lambda}\qquad\left(\text{Using that }4\varepsilon\leq\lambda,\ \alpha\leq 1,\text{ and }0\leq\lambda\leq 1\right)\end{aligned}$$

Substituting this in Equation (13), implies Equation (8). The result follows as $\mathscr{E}$ holds with probability at least $1-2\delta$. □

Remark C.3. Note that the above proof does not use Assumption 2.3. This assumption is only required to ensure that Program (2) in Step 3 of Algorithm 1 is feasible for all $t \in \{1, 2, \dots\}$.

C.1.1. Proof of Lemma C.1

Discussion of techniques. The generalization bound in Lemma C.1 shows that the reweighted distribution η_w in Step 3 of Algorithm 1 is a good approximation of μ on $\textsc{Exploit}_t$. Recall that this bound is as follows: if $n \geq n_0 := |D| \cdot \text{poly}\left({}^1\!/\!_\lambda, {}^1\!/\!_\tau, {}^1\!/\!_{\min\{\varepsilon, \alpha-\varepsilon\}}\right) \cdot \log\left({}^{|D|}\!/\!_{\sigma\delta}\right)$, then at any iteration t and for any bounded function $h\colon \{0,1\} \times \{0,1\} \times [p] \to [0,1]$ the following holds

$$\Pr\left[\forall_{f \in \mathcal{F}}, \quad \left| \begin{array}{c} \mathbb{E}_{\eta_w \mid (X,Z) \in \textsc{Exploit}_t}\left[h(f(X,Z), Y, Z)\right] \\ -\ \mathbb{E}_{\mu \mid (X,Z) \in \textsc{Exploit}_t}\left[h(f(X,Z), Y, Z)\right] \end{array} \right| \leq O(\varepsilon)\right] \geq 1 - \delta. \tag{15}$$

At any t, η_w has the following product form

$$\eta_w(x,y,z) \propto \mathbb{I}\left[(x,z) \in \textsc{Exploit}_t\right] \cdot \Pr_{S_0,\dots,S_t}\left[(X,Z) = (x,z)\right] \cdot \Pr_{S_0,\dots,S_t}\left[(x,y,z) \in \bigcup_{i=0}^{t-1} L_i\right] \tag{16}$$

The first term ensures that the support of η_w is $\textsc{Exploit}_t$. The second term is an estimate of $\mu(x,z)$. The third term is an estimate of $\Pr_\mu\left[Y = y \mid (X,Z) = (x,z)\right]$. If both of these estimates are exact, then Equation (15) follows. We prove that the estimates are nearly-correct for sample (x,z) with a sufficient probability of mass under μ. Consider the set R of all samples that have a small probability mass under μ: $R := \left\{(x,z) \mid \mu(x,z) \leq {}^\varepsilon\!/\!_{|D|}\right\}$. The proof of the claim for the first estimate is straightforward: the Chernoff bound and the union bound imply that if $n \geq n_0$, then

$$\Pr\left[\forall_{(x,z) \in \textsc{Exploit}_t \setminus R}, \quad \Pr_{S_t}\left[(X,Z) = (x,z)\right] \in (1 \pm \varepsilon) \cdot \mu(x,z)\right] \geq 1 - \delta. \tag{17}$$

The claim for the second estimate holds if $\bigcup_{i=0}^{t-1} L_i$ has at least $k_0 := \text{poly}\left({}^1\!/\!_\varepsilon\right) \cdot \log\left({}^{|D|}\!/\!_\delta\right)$ copies of each $(x,z) \in \textsc{Exploit}_t \setminus R$. Under this assumption, standard techniques imply that

$$\Pr\left[\forall_{(x,z) \in \textsc{Exploit}_t \setminus R}, \Pr_{S_0,\dots,S_t}\left[(x,y,z) \in \bigcup_{i=0}^{t-1} L_i\right] \in (1 \pm \varepsilon) \Pr_\mu\left[Y = y \mid (X,Z) = (x,z)\right]\right] \geq 1 - \delta. \tag{18}$$

Together, Equations (17) and (18), imply that: with probability at least $1 - \delta$, $\eta_w(x,y,z) \in (1 \pm O(\varepsilon)) \cdot \mu(x,y,z)$ for any $(x,z) \in \textsc{Exploit}_t \setminus R$ and $y \in \{0,1\}$. That is, η_w is a good approximation of μ on $\textsc{Exploit}_t \setminus R$. Since the total probability mass of samples in R is at most ε, the required generalization bound (Equation (15)) follows. It remains to show that there are at least k_0 copies of each $(x,z) \in \textsc{Exploit}_t \setminus R$ in $\bigcup_{i=0}^{t-1} L_i$ (which was used to prove Equation (18)).

Fix any $(x,z) \in \textsc{Exploit}_t \setminus R$. Let N_i be the number of copies of (x,z) in L_i and $N = \sum_{i=0}^{t-1} N_i$ be the number of copies of (x,z) in $\bigcup_{i=0}^{t-1} L_i$. One can show that $\mathbb{E}\left[N_i\right] = \Omega(n \times g(x,z;f_i))$. Since $(x,z) \in \textsc{Exploit}_t$, $\sum_{i=0}^{t} g(x,z;f_i) \geq \tau$ and, hence, by linearity of expectation $\mathbb{E}\left[N\right] \geq \Omega(n \times \tau) = k_0$. Thus, it suffices to show that $N \geq \Omega\left(\mathbb{E}\left[N\right]\right)$ with high probability. Here, N is a sum of 0/1 random variables Z_{ij}: denoting whether (x,z) is the j-th sample in L_i. If $\left\{Z_{ij}\right\}_{i,j}$ are independent, then one may hope to prove the concentration of N via standard inequalities.

The main challenge is that, since Algorithm 1's choices at the i-th iteration depend on past observations, independence may not hold. Let T be the last iteration when (x,z) is in the exploration region. We overcome this using the fact that $\{Z_{ij} \mid i \in [T], j \in |L_i|\}$ are mutually independent because till (x,z) is included in the exploitation region, labels observed for (x,z) do not affect η_w and, hence, Algorithm 1's choices.

Proof of Lemma C.1. Recall the following definitions

$$\sigma := \min_{z \in [p]} \sigma_z, \quad \text{where,} \quad \forall z \in [p], \quad \sigma_z := \min_{f \in \mathcal{F}} \frac{\min_{x \in \mathcal{X}} g(x,z;f)}{\sum_{(x,\ell) \in \mathcal{D}} g(x,\ell;,f)}. \tag{19}$$

Fix any bounded function $h\colon \{0,1\} \times \{0,1\} \times [p] \to [0,1]$, any number $t \in \mathbb{N}$, and any constants $\varepsilon, \tau, \delta \in (0,1]$. Fix any n satisfying the following lower bound

$$n \geq \frac{12\,|D|}{\tau\varepsilon^3} \cdot \frac{1}{\alpha_{\text{explore}}\lambda} \cdot \log\frac{|D|}{\delta\sigma}.$$

The proof is divided into three steps. In the first step, we show that, when D is finite and n is sufficiently large, with probability at least $1-\delta$, at each t, it holds that: for each $(x,z) \in \textsc{Exploit}_t$ satisfying $\mu(x,z) \geq \frac{\varepsilon}{|D|}$

$$\Pr_{(X,Y,Z)\sim S_i}[(X,Z)=(x,z)] \in (1\pm\varepsilon)^2 \cdot \mu(x,z).$$

In the second step, we show that with probability at least $1-\delta$, at each t, it holds that: for each $(x,z) \in \textsc{Exploit}_t$ satisfying $\mu(x,z) \geq \frac{\varepsilon}{|D|}$ and $y \in \{0,1\}$

$$\Pr_{(X,Y,Z)\sim L_0\cup L_1\cup\cdots\cup L_t}[Y=y \mid (X,Z)=(x,z)] \in (1\pm\varepsilon)^2 \cdot \Pr_{(X,Y,Z)\sim\mu}[Y=y \mid (X,Z)=(x,z)]\,.$$

In the third step, we conclude the proof.

