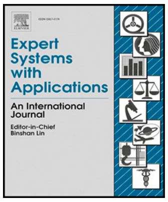
Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



# Fair active learning

Hadis Anahideh [a,*], Abolfazl Asudeh [a], Saravanan Thirumuruganathan [b]

[a] University of Illinois Chicago, United States of America
[b] QCRI, HBKU, Qatar

A R T I C L E I N F O

*Keywords:*
Active learning
Algorithmic fairness
Limited labeled data

A B S T R A C T

Machine learning (ML) is increasingly being used in high-stakes applications impacting society. Therefore, it is of critical importance that ML models do not propagate discrimination. Collecting accurate labeled data in societal applications is challenging and costly. Active learning is a promising approach to build an accurate classifier by interactively querying an oracle within a labeling budget. We introduce the fair active learning framework to carefully select data points to be labeled so as to balance model accuracy and fairness. To incorporate the notion of fairness in the active learning sampling core, it is required to measure the fairness of the model after adding each unlabeled sample. Since their labels are unknown in advance, we propose an expected fairness metric to probabilistically measure the impact of each sample if added for each possible class label. Next, we propose multiple optimizations to balance the trade-off between accuracy and fairness. Our first optimization linearly aggregate the expected fairness with entropy using a control parameter. To avoid erroneous estimation of the expected fairness, we propose a nested approach to maintain the accuracy of the model, limiting the search space to the top bucket of sample points with large entropy. Finally, to ensure the unfairness reduction of the model after labeling, we propose to replicate the points that truly reduce the unfairness after labeling. We demonstrate the effectiveness and efficiency of our proposed algorithms over widely used benchmark datasets using demographic parity and equalized odds notions of fairness.

## 1. Introduction

Data-driven decision making plays a significant role in modern societies by enabling wise decisions and to make societies more just, prosperous, inclusive, and safe. However, this comes with a great deal of responsibilities as improper development of data science technologies cannot only fail but make matters worse. Judges in US courts, for example, use criminal assessment algorithms that are based on the background information of individuals for setting bails or sentencing criminals. While it could potentially lead to safer societies, an improper usage could result in deleterious consequences on people's lives. For instance, the recidivism scores provided for the judges are highly criticized as being discriminatory, as they assign higher risks to African American individuals (Angwin, Larson, Mattu, & Kirchner, 2016).

Machine learning (ML) is at the center of data-driven decision making as it provides insightful unseen information about phenomena based on available observations. Two major reasons of unfair outcome of ML models are *Bias in training data* and *Proxy attributes*. The former is mainly due to the inherent bias (discrimination) in the historical data that reflects unfairness in society. For example, redlining is a systematic denial of services used in the past against specific racial communities, affecting historical data records (Jan, 2018). Proxy attributes on the other hand, are often used due to the limited access to labeled data, especially when it comes to societal applications. For example, when actual future recidivism records of individuals are not available, one may resort to information such as "prior arrests" that are easy to collect and use it as a proxy for the true labels, albeit a discriminatory one.

**Example 1.** A company is interested in creating a model for predicting recidivism to be used by judges when setting bails; they want to predict how likely a person is to commit a crime in the future. Suppose the company has access to the background information of some criminal defendants.[1] However, the collected data is not labeled. That is because there is no evidence available at the time of the trial if an individual will commit a crime in the future or not. Considering a time window, it is possible to label an individual in the dataset by checking the background of the individual within the time window after being released. However, it is costly as it might require expert efforts for data collection, integration and entity resolution.

* Corresponding author.
*E-mail addresses:* hadis@uic.edu (H. Anahideh), asudeh@uic.edu (A. Asudeh), s.thirumuruganathan@hbku.edu.qa (S. Thirumuruganathan).