Step 1 ($\Pr_{S_i}[(X,Z)=(x,z)] \in (1\pm\varepsilon)^2\cdot\mu(x,z)$ for each $(x,z) \in \textsc{Exploit}_t$): Fix any $(x,z) \in D$ with $\mu(x,z) \geq \frac{\varepsilon}{|D|}$. Since for each t, S_t contains of n iid samples from μ, it follows that the expected number of copies of (x,z) in S_t, say $N_{x,z,t}$, is

$$\mathbb{E}\left[N_{x,z,t}\right] = n\cdot\mu(x,z) \overset{\mu(x,z)\geq\frac{\varepsilon}{|D|}}{\geq} \frac{12}{\tau\varepsilon^2}\cdot\frac{1}{\alpha_{\text{explore}}\lambda}\cdot\log\frac{|D|}{\delta\sigma}. \tag{20}$$

Moreover, the Chernoff bound implies that with a probability of at least

$$1-2\exp\left(-\frac{\varepsilon^2}{3}\cdot\frac{12}{\tau\varepsilon^2}\frac{1}{\alpha_{\text{explore}}\lambda}\cdot\log\frac{|D|}{\delta\sigma}\right) \geq 1-2\frac{\delta\sigma}{|D|} \qquad \text{(Using that } 0\leq\tau\alpha_{\text{explore}}\lambda\leq 1\text{)} \tag{21}$$

$N_{x,z,t}$ lies in the following interval

$$\left[(1-\varepsilon)n\cdot\mu(x,z), (1+\varepsilon)n\cdot\mu(x,z)\right]. \tag{22}$$

Now the union bound over all $x \in \mathcal{D}$ implies that with a probability of at least $1-2\delta\sigma$, the following holds: for all $(x,z) \in \textsc{Exploit}_t$ satisfying $\mu(x,z) \geq \frac{\varepsilon}{|D|}$

$$\Pr_{S_t}[(X,Z)=(x,z)] = \frac{N_{x,z,t}}{n} \overset{(21),(22)}{\in} (1\pm\varepsilon)^2\cdot\mu(x,z). \tag{23}$$

Step 2 ($\Pr_{L_0\cup\cdots\cup L_t}[y \mid (x,z)] \in (1\pm\varepsilon)^2\cdot\Pr_\mu[y \mid (x,z)]$ for each $(x,z) \in \textsc{Exploit}_t$): Fix any $(x,z) \in D$ with $\mu(x,z) \geq \frac{\varepsilon}{|D|}$. Let $T_{x,z} \in \mathbb{N}$ be the last iteration where (x,z) is in the exploration region. Since $g(x,z;f_i) \geq \sigma$ for each $i \in \mathbb{N}$, it follows that

$$T_{x,z} \leq \sigma^{-1}. \tag{24}$$

For any $i \in \mathbb{N}$, let $N_{x,z,i}$ be the number of copies of (x,z) in S_i. Let $\mathscr{E}_{x,z}$ be the event that

$$\forall 1\leq i\leq T_{x,z}, \quad N_{x,z,i} \geq \frac{6}{\tau\varepsilon^2}\cdot\frac{1}{\alpha_{\text{explore}}\lambda}\cdot\log\frac{|D|}{\delta\sigma}.$$

The Chernoff bound and the union bound imply that

$$\Pr\left[\mathscr{E}_{x,z}\right] \geq 1-T_{x,z}\times 2\frac{\delta\sigma}{|D|} \overset{(24)}{\geq} 1-2\frac{\delta}{|D|}.$$

Order the $N_{x,z,i}$ copies of (x,z) in S_i arbitrarily for each $i \in \mathbb{N}$. Observe that each copy of (x,z) in S_i (for any $i \in [T_{x,z}]$) is positively labeled with probability $\alpha_{\text{explore}}\lambda\cdot g(x,z;f_i)$. Henceforth, we abbreviate $g(x,z;f_i)$ as $g_i(x,z)$. The event that (x,z) is positively labeled in iterations $1\leq i_1,i_2\leq T_{x,z}$ are independent as till (x,z) is not in the exploitation region, it does not affect Algorithm 1's decision to positively label samples. Let $Z_j \in \{0,1\}$ be the indicator random variable that the j-th copy of (x,z) in S_i exists and is positively labeled for some $1\leq i\leq T_{x,z}$. Define

$$\Delta \coloneqq \frac{6}{\tau\varepsilon^2}\cdot\frac{1}{\alpha_{\text{explore}}\lambda}\cdot\log\frac{|D|}{\delta\sigma}. \tag{25}$$

Conditioned on $\mathscr{E}_{x,z}$ it holds that: for all $1 \leq j \leq \Delta$

$$\begin{aligned}
\Pr\left[Z_j = 1 \mid \mathscr{E}_{x,z}\right] &= 1 - \prod_{i=0}^{T} \left(1 - g_i(x,z)\alpha_{\text{explore}}\lambda\right) \\
&\geq 1 - \prod_{i=0}^{T} \exp\left(-g_i(x,z)\alpha_{\text{explore}}\lambda\right) && \text{(Using that } e^{-x} \geq 1 - x \text{ for all } x \in \mathbb{R}\text{)} \\
&= 1 - \exp\left(-\sum_{i=0}^{T} g_i(x,z)\alpha_{\text{explore}}\lambda\right) \\
&\geq 1 - \exp\left(-\tau\alpha_{\text{explore}}\lambda\right) \\
&\qquad \text{(Using that } T \text{ is the last iteration when } (x,z) \text{ is in the exploration region and, hence, } \textstyle\sum_{i=0}^{T} g_i(x,z) \geq \tau\text{)} \\
&\geq 1 - \left(1 - \frac{1}{2}\tau\alpha_{\text{explore}}\lambda\right) && \text{(Using that } e^{-x} \leq 1 - \tfrac{x}{2} \text{ for all } x \in [0,1] \text{ and } 0 \leq \tau\alpha_{\text{explore}}\lambda \leq 1\text{)} \\
&= \frac{1}{2}\tau\alpha_{\text{explore}}\lambda. && (26)
\end{aligned}$$

Moreover, Z_i and Z_j are independent for all $i \neq j$. Hence, the Chernoff bound implies that

$$\begin{aligned}
\Pr\left[\frac{\sum_{j=0}^{\Delta} Z_j}{\Delta} \geq \frac{1}{2} \cdot \frac{1}{2}\tau\alpha_{\text{explore}}\lambda\right] \quad &\geq \quad 1 - 2\exp\left(-\frac{1}{3} \cdot \frac{1}{4} \cdot \frac{\Delta}{2}\tau\alpha_{\text{explore}}\lambda\right) \\
&\overset{(25)}{=} \quad 1 - 2\exp\left(-\frac{\tau\alpha_{\text{explore}}\lambda}{4\tau\varepsilon^{-2}\alpha_{\text{explore}}\lambda} \cdot \log\frac{|D|}{\delta\sigma}\right) \\
&\overset{(25)}{=} \quad 1 - 2\frac{\delta\sigma}{|D|}. && \text{(Using } 0 < \varepsilon \leq \tfrac{1}{2}\text{)} \quad (27)
\end{aligned}$$

Let $P_{x,z,i}$ be the number of copies of (x,z) that are positively labeled in the first i iterations. Observe that $P_{x,z,\Delta} \geq \sum_{j=0}^{\Delta} Z_j$. Thus, the union bound implies that with probability at least $1 - 2\delta$, for each $(x,z) \in \textsc{Exploit}_t$ satisfying $\mu(x,z) \geq \frac{\varepsilon}{|D|}$, at least

$$P_{x,z,t} \geq P_{x,z,\Delta} \geq \frac{3}{2}\varepsilon^{-2}\log\frac{|D|}{\delta\sigma}$$

copies of (x,z) are positively labeled in the first t iterations. Since each positively lebled copy of (x,z) is inclded in $\bigcup_{i=0}^{t-1} L_i$, it follows that with probability at least $1 - 2\delta$, at least $\frac{3}{2\varepsilon^2}\log\frac{|D|}{\delta\sigma}$ copies of (x,z) are included in $\bigcup_{i=0}^{t-1} L_i$ for each $(x,z) \in \textsc{Exploit}_t$ satisfying $\mu(x,z) \geq \frac{\varepsilon}{|D|}$. Suppose this event is $\mathscr{E}_t$. Consider $(X,Y,Z) \sim \mu$. Since conditioned on $(X,Z) = (x,z)$, Y is a bernoulli random variable with

$$\Pr[Y = 1] = \Pr_{\mu}[Y = 1 \mid (X,Z) = (x,z)], \tag{28}$$

conditioned on $\mathscr{E}_t$, it holds that

$$\Pr_{(X,Y,Z) \sim L_0 \cup L_1 \cup \cdots \cup L_t}\left[Y = y \mid (X,Z) = (x,z)\right] = \frac{\sum_{(X,Y,Z) \in L_0 \cup \cdots \cup L_t \mid (X,Z) = (x,z)} \mathbb{I}\left[Y = y\right]}{P_{x,z,t}}.$$

Using standard concentration properties of sums of Bernoulli random variables and the fact that conditioned on $\mathscr{E}_t$, $P_{x,z,t} \geq \frac{3}{2\varepsilon^2}\log\frac{|D|}{\delta\sigma}$, it follows, that with probability at least $1 - \frac{\delta\sigma}{|D|}$

$$\sum_{(X,Y,Z) \in L_0 \cup \cdots \cup L_t \mid (X,Z) = (x,z)} \mathbb{I}\left[Y = y\right] \overset{(28)}{\in} (1 \pm \varepsilon) \cdot P_{x,z,t} \cdot \Pr_{\mu}[Y = 1 \mid (X,Z) = (x,z)].$$

Thus, conditioned on event $\mathscr{E}_t$, with probability at least $1 - \frac{\delta\sigma}{|D|}$, it holds that

$$\Pr_{(X,Y,Z) \sim L_0 \cup L_1 \cup \cdots \cup L_t}\left[Y = y \mid (X,Z) = (x,z)\right] \in (1 \pm \varepsilon)\Pr_{\mu}[Y = 1 \mid (X,Z) = (x,z)].$$

The union bound now implies that, conditioned on $\mathscr{E}_t$, with probability at least $1-2\delta\sigma$, for an $(x,z) \in \textsc{Exploit}_t$ satisfying $\mu(x,z) \geq \frac{\varepsilon}{|D|}$, it holds that

$$\Pr_{(X,Y,Z)\sim L_0\cup L_1\cup\cdots\cup L_t}[Y=y \mid (X,Z)=(x,z)] \in (1\pm\varepsilon)\Pr_{\mu}[Y=1 \mid (X,Z)=(x,z)].$$

This completes Step 2.

Step 3 (Completing the proof of Lemma C.1): Let R be the set of all samples (x,z) such that $\mu(x,z) \leq \frac{\varepsilon}{|D|}$:

$$R \coloneqq \left\{(x,z) \in \mathcal{X}\times[p] \mid \mu(x,z) \leq \frac{\varepsilon}{p\,|D|}\right\}.$$

Since $|S| \leq |D|$, $\mu(S) \leq \varepsilon$. Hence, the Chernoff bound implies that with probability at least $1-\delta$,

$$|R\cap S_t| \leq n\varepsilon + \sqrt{n\varepsilon\log\frac{1}{\delta}}. \tag{29}$$

Consequently, with probability at least $1-\delta$, it holds that

$$\begin{aligned}\Pr_{S_t}[(X,Z)\in R] &\overset{(29)}{\leq} \frac{n\varepsilon + \sqrt{n\varepsilon\log\frac{|D|}{\delta}}}{n}\\ &\leq \varepsilon + \varepsilon^2\sqrt{\frac{\tau\alpha_{\text{explore}}\lambda}{12\,|D|}} && \text{(Using that } n \geq \tfrac{12|D|}{\tau\varepsilon^3}\cdot\tfrac{1}{\alpha_{\text{explore}}\lambda}\cdot\log\tfrac{|D|}{\delta\sigma}\text{)}\\ &\leq 2\varepsilon. && \text{(Using that } |D|\geq 1,\ 0\leq\alpha\lambda\tau\leq 1, \text{ and } \varepsilon\leq 1\text{)}\end{aligned}$$

For any t, let $\mathscr{G}_t$ be the event that the following holds:

$$\begin{aligned} & \Pr_{S_t}[(X,Z)\in R] \leq 2\varepsilon, \\ \forall (x,z)\in\textsc{Exploit}_t\setminus R, \quad & \Pr_{S_t}[(X,Z)=(x,z)] \in (1\pm\varepsilon)^2\cdot\mu(x,z), \\ \forall (x,z)\in\textsc{Exploit}_t\setminus R, \quad & \Pr_{L_0\cup\cdots\cup L_t}[Y=y \mid (X,Z)=(x,z)] \in (1\pm\varepsilon)\cdot\Pr_{\mu}[Y=1 \mid (X,Z)=(x,z)]. \end{aligned} \tag{30}$$

Steps 1 and 2 show that for each t,

$$\Pr[\mathscr{G}] \geq 1 - 4\delta\sigma - 2\delta \geq 1-6\delta.$$

Fix any classifier $f \in \mathscr{F}$, conditioned on $\mathscr{G}$ the following inequalities hold

$$
\begin{aligned}
&\left|\mathop{\mathbb{E}}_{\eta_w}\left[h(f(X,Z),Y,Z) \mid (X,Z) \in \textsc{Exploit}_t\right] - \mathop{\mathbb{E}}_{\mu}\left[h(f(X,Z),Y,Z) \mid (X,Z) \in \textsc{Exploit}_t\right]\right| \\
&\quad = \left| \sum_{(x,z)\in\textsc{Exploit}_t} \sum_{y\in\{0,1\}} h(f(x,z),y,z) \left(\Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right) \right| \\
&\quad \leq \sum_{(x,z)\in\textsc{Exploit}_t} \sum_{y\in\{0,1\}} \left| \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right| \\
&\hspace{20em} \text{(Using that the range of } h \text{ is } [0,1] \text{ and triangle inequality)} \\
&\quad \leq \sum_{(x,z)\in\textsc{Exploit}_t\setminus R} \sum_{y\in\{0,1\}} \left| \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right| \\
&\qquad + \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \left| \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right| \\
&\quad = \sum_{(x,z)\in\textsc{Exploit}_t\setminus R} \sum_{y\in\{0,1\}} \left| \begin{array}{l} \Pr_{S_t}[(X,Z)=(x,z)] \cdot \Pr_{(X,Y,Z)\sim L_0\cup\cdots\cup L_t}[Y=y \mid (X,Z)=(x,z)] \\ -\ \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \end{array} \right| \\
&\qquad + \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \left| \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right| \\
&\quad \in \sum_{(x,z)\in\textsc{Exploit}_t\setminus R} \sum_{y\in\{0,1\}} \left| \begin{array}{l} (1\pm\varepsilon)^2 \Pr_{\mu}[(X,Z)=(x,z)] \cdot \Pr_{(X,Y,Z)\sim \mu}[Y=y \mid (X,Z)=(x,z)] \\ -\ \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \end{array} \right| \\
&\qquad + \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \left| \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right| \\
&\quad \leq \sum_{(x,z)\in\textsc{Exploit}_t\setminus R} \sum_{y\in\{0,1\}} \left|(1\pm\varepsilon)^2 - 1\right| \cdot \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \\
&\qquad + \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \left| \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right| \\
&\quad \leq 3\varepsilon \cdot \sum_{(x,z)\in\textsc{Exploit}_t\setminus R} \sum_{y\in\{0,1\}} \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \\
&\hspace{16em} \text{(Using that } (1-x)^2 \leq 1-3x \text{ and } (1+x)^2 \leq 1+3x \text{ for all } x\in[0,1]) \\
&\qquad + \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \left| \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right| \\
&\quad \leq 3\varepsilon + \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \left| \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] - \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right| \\
&\quad \leq 3\varepsilon + \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \left(\Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] + \Pr_{\mu}[(X,Y,Z)=(x,y,z)] \right) \\
&\quad \leq 3\varepsilon + 2\varepsilon + \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)]. \qquad (31)
\end{aligned}
$$