[1] In the US, such information is provided by sheriff offices of the counties. For instance, for the COMPAS dataset, ProPublica used information obtained from the Sheriff Office of the Broward County. https://bit.ly/36CTc2F

https://doi.org/10.1016/j.eswa.2022.116981
Received 24 August 2021; Received in revised form 15 January 2022; Accepted 23 March 2022
Available online 4 April 2022

**Example 2.** A loan consulting company would like to create a model for financial agencies to identify "valuable customers" who will pay off their loans on time. The company has collected a dataset of customers who have received a loan in the past few years. The dataset includes information such as demographics, education and income level of individuals. Unfortunately, at the time of approving loans, it is not known whether customers will pay their debt on time, and hence, the data are not labeled. Nevertheless, the company has hired experts who, given the information of admitted applicants in the past, can verify their background and assess if payments were made on time. Of course, considering the costs associated with a background check, it is not viable to freely label all customers.

Both of the above examples use biased historical data for building their models. For instance, the `income` in Example 2 is known to include gender bias (Jones, 1983). Similarly, using `prior count` as a proxy attribute in Example 1 is racially biased (Angwin et al., 2016). Furthermore, the datasets in both cases are unlabeled.

A new paradigm of *fairness* in machine learning has emerged (Barocas, Hardt, & Narayanan, 2019) to address the unfairness issues of predictive outcomes. These work often assume the availability of (possibly biased) *labeled data* in sufficient quantity. When this assumption is violated, their performance degrades. In many practical societal applications such as Example 1, one operates in a constrained environment. Obtaining accurate labeled data is expensive, and could only be obtained in a limited amount. Training the model by using the (problematic) proxy attribute as the true label[2] will result in an unfair model.

Our goal is to develop efficient and effective algorithms for fair models in an environment where the budget for labeled data is restricted. An obvious baseline is to randomly select a subset of data (depending on the available budget), obtain their labels, and use it for training. However, a more sophisticated approach would be to use an adaptive sampling strategy. Active learning (Settles, 2012) is a widely used strategy for such a scenario. It sequentially chooses the unlabeled instances where their labeling is the most beneficial for the performance improvement of the ML model.

In this paper, we aim to develop an active learning framework that will yield fair(er) models. Fairness has different definitions and is measurable in various ways. Specifically, we consider a model fair if its outcome does not depend on sensitive attributes such as race or gender. We adopt demographic parity (aka statistical parity), one of the popular fairness measures (Dwork, Hardt, Pitassi, Reingold, & Zemel, 2012; Kusner, Loftus, Russell, & Silva, 2017).

**Summary of contributions.** We introduce *fairness in active learning* for constructing fair models in the context of limited labeled data. We propose a fair active learning (FAL) framework to balance model performance and fairness (Section 4). At a high-level, FAL uses an accuracy–fairness optimizer for selecting samples to be labeled. We propose three strategies for the optimizer, namely FAL $\alpha$-aggregate (Section 5.1), **FAL Nested** (Section 5.2), and **FAL Nested Append** (Section 5.3). Given that sample points are unlabeled in the context of AL, the optimizer uses *expected unfairness reduction*, proposed in Section 4. For the special case of generalized linear models, we propose a *fairness by covariance* (Section 6), an efficient alternative for expected unfairness reduction that reduces the asymptotic time complexity of FAL to be the *same* as traditional active learning. While our default notion of fairness is based on Demographic Parity, we provide an extension to other fairness models (Section 7). We conduct comprehensive experiments to evaluate the performance of our proposal on benchmark datasets (Section 8). Our results confirm that FAL can significantly reduce unfairness while not significantly impacting the model accuracy. In particular, our optimization **FAL Nested-Append** had on average a better performance, significantly reducing unfairness, while sacrificing a small amount on the accuracy.