It remains to upper bound $\sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \left|\Pr_{\eta_w}[(X,Y,Z)=(x,y,z)]\right|$. Towards this, observe that

$$
\begin{aligned}
&\sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \Pr_{\eta_w}[(X,Y,Z)=(x,y,z)] \\
&\quad = \sum_{(x,z)\in\textsc{Exploit}_t\cap R} \sum_{y\in\{0,1\}} \Pr_{S_t}[(X,Z)=(x,z)] \cdot \Pr_{(X,Y,Z)\sim L_0\cup L_1\cup\cdots\cup L_t}[Y=y \mid (X,Z)=(x,z)]
\end{aligned}
$$

$$\begin{aligned}
&\leq 2 \sum_{(x,z)\in \textsc{Exploit}_t \cap R} \Pr_{S_t}[(X,Z)=(x,z)] \\
&= 2 \Pr_{S_t}[(X,Z)\in \textsc{Exploit}_t \cap R] \\
&\leq 4\varepsilon. && \text{(Using Equation (30))}
\end{aligned}$$

Substituting this in Equation (31), implies that conditioned on $\mathscr{G}$, for any $f \in \mathscr{F}$

$$\left| \begin{array}{rl} & \mathbb{E}_{\eta_w}\left[h(f(X,Z),Y,Z) \mid (X,Z)\in \textsc{Exploit}_t\right] \\ - & \mathbb{E}_{\mu}\left[h(f(X,Z),Y,Z) \mid (X,Z)\in \textsc{Exploit}_t\right] \end{array} \right| \leq 7\varepsilon.$$

Since $\Pr[\mathscr{G}] \geq 1-6\delta$, the result follows by rescaling δ and ε by 6 and 7 respectively.

C.2. Proof of Theorem 4.2

In this section, we prove Theorem 4.2. For ease of reference, we restate Theorem 4.2 below.

Theorem 4.2 (Fairness: Improvement in group-wise utility). *Suppose f_0 is (α,λ)-feasible (Assumption 2.3) and $\mathcal{F}$ is derived from $\mathcal{B}\subseteq\{0,1\}^{\mathcal{X}}$. For any $\varepsilon,\delta,\tau\in(0,1]$ and tuple γ, Algorithm 1 satisfies the following at every iteration t and $z\in[p]$: If $n \geq |D|\cdot \operatorname{poly}\left(1/\lambda, 1/\tau, 1/\min\{\varepsilon,\alpha-\varepsilon\}\right)\cdot\log\left(|D|/\sigma\delta\right)$, then with probability at least $1-\delta$,*

$$\mathit{Util}_{\mu,t}(f_t,\gamma,z) \geq \max_{0\leq i\leq t-1} \mathit{Util}_{\mu,t}(f_i,\gamma,z) - \varepsilon.$$

Where $\mathit{Util}_{\mu,t}(f,z)$ is the utility of f over draws $(X,Y,Z)\sim\mu$ conditioned on $(X,Z)\in\textsc{Exploit}_t$ and $Z=z$. The randomness at the t-th iteration is w.r.t. the randomness in $S_1,\dots,S_t$ and Algorithm 1.

Recall that we assume that Assumption 2.3 holds with constants $\alpha,\lambda\in(0,1]$. This implies that the classifier $f_0\in\mathcal{F}$ satisfies the α-FDR constraint and has a selection rate of at least λ, i.e.,

$$\frac{\Pr_\mu\left[f_0(X,Z)\neq Y\right]}{\Pr_\mu\left[f_0(X,Z)=1\right]} \leq \alpha \quad\text{and}\quad \Pr_\mu\left[f_0(X,Z)=1\right]\geq\lambda. \tag{32}$$

Where, as in the rest of the proof, we use $\Pr_\mu[\cdot]$ to denote $\Pr_{(X,Y,Z)\sim\mu}[\cdot]$ and $\mathbb{E}_\mu[\cdot]$ to denote $\mathbb{E}_{(X,Y,Z)\sim\mu}[\cdot]$. We also assume that the hypothesis class $\mathcal{F}$ is derived from some base hypothesis class $\mathcal{B}\subseteq\{0,1\}^{\mathcal{X}}$ (Section 4). We need to show that for any utility function Util and numbers α, λ, δ, and τ, if n is sufficiently large, then at every iteration $t\in\{1,2,\dots\}$ with probability at least $1-\delta$, the classifier f_t learned by the framework in Algorithm 1 satisfies

$$\mathrm{Util}_{\mu,t}(f_t,z) \geq \max_{0\leq i\leq t-1}\mathrm{Util}_{\mu,t}(f_i,z).$$

Where $\mathrm{Util}_{\mu,t}(f,z)$ is the utility of classifier f over samples $(X,Y,Z)\sim\mu$ conditioned on the facts that $(X,Z)\in\textsc{Exploit}_t$ and $Z=z$. We will show that the following lower bound on n is sufficient

$$n \geq \frac{12\,|D|}{\tau\varepsilon^3}\cdot\frac{1}{\alpha_{\mathrm{explore}}\lambda}\cdot\log\frac{|D|}{\delta\sigma}. \tag{33}$$

Fix any two iterations $t_1 > t_2$. We use $\eta(w,t)$ to denote the distribution η_w in the t-th iteration. Since $\mathcal{F}$ is derived from $\mathcal{B}$, each hypothesis $f_t\in\mathcal{F}$ is a tuple $(f_{t1},f_{t2},\dots,f_{tp})$ of hypothesis from $\mathcal{B}$. Using this and the fact that the classifier f_t learned by Algorithm 1 is an optimal solution of Program (2), it follows that the corresponding hypothesis $f_{t,z}$, for each $z\in[p]$, is a solution to the following optimization program (which is an alternate version of Program (2) specifically over samples in the z-th group): for each $z\in[p]$

$$\begin{aligned}
f_{t,z} = &\arg\max_{h\in\mathcal{F}}\ \mathrm{Util}_{\eta(w,t)}\left(f,\gamma,z\right), \\
&\text{s.t., } \Pr_{\eta(w,t)}\left[h\neq y \mid h=1, Z=z\right] \leq \alpha_{\mathrm{exploit}}+\varepsilon \text{ and } \Pr_{\eta_w}\left[h=1, Z=z\right]\geq\lambda-\varepsilon.
\end{aligned} \tag{34}$$

Let $\mathscr{H}$ be the event that for both $t\in\{t_1,t_2\}$ and any $h\colon\{0,1\}\times\{0,1\}\times[p]\to[0,1]$

$$\Pr\left[\forall_{f\in\mathcal{F}},\quad \left| \begin{array}{rl} & \mathbb{E}_{\eta_w}\left[h(f(X,Z),Y,Z) \mid (X,Z)\in \textsc{Exploit}_t\right] \\ - & \mathbb{E}_{\mu}\left[h(f(X,Z),Y,Z) \mid (X,Z)\in \textsc{Exploit}_t\right] \end{array} \right| \leq \varepsilon\right] \geq 1-\delta.$$

By Lemma C.1 $\Pr\left[\mathscr{H}\right] \geq 1-2\delta$. Conditioned on $\mathscr{H}$

$$\begin{aligned}
\mathrm{Util}_{\mu}\left(f_{t_1},\gamma,z\right) &= \sum_{i,j\in\{0,1\}} \gamma_{ij} \Pr_{\mu}\left[f_{t_1}(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right] && \text{(Using Definition 2.1)}\\
&\geq \sum_{i,j\in\{0,1\}} \gamma_{ij} \Pr_{\eta(w,t_1)}\left[f_{t_1}(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right] - \gamma_{ij}\varepsilon && \text{(Using Lemma C.1)}\\
&\geq \sum_{i,j\in\{0,1\}} \gamma_{ij} \Pr_{\eta(w,t_1)}\left[f_{t_1}(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right] - O(\varepsilon) && \text{(Using γ_{ij} is a bounded constant)}\\
&= \mathrm{Util}_{\eta(w,t_1)}\left(f_{t_1},\gamma,z\right) - O(\varepsilon) && \text{(Using Definition 2.1)}\\
&= \mathrm{Util}_{\eta(w,t_1)}\left(f_{t_2},\gamma,z\right) - O(\varepsilon).
\end{aligned}$$

Where the last equality holds because f_{t_1} and f_{t_2} are feasible for Program (34) at the t_1-th iteration and f_{t_1} is the optimal solution of Program (34) at the t_1-th iteration. Proceeding with the above chain of inequalities, it follows that

$$\begin{aligned}
\mathrm{Util}_{\mu}\left(f_{t_1},\gamma,z\right) &\geq \mathrm{Util}_{\eta(w,t_1)}\left(f_{t_2},\gamma,z\right) - O(\varepsilon) && \text{(Using Definition 2.1)}\\
&= \sum_{i,j\in\{0,1\}} \gamma_{ij} \Pr_{\eta(w,t_1)}\left[f_{t_2}(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right] - O(\varepsilon)\\
&= \sum_{i,j\in\{0,1\}} \gamma_{ij} \left(\Pr_{\mu}\left[f_{t_2}(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right] - \varepsilon\right) - O(\varepsilon) && \text{(Using Lemma C.1)}\\
&= \sum_{i,j\in\{0,1\}} \gamma_{ij} \Pr_{\mu}\left[f_{t_2}(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right] - O(\varepsilon)\\
& && \text{(Using that γ_{ij} is a bounded constant)}\\
&= \mathrm{Util}_{\mu}\left(f_{t_1},\gamma,z\right) - O(\varepsilon). && \text{(Using Definition 2.1)}
\end{aligned}$$

The result follows as $\mathscr{H}$ holds with probability at least $1-O(\delta)$.

C.3. Proof of Theorem 4.3

In this section, we prove Theorem 4.3. For ease of reference, we restate Theorem 4.3 below.

Theorem 4.3 (Group-wise convergence to $f_{\mathrm{opt}}^{\alpha}$). *Suppose f_0 is $(\alpha-\varepsilon,\lambda)$-feasible. For any $\alpha,\varepsilon,\delta,\tau\in(0,1]$ and $\alpha_{\mathrm{exploit}}=\alpha-\varepsilon$, $\alpha_{\mathrm{explore}}=\varepsilon$, Algorithm 1 satisfies the following: if $t\geq {}^{1}/_{\sigma(z)}$ and $n\geq |D|\cdot \mathrm{poly}\left({}^{1}/_{\lambda}, {}^{1}/_{\tau}, {}^{1}/_{\min\{\varepsilon,\alpha-\varepsilon\}}\right)\cdot\log\left({}^{|D|}/_{\sigma\delta}\right)$ then with probability at least $1-\delta$, the utility of classifier f_t learned by the framework in t-th iteration is at least as large as the utility of $f_{\mathrm{opt}}^{\alpha}$ on samples in the z-th group drawn from μ, i.e.,*

$$\mathit{Util}_{\mu}\left(f_t,\gamma,z\right) \geq \mathit{Util}_{\mu}\left(f_{\mathrm{opt}}^{\alpha},\gamma,z\right)-\varepsilon.$$

Where the randomness at the t-th iteration is w.r.t. the randomness in $S_1,S_2,\ldots,S_t$ and Algorithm 1.

Since Assumption 2.3 holds with constants $\alpha,\lambda\in(0,1]$, f_0 satisfies the α-FDR constraint and has a selection rate of at least λ, i.e.,

$$\frac{\Pr_{\mu}\left[f_0(X,Z)\neq Y\right]}{\Pr_{\mu}\left[f_0(X,Z)=1\right]}\leq\alpha \quad\text{and}\quad \Pr_{\mu}\left[f_0(X,Z)=1\right]\geq\lambda. \tag{35}$$

Where we use $\Pr_{\mu}[\cdot]$ to denote $\Pr_{(X,Y,Z)\sim\mu}[\cdot]$ and $\mathbb{E}_{\mu}[\cdot]$ to denote $\mathbb{E}_{(X,Y,Z)\sim\mu}[\cdot]$. To prove Theorem 4.3, given $\varepsilon,\delta,\tau\in(0,1]$ and a function $g\colon D\times\mathcal{F}\to\mathbb{R}_{\geq 0}$, we need to show that if n and t are sufficiently large (Equation (36)), then with probability at least $1-\delta$, the classifier f_t learned by the data collection and prediction framework in Algorithm 1, satisfies the following inequality:

$$\forall_{z\in[p]},\quad \mathrm{Util}_{\mu}(f_t,\gamma,z)\geq \mathrm{Util}_{\mu}({f_{\mathrm{opt}}}^{(\alpha+\alpha')},\gamma,z)-\varepsilon.$$

Where $f_{\mathrm{opt}}^{(\alpha+\alpha')}$ is the optimal offline classifier defined in Equation 1. Theorem 4.3 claims that the following lower bounds on t and n are sufficient.

$$t\geq\frac{1}{\sigma(z)}\quad\text{and}\quad n\geq\frac{12\,|D|}{\tau\varepsilon^3}\cdot\frac{1}{\alpha_{\mathrm{explore}}^2\lambda}\cdot\log\frac{|D|}{\delta\sigma}. \tag{36}$$

The bounds on t and n serve different purposes. The lower bound on t ensures that $\textsc{Explore}_t$ has no samples from the z-th group. The lower bound on n ensures that the classifier f_t, trained over the samples in $\textsc{Exploit}_t$ has a high utility on samples from the z-th group drawn from the underlying distribution μ.

Fix any index $z \in [p]$. We first prove the first claim. Concretely, we will prove that with probability at least $1-\delta$, $\textsc{Explore}_t$ does not contain any samples (x, z) satisfying $\mu(x, z) \geq \frac{\varepsilon}{|D|}$. Fix any sample (x, z) from the z-th group in $\textsc{Explore}_0$. In each iteration t, where $(x, z) \in \textsc{Explore}_t$, its weight $w(x, z)$ increases by at least $\sigma(z)$ as the marginal distribution of L_t has density at least $\sigma(z)$ on (x, z). Hence, (x, z) can be in the exploration region for at most $\frac{1}{\sigma(z)}$ iterations. Thus, after $t \geq \frac{1}{\sigma(z)}$ iterations all items from the z-th group are in the exploitation region.