## 2. Related work

**Algorithmic Fairness in ML.** Algorithmic fairness is a topic of extensive interest with (Barocas, Hardt, & Narayanan, 2017; Žliobaitė, 2017), and Mehrabi, Morstatter, Saxena, Lerman, and Galstyan (2019) providing surveys on discrimination and fairness in machine learning. Fairness, at a high level, is partitioned into individual fairness, which deals with discrimination against individuals, and group fairness, which considers parity over different demographic groups. While some works study individual fairness (Dwork et al., 2012), considering the social implications, most attention has been on group fairness. Kearns et al. (Kearns, Neel, Roth, & Wu, 2018) proposed the notion of rich subgroup fairness to bridge between group fairness and individual fairness. Probably the more popular notion of fairness is based on model *independence* or *demographic parity* (Barocas et al., 2019; Narayanan, 2018; Žliobaitė, 2017), also referred to by terms such as statistical parity (Dwork et al., 2012), and disparate impact (Barocas & Selbst, 2016). Model independence simply requires the sensitive characteristic to be statistically independent of the score (Barocas et al., 2019). There is a similarity between this model and diversity (Drosou, Jagadish, Pitoura, & Stoyanovich, 2017). In addition to independence, fairness can be defined using the notions of *separation* and *sufficiency* (Barocas et al., 2019). Considering a target variable (true label in classification) for every tuple in a supervised learning setting, the separation model allows correlation between the score and a sensitive attribute to the extent that it is justified by the target variable. Fairness measures such as *predictive equality*, *Equal opportunity*, and *Equalized odds* follow the separation model. Sufficiency model requires independence of a target variable and a sensitive attribute conditional to the scores. In other words, a score satisfies sufficiency if the sensitive attribute and target variable are clear from the context. *Predictive parity* is an example fitting into this model.

The goal of improving fairness in learning problems can be achieved by intervention at pre-processing, in-processing(algorithms), or post-processing strategies. Pre-processing strategies involve the fairness measure in the data preparation step to mitigate the potential bias in the input data and produce fair outcomes (Asudeh, Jin & Jagadish, 2019; Asudeh, Shahbazi, Jin, & Jagadish, 2021; Calmon, Wei, Vinzamuri, Ramamurthy, & Varshney, 2017; Celis, Keswani, & Vishnoi, 2020; Feldman, Friedler, Moeller, Scheidegger, & Venkatasubramanian, 2015; Kamiran & Calders, 2012; Krasanakis, Spyromitros-Xioufis, Papadopoulos, & Kompatsiaris, 2018; Salimi, Rodriguez, Howe, & Suciu, 2019; Soen, Husain, & Nock, 2020; Zemel, Wu, Swersky, Pitassi, & Dwork, 2013). In-process approaches (Asudeh, Jagadish, Stoyanovich & Das, 2019; Calders & Verwer, 2010; Corbett-Davies, Pierson, Feller, Goel, & Huq, 2017; Fish, Kun, & Lelkes, 2016; Goh, Cotter, Gupta, & Friedlander, 2016; Komiyama, Takeda, Honda, & Shimao, 2018; Xu, Yuan, Zhang, & Wu, 2018; Zafar, Valera, Rodriguez, & Gummadi, 2015) incorporate fairness in the design of the algorithm to generate a fair outcome. Existing works have formulated fairness in classification as a constrained optimization (Celis, Huang, Keswani, & Vishnoi, 2019; Hardt, Price, Srebro, et al., 2016; Huang & Vishnoi, 2019; Menon & Williamson, 2018; Zafar, Valera, Gomez Rodriguez, & Gummadi, 2017; Zhang, Chu, Asudeh, & Navathe, 2021). Post-process methods (Hardt et al., 2016; Hébert-Johnson, Kim, Reingold, & Rothblum, 2017; Kim, Ghorbani, & Zou, 2018; Pleiss, Raghavan, Wu, Kleinberg, & Weinberger, 2017; Stoyanovich, Yang, & Jagadish, 2018; Zehlike et al., 2017), manipulate the outcome of the algorithm to mitigate the unfairness of the outcome for the decision making process.

*Our proposal is orthogonal to the fair ML literature.* While our goal in this paper is on selecting the samples to be labeled, fair ML algorithms aim to build fair models for a given set of labeled samples.