Now, we can bound the utility of f_t using Lemma C.1. Let $\mathscr{H}$ be the event in Lemma C.1, conditioned on $\mathscr{H}$, we have the following lower bounds

$$\begin{aligned}
\mathrm{Util}_\mu\left(f_t, \gamma, z\right) &= \sum_{i,j\in\{0,1\}} \gamma_{ij} \Pr_\mu\left[f_t(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right] && \text{(Using Definition 2.1)}\\
&\geq \sum_{i,j\in\{0,1\}} \gamma_{ij} \left(\Pr_{\eta_w}\left[f_t(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right]-\varepsilon\right) && \text{(Using Lemma C.1)}\\
&= \sum_{i,j\in\{0,1\}} \gamma_{ij} \Pr_{\eta_w}\left[f_t(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right]-O(\varepsilon) && \text{(Using γ_{ij} is a bounded constant)}\\
&= \mathrm{Util}_{\eta_w}\left(f_t, \gamma, z\right) - O(\varepsilon) && \text{(Using Definition 2.1)}\\
&= \mathrm{Util}_{\eta_w}\left(f_{\mathrm{opt}}^{(\alpha+\beta)}, \gamma, z\right) - O(\varepsilon).
\end{aligned}$$

Where the last equality is implied by the facts that $f_{\mathrm{opt}}^{(\alpha+\beta)}$ is feasible for Program (2), f_t is the optimal solution of Program (2), and $\mathrm{Util}_{\eta_w}(\cdot, \gamma, z)$ is the objective function of Program (2). Proceeding with the above chain of inequalities, we get the following lower bound.

$$\begin{aligned}
\mathrm{Util}_\mu\left(f_t, \gamma, z\right) &\geq \sum_{i,j\in\{0,1\}} \gamma_{ij} \left(\Pr_{\eta_w}\left[f_{\mathrm{opt}}^{(\alpha+\beta)}(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right]-\varepsilon\right)-O(\varepsilon)\\
& && \text{(Using Lemma C.1 and Definition 2.1)}\\
&\geq \sum_{i,j\in\{0,1\}} \gamma_{ij} \Pr_\mu\left[f_{\mathrm{opt}}^{(\alpha+\beta)}(X,Z)=i \text{ and } Y=j \text{ and } Z=z\right]-O(\varepsilon) && \text{(Using γ_{ij} is a bounded constant)}\\
&= \mathrm{Util}_\mu\left(f_{\mathrm{opt}}^{(\alpha+\beta)}, \gamma, z\right) - O(\varepsilon). && \text{(Using Definition 2.1)}
\end{aligned}$$

The result now follows as $\mathscr{H}$ holds with probability at least $1 - O(\delta)$.

D. Implementation Details

In this section, we provide additional details on how to implement Step 3 of Algorithm 1. Step 3 involves learning a classifier that satisfies the FDR constraints and the fairness constraints if required. The classifier optimizes a given utility measure, and we show to implement this program if the utility measure corresponds to accuracy or revenue first.

For accuracy, one can simply minimize a weighted logistic regression model over L_t, with FDR and fairness constraints. To implement this in practice (and for simulations in Section 5), we use Python's SLSQP program. Alternately, any common constrained optimization techniques can be employed here.

To learn a classifier that maximizes revenue, we first optimize a constrained optimization program to obtain a model that accurately assigns likelihoods to all individuals and then adjust the classifier threshold over these likelihoods to maximize revenue. We can accomplish this process in the following manner: Firs partition L_t into two equal random parts: $L_{t,1}$ and $L_{t,2}$. For learning likelihoods, we can again use constrained logistic regression, i.e., first learn the parameters $\omega \in \mathbb{R}^d$ of the logistic model (d is the number of features) which minimizes the weighted log-loss associated with ω over $L_{t,1}$, subject to the likelihoods satisfying the given α-FDR constraint. The constrained optimization program can again be implemented using Python's SLSQP function. Then, using $L_{t,2}$, choose a likelihood threshold in $[0, 1]$ (such that points with likelihood greater than the threshold are to be classified positively) which maximizes revenue$_{c_1,c_2}$. This method essentially corresponds to choosing an appropriate threshold from the model's ROC curve and has been used in other papers that optimize revenue for lending settings (Gramespacher and Posth, 2021).

E. Additional Details and Results for Adult and German Datasets

In this section, we provide the implementation details and additional results for the Adult and German datasets which were omitted from Section 5.

Description and pre-processing of Adult and German dataset. For the Adult dataset, each individual is characterized by the following features: age, class of worker, educational attainment, marital status, occupation, place of birth, usual hours worked per week past 12 months, gender, and race. We pre-process the dataset to ensure that it is limited to the subset with only individuals belonging to the races white/Caucasian and black/African-American. All features, other than the protected attribute are also scaled to have a mean of 0 and a standard deviation of 1.

For the German Credit dataset, the features are every individual's credit amount, duration, installment rate in percentage of disposable income, residential status, age, number of existing credits, number of people liable for, and gender. Once again, all features, other than the protected attribute are also scaled to have a mean of 0 and a standard deviation of 1.

Additional parameter details for our algorithm. We implement the constrained optimization program using the method described in Appendix D. For the SLSQP function (used to solve the optimization program), we use parameters ftol$= 1e^{-3}$ and eps$= 1e^{-3}$.

Implementation of g^{clf} and g^{fair} exploration strategies in Algorithm 1. As described in Section 5.1, the exploration function g is either $g^{\text{clf}}(x, z) \propto \beta + (1-\beta) f_t(x, z)$ or $g^{\text{fair}}(x, z) \propto (\beta + (1-\beta) f_t(x, z)) \cdot \Pr_\mu [Z = z \mid (X, Z) \in \textsc{Explore}_t]$, depending on whether exploration fairness is used or not. Both functions require a choice of β which ensures that every sample is assigned a non-zero exploration probability. We implement our approaches with $\beta = 0$, but instead of using binary outcome $f_t(x, z)$ in the above functions, we use the likelihood derived from the logistic model (described in Appendix D). Since the likelihood assigned by the logistic regression model is non-zero for every point, we ensure that g takes positive values and simultaneously use classifier performance to assign exploration probabilities.

Implementation details for baselines. We implement the baselines to correspond to our iterative setting and we mention the implementation details below.

OPT-OFFLINE. As mentioned earlier, the OPT-OFFLINE baseline is trained using the initial part of the split dataset, i.e., S_0. This baseline simply maximizes revenue subject to the α-FDR constraint and is implemented using Python's SLSQP function.

FAIR-CLF. The FAIR-CLF baseline is trained to test an only exploitation algorithm with statistical parity constraints. It maximizes revenue subject to the α-FDR constraint and statistical parity constraint and is trained over the available labeled dataset every iteration. This baseline is also implemented using Python's SLSQP function.

KILBERTUS ET AL. The algorithm from the Kilbertus et al. (2020) paper is implemented using the stochastic batch gradient descent method, as suggested in their paper. We also use the demographic parity regularizer employed in their paper. As they recommend, we also use a logistic regression model for the classifier in this algorithm. The parameter c in their algorithm is set to be 0.6 (similar to their experiments), the batch size is kept to 128, the learning rate is 0.01, number of iterations for gradient descent is also kept to 128. At every iteration, the parameters of the logistic model are updated using stochastic gradient descent over B elements randomly sampled from the recently labeled elements.

YANG ET AL. The algorithm from the Yang et al. (2022) paper is implemented the code provided by the authors [4]. We had to make certain modifications to the code to make it suitable for our setting and we list the modifications below. The full algorithm presented in Appendix C of their paper uses a while loop over the entire dataset. However, since the entire dataset is not available in our setting, we modify this step to a for loop over the iterations. In each iteration t, we provide their algorithm with the batch of samples S_t that arrive at the beginning of iteration t. All other components of their code are kept unchanged.

[4] https://github.com/Yifankevin/adaptive_data_debiasing

Table 2. Comparison of the overall performance of all methods on the Adult dataset with gender as the protected attribute. The average revenue per iteration across all repetitions, average FDR, and average acceptance rate disparity (statistical rate) are reported along with the standard deviation of all metrics in parenthesis.

	Protected attribute - Gender			
Method	Revenue (in thousands)	FDR	Stat. Rate	TPR Disparity
Algorithm 1- no fairness constraint	75.6 (15.0)	.15 (.02)	.09 (.02)	.15 (.04)
Algorithm 1- only exploit fairness	78.1 (12.4)	.15 (.02)	.07 (.02)	.07 (.03)
Algorithm 1- only explore fairness	77.9 (12.5)	.15 (.02)	.08 (.02)	.11 (.05)
Algorithm 1- both fairness constraints	78.0 (13.0)	.15 (.02)	.07 (.02)	.08 (.03)
Baseline - OPT-OFFLINE	80.4 (9.0)	.15 (.02)	.10 (.02)	.15 (.06)
Baseline - FAIR-CLF	72.4 (9.4)	.13 (.02)	.02 (.01)	.03 (.02)
KILBERTUS ET AL.	66.1 (12.1)	.20 (.03)	.15 (.02)	.23 (.04)
YANG ET AL.	-45.3 (11.6)	.47 (.08)	.09 (.02)	.02 (.01)
RATEIKE ET AL.	-17.1 (7.4)	.12 (.01)	.02 (.01)	.02 (.01)

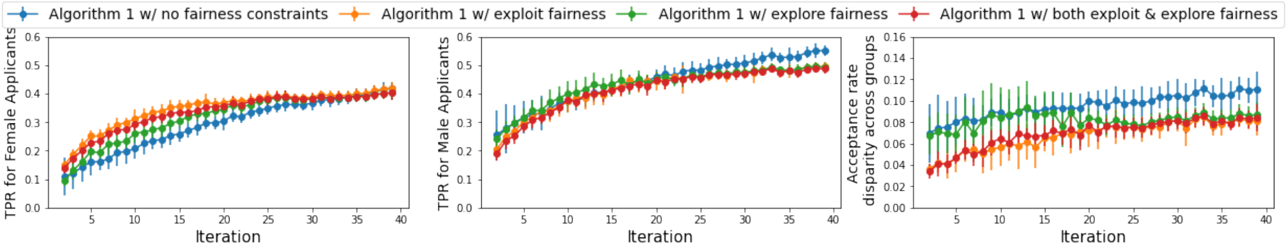

Figure 4. Iteration-wise performance of all variants of Algorithm 1 (with or without each of explore and exploit fairness constraints) on the Adult dataset with gender as the protected attribute. All algorithms can be seen to improve TPR for both groups.

RATEIKE ET AL. The algorithm from the Rateike et al. (2022) paper is implemented the code provided by the authors [5]. We execute this baseline using the initial data and batch size configuration specified in Section 5. All other components are kept unchanged and we use the same parameters as specified in the original code.

Additional results. *Performance comparison over the Adult dataset with gender as the protected attribute.* Overall performance on the Adult dataset for gender protected attribute is presented in Table 2 and iteration-wise performance is presented in Figure 4.

For gender, Algorithm 1 with exploit fairness achieves the lowest average statistical rate and true positive rate disparity and highest cumulative revenue among all variants. Algorithm 1 with both explore and exploit fairness also has similar performance. Hence, for both protected attributes, using both fairness constraints lead to high revenue while ensuring small performance disparities.

Figure 4 present the iteration-wise performance of our algorithms. The first two plots in the figure show that TPR increases with increasing iterations for all demographic groups. TPR is also generally larger in the initial iterations when using exploit fairness, but is overtaken by or is similar to the TPR of Algorithm 1 with no fairness constraints in the final iterations. Hence, fairness constraints can assist in accelerated data collection in the initial iterations, but once sufficient data is available, it seems to have a similar TPR as the variant with no fairness constraints. The third plot in Figure 4 also show that using exploit fairness results in the smallest statistical rate in all iterations.

Iteration-wise comparison of Algorithm 1 to baselines. We present plots for iteration-wise comparison of all methods and baselines with respect to the following metrics: cumulative revenue, FDR, TPR disparity, group-wise TPR, and statistical rate. We exclude the YANG ET AL and RATEIKE ET AL baselines from these plots as it achieves negative revenue and assigns positive prediction to a large fraction of samples, which makes it difficult to analyze the performance of other algorithms

[5] https://github.com/ayanmaj92/fairall

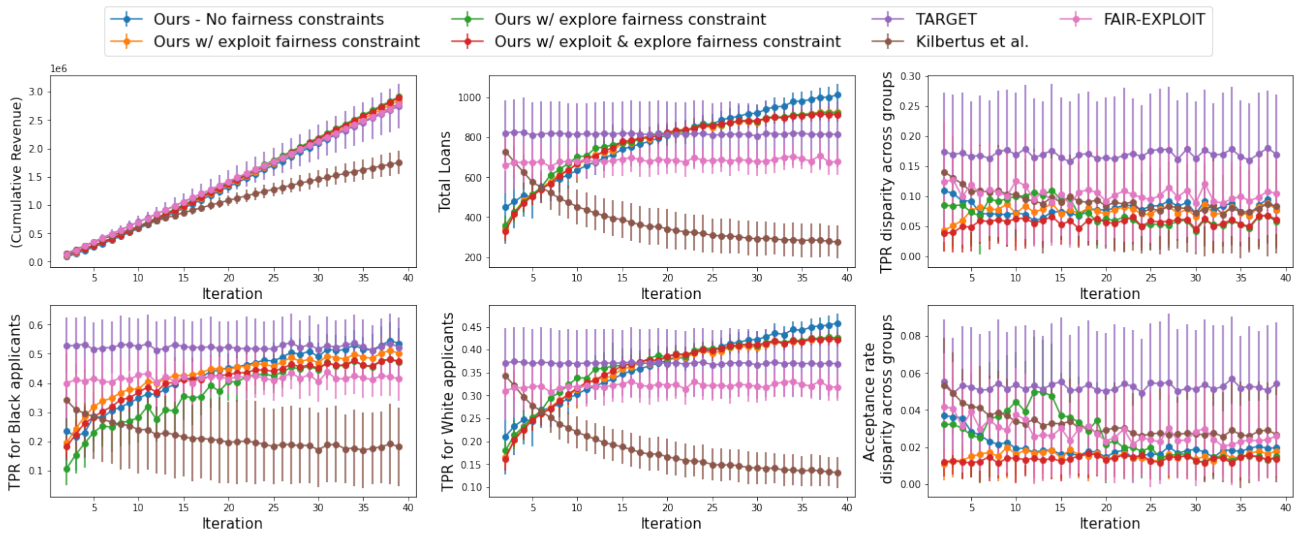

Figure 5. Performance of all versions of Algorithm 1 (with or without each of explore and exploit fairness constraints) and baselines on the Adult dataset with race as the protected attribute.

through the plots.

Figure 5 presents the performance of all methods over the Adult dataset with race as the protected attribute. Figure 6 presents the performance over the Adult dataset with gender as the protected attribute. Figure 7 presents the performance over the German dataset. All sets of plots show that the cumulative revenue from our methods are similar to the OPT-OFFLINE baseline across all iterations. Furthermore, unlike other methods, the TPR of our method consistently improves with increasing iterations. Baselines FAIR-CLF and KILBERTUS ET AL achieve high TPR in certain cases but in many settings, their TPR stagnates or decreases with increasing iterations.