This paper is the first to introduce fair active learning. Besides this paper, we are aware of one *subsequent* work that considers fairness in active learning (Sharaf & Daumé III). The setting of this work differs from the standard active learning setting: instead of seeking

[2] In the rest of the paper we refer to true label as label.

**Table 1**
Table of notations

| Symbol | Description |
|---|---|
| $\mathcal{U}$ | The pool of unlabeled data |
| $\mathcal{L}$ | The pool of labeled data |
| $\mathbf{x}$ | The input features (observation) vector |
| $d$ | Number of input features |
| $y$ | True label, aka target variable |
| $\hat{y}$ | The model prediction |
| $s$ | The sensitive attribute |
| $B$ | Labeling budget |
| $K$ | Cardinality of $y$ |
| $C(.)$ | A classifier that maps input features vector to a predicted label |
| $\mathcal{F}(C)$ | The unfairness of a classifier $C$ |
| $E[\mathcal{F}_t^i]$ | Expected unfairness if labeling $\mathbf{x}^{(i)} \in \mathcal{U}$ and adding it to $\mathcal{L}$ at iter $t$ |

to minimize the number of labeled data, Sharaf and Daumé III starts with a pre-existing labeled data and seek to minimize the disparity by labeling *additional* samples. In contrast, we tackle the traditional active learning setting with no labeled data. Fairness has also been studied in few works that consider the intersection of fairness and active feature acquisition (Bakker et al., 2020; Noriega-Campero, Bakker, Garcia-Bulle, & Pentland, 2019). Our work is orthogonal to this research since our goal is not feature acquisition but rather active learning.

**Active Learning.** Active learning is the preferred learning strategy in limited labeled data settings, where collecting new labels is costly. Different active learning scenarios (Membership Query Synthesis (Angluin, 1988; Cohn, Ghahramani, & Jordan, 1996; King et al., 2004), Stream-Based Selective Sampling (Cohn, Atlas, & Ladner, 1994; Dagan & Engelson, 1995; Dasgupta, Hsu, & Monteleoni, 2007; Mitchell, 1982; Moskovitch et al., 2007), Pool-Based Active Learning (Hoi, Jin, & Lyu, 2006; McCallumzy & Nigamy, 1998; Settles & Craven, 2008; Tong & Koller, 2001; Tur, Hakkani-Tür, & Schapire, 2005)) and sampling strategies (Uncertainty Sampling (Lewis & Gale, 1994; Settles & Craven, 2008; Shannon, 2001), Query-By-Committee (Gilad-Bachrach, Navot, & Tishby, 2006; Melville & Mooney, 2004; Seung, Opper, & Sompolinsky, 1992), Expected Model Change (Freytag, Rodner, & Denzler, 2014; Roy & McCallum, 2001; Settles, Craven, & Ray, 2007), Variance Reduction (Cohn et al., 1996; Hoi et al., 2006), etc.) have been proposed and are surveyed in Settles (2012). Similarly (Kumar & Gupta, 2020) reviews recent advancements in the area of active learning. Uncertainty sampling is one of the most popular approaches for active learning (Balcan, Broder, & Zhang, 2007; Lewis & Catlett, 1994; Tong & Koller, 2001), which merely selects data points based on the single objective function of informativeness. There are several active learning approaches proposed to incorporate more than one criteria for sampling, such as representativeness (Donmez, Carbonell, & Bennett, 2007; Huang, Jin, & Zhou, 2010; Xu, Yu, Tresp, Xu, & Wang, 2003). Other advanced sampling strategies based on deep learning has also been proposed for active learning (Wu, Chen, Zhong, Wang, & Shi, 2021).

## 3. Background

In this section, we introduce the data model, the active learning framework with uncertainty sampling heuristic, and fairness model.

### 3.1. Learning model

Given a classifier and a pool of unlabeled data $\mathcal{U}$, Active Learning (AL) identifies the data points to be labeled so that an accurate model could be learned as quickly as possible. $\mathcal{U}$ is assumed to be an independent and identically distributed (i.i.d) sample set collected from the underlying unknown distribution. For each data point $\mathbf{p}_i \in \mathcal{U}$, we use the notation $\mathbf{x}^{(i)}$ for the $d$-dimensional vector of input features and $\mathbf{x}_j^{(i)}$ to refer to the value of $j$th feature. Each data point is associated with a non-ordinal categorical sensitive attribute $\mathbf{s}$ such as `gender` and `race`. We use the notation $s^{(i)}$ to refer to the sensitive attribute of $\mathbf{p}_i$. We also use $y^{(i)}$ to refer to the label of a point $\mathbf{p}_i$ with $K$ possible values $\{0, \dots, K-1\}$. The notations used in this paper are listed in Table 1.

**Algorithm 1 Active Learning (with uncertainty sampling)**

```
for $t = 1$ to $B$ do
    $\mathbf{x}^* = \underset{\mathbf{x} \in \mathcal{U}}{\text{argmax}}\ \mathcal{H}(y|\mathbf{x}, \mathcal{L})$
    $y$ = label $\mathbf{x}^*$ using the labeling oracle
    add $\langle \mathbf{x}^*, y \rangle$ to $\mathcal{L}$
    train the classifier $C_t$ using $\mathcal{L}$
end for
return $C_B$
```

Without loss of generality and to simplify the explanations, unless explicitly stated, we assume $\mathbf{s}$ is a single sensitive attribute. Still, we would like to emphasize that our techniques are not limited to the number of sensitive attributes.

The goal is to learn a classifier function $C : \mathbb{R}^d \to [0, K-1]$ that maps the feature space $\mathbf{X}$ to the labels $\mathbf{y}$. Let $\hat{\mathbf{y}} = C(\mathbf{x})$ be the predicted label for $\mathbf{x}$. Pool-based active learning (Lewis & Gale, 1994), sequentially selects instances from $\mathcal{U}$ to be labeled by an *expert oracle* and forms a labeled set $\mathcal{L}$ for training. Labeling, however, is costly and usually there is a *limited labeling budget* $B$. The challenge is to design an effective sampling strategy that wisely utilizes the budget to build the most accurate model. Uncertainty sampling (Lewis & Gale, 1994), a widely used strategy, chooses the point $\mathbf{p} \in \mathcal{U}$ that the current model is least certain about its label. The classifier $C_{t-1}$ for iteration $t$ chooses the data point that maximizes the Shannon entropy ($\mathcal{H}$) (Shannon, 1948) over the label probabilities.

$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathcal{U}}{\text{argmax}}\ \mathcal{H}(y|\mathbf{x}, \mathcal{L}) \tag{1}$$

Algorithm 1 presents the standard active learning algorithm. Using Eq. (1), the active learning algorithm iteratively selects a point from $\mathcal{U}$ to be labeled next. It uses the classifier trained in the previous step $C_t$ to obtain probabilities of the labels. The algorithm obtains the label from the labeling oracle, and adds the point to the set of labeled dataset $\mathcal{L}$, using it to train the classifier $C_t$. This process continues until the labeling budget is exhausted.

### 3.2. (Un)fairness model

We develop our fairness model on the notion of *model independence* or *demographic parity* (DP) (Barocas et al., 2019; Zafar et al., 2017; Žliobaitė, 2017), also referred by terms such as statistical parity (Dwork et al., 2012; Simoiu, Corbett-Davies, Goel, et al., 2017), and disparate impact (Barocas & Selbst, 2016; Feldman et al., 2015). Although our focus in this paper is on fairness based on model independence, in Section 7 we show how to extend our framework for other measures based on separation ($\hat{y} \perp s \mid y$) and sufficiency ($y \perp s \mid \hat{y}$) (Barocas et al., 2019). Given a classifier $C$ and a random point $\langle \mathbf{x}, s \rangle$ with a predicted label $\hat{y} = C(\mathbf{x})$, DP holds iff $\hat{y} \perp s$ (Barocas et al., 2017, 2019). For a binary classifier, let $\hat{y} = 1$ count as "acceptance" (such as receiving a loan). DP requires that the acceptance rate be the same for all groups of $S$ i.e. female or male in this case. For a binary classifier and a binary sensitive attribute, the statistical independence of a sensitive attribute from the predicted label induces the following notions for DP:

1. $\mathbb{P}(\hat{y} = 1|s = 0) = \mathbb{P}(\hat{y} = 1|s = 1)$: The probability of acceptance is equal for members of different demographic groups. For instance, in Example 1 members of different race groups have an equal chance for being classified as low risk.
2. $\mathbb{P}(s = 1|\hat{y} = 1) = \mathbb{P}(s = 1)$: If the population ratio of a particular group is $\rho$ (i.e. $\mathbb{P}(s = 1)$), the ratio of this group in the accepted class is also $\rho$. For instance, in Example 2, let $\rho$ be the female ratio in the applicants' pool. Under DP, female ratio in the set of admitted applications for a loan equals to $\rho$.

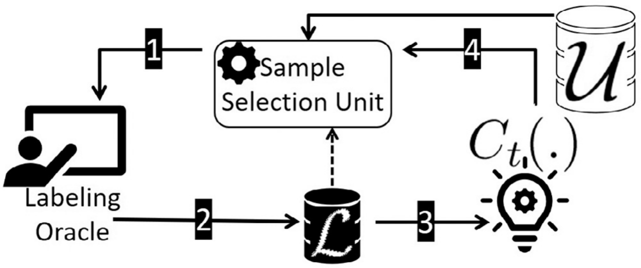


**Fig. 1.** FAL framework.

3. $I(\hat{y}; s) = 0$: Mutual information is the measure of mutual dependence between two variables. When $y$ and $s$ are independent, their mutual information is zero. That is, the conditional entropy $\mathcal{H}(s|y)$ is equal to $\mathcal{H}(s)$.
4. $cov(\hat{y}, s) = 0$: When $\hat{y}$ and $\mathbf{s}$ are independent, $cov(\hat{y}, \mathbf{s})$ is equal to zero.

A disparity (or unfairness) *measure* can be defined using any of the above notions. The absolute differences or the ratio of probabilities in bullets 1 or 2 provide four disparity measurements. Mutual information and covariance (or correlation) provide two natural measures, since the greater the absolute value of the two is the greater the disparity. In this paper, we do not limit ourselves to any of the unfairness measures (demographic disparity) and give the user the freedom to provide a customized measure. We denote (user-provided) measure of unfairness as $\mathcal{F}(s, C)$. When $s$ is clear by context, we simplify it to $\mathcal{F}(C)$.

## 4. Fair active learning (FAL) framework

By carefully selecting samples to be labeled, AL has the potential to mitigate algorithmic bias by incorporating the fairness measure into its sampling process. Still, not considering fairness while building models can result in model unfairness. As a naïve resolution, one could decide to drop the sensitive attribute from the training data. This, however, is not sufficient since the bias in the features may cause model unfairness (Buolamwini & Gebru, 2018; Zou & Schiebinger, 2018). Hence, a smart sampling strategy is needed to mitigate the bias. Blindly optimizing for fairness could result in an inaccurate model. For instance, in Example 1, consider a model that randomly classifies individuals as high-risk. This model indeed satisfies demographic parity since the probability of the outcome is (random and therefore) independent of $S$. However, such a model provides zero information about how risky an individual is.