Variation in performance of Algorithm 1 with α. The results presented in Section 5 used $\alpha = 0.15$ for the α-FDR constraint. We also present performance variation with respect to α over the Adult dataset. Figures 8 and 9 present the results for race and gender protected attributes respectively. As α increases, the framework is allowed to make more false positive errors which result in larger variability in revenue. However, across all iterations, our methods with appropriate fairness constraints result in low performance disparity across protected attribute groups.

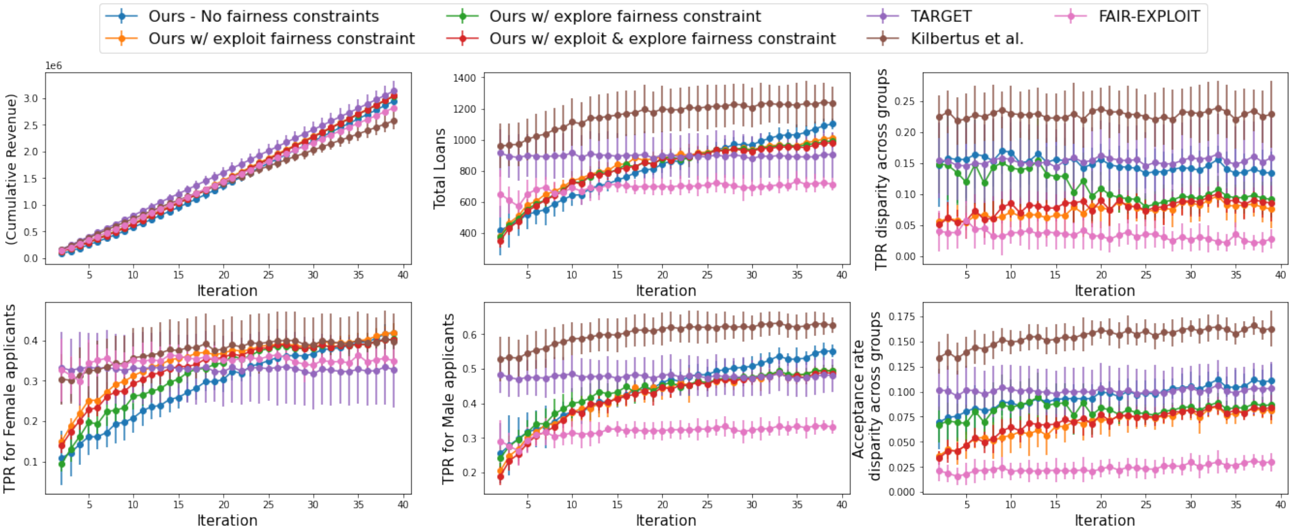

Figure 6. Performance of all versions of Algorithm 1 (with or without each of explore and exploit fairness constraints) and baselines on the Adult dataset with gender as the protected attribute.

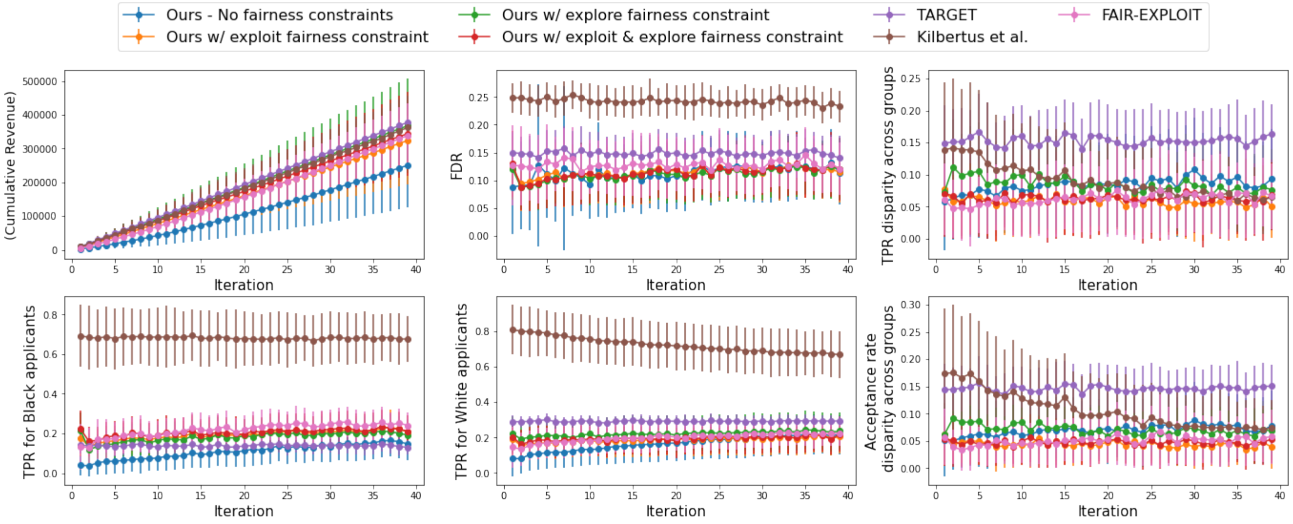

Figure 7. Performance of all versions of Algorithm 1 (with or without each of explore and exploit fairness constraints) and baselines on the German dataset with gender as the protected attribute.

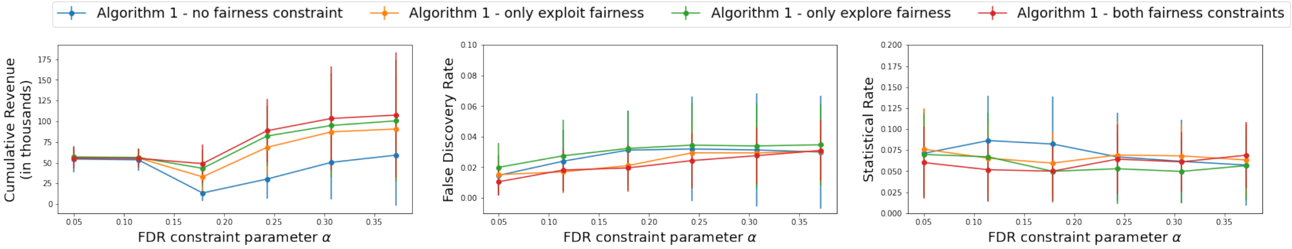

Figure 8. Performance of all versions of Algorithm 1 (with or without each of explore and exploit fairness constraints) with respect to different α parameters on the Adult dataset with race as the protected attribute.

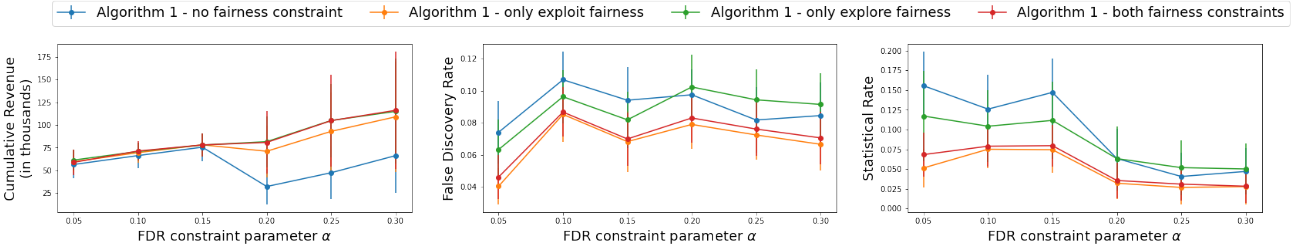

Figure 9. Performance of all versions of Algorithm 1 (with or without each of explore and exploit fairness constraints) with respect to different α parameters on the Adult dataset with gender as the protected attribute.