We propose the Fair Active Learning (FAL) framework to balance between accuracy and fairness. FAL is an iterative approach similar to standard active learning approaches. As shown in Fig. 1, the central component of FAL is the sample selection unit (SSU) that chooses an unlabeled point $\langle \mathbf{x}^{(i)}, s^{(i)} \rangle$ from $\mathcal{U}$ and obtains the label from an oracle. The labeled point $\langle \mathbf{x}^{(i)}, s^{(i)}, y^{(i)} \rangle$ is moved to $\mathcal{L}$, the set of labeled points that is used to train $C_t$. In the next iteration, $t+1$, SSU employs $C_t$ and selects the next point to be labeled. This process continues until the budget for labeling is exhausted.

At a high level, SSU can be viewed as two computation blocks stacked on top of each other. The upper block is the fairness–accuracy optimizer that selects a point from $\mathcal{U}$ to be labeled next such that a combination of accuracy (misclassification reduction) and fairness (unfairness reduction) is maximized. We shall provide the details of this block in Section 5.

The accuracy–fairness optimizer relies on the lower block for estimating unfairness values. Let $C_{t-1}$ be the current model, created in the previous iteration $t-1$. In order to evaluate the unfairness reduction after labeling a sample point $\langle \mathbf{x}^{(i)}, s^{(i)} \rangle \in \mathcal{U}$, the optimizer block needs to compute the unfairness of the current model $\mathcal{F}(C_{t-1})$, as well as the unfairness of the model after labeling the sample point $\mathcal{F}(C_t^i)$. Computing $\mathcal{F}(C_t^i)$ turns out to be problematic as at the time of evaluating the candidate points in $\mathcal{U}$, we still do not know their labels. On the other hand, to evaluate $\mathcal{F}(C_t^i)$, we need to know what the model parameters will be after labeling the point $\langle \mathbf{x}^{(i)}, s^{(i)} \rangle$ and adding it to $\mathcal{L}$. In other word, in order to evaluate a point to whether or not it should be labeled, we need to know its label in advance! This contradicts with the fact that $\mathcal{U}$ is unlabeled.

To resolve this issue, using a decision theoretic approach (Settles, 2012), we consider the *expected unfairness reduction*: selecting the point that is expected to impart the largest reduction to the current model unfairness, *after acquiring its label*. Therefore, instead of $\mathcal{F}(C_t^i)$, we plug in the expected fairness $E\left[\mathcal{F}_t^i\right]$. In this way, we are approximating the expected future fairness of a model using $\mathcal{L} \cup \{\langle \mathbf{x}^{(i)}, s^{(i)}, y^{(i)} \rangle\}, \forall\, \mathbf{x}^{(i)} \in \mathcal{U}$ *over all possible labels* under the current model. Consider a point $\mathbf{p}_i \in \mathcal{U}$ and let $C_t^{i,k}$ be the model after adding $\mathbf{p}_i$ to $\mathcal{L}$ if its true label is $y^{(i)} = k$. Inevitably, SSU does not know the label in advance. Hence, it must instead calculate the unfairness as an expectation over the possible labels. That is, it considers the fairness reduction *IF* the label was $C_0$ (class 0), $C_1$ (class 1), $\ldots, C_{K-1}$ (class $K-1$), and aggregate the values into an expected value. Eq. (2) denotes the expected (un)fairness computation used by SSU:

$$E\left[\mathcal{F}_t^i\right] = \sum_{k=0}^{K-1} \mathcal{F}(C_t^{i,k})\mathbb{P}(y = k|\mathbf{x}^{(i)}) \tag{2}$$

Using Fig. 2 for explanation, for every point $\mathbf{p}_i = \langle \mathbf{x}^{(i)}, s^{(i)} \rangle$ in the unlabeled pool, SSU considers different values of $\{0, \ldots, K-1\}$ as possible labels for $\mathbf{p}_i$. For every possible label $y_k$, it updates the model parameters to the intermediate model $C_t^{i,k}$ using $\mathcal{L} \cup \{\langle \mathbf{x}^{(i)}, s^{(i)}, k \rangle\}$.

**Algorithm 2 ExpF**

```
Input: $\langle \mathbf{x}^{(i)}, s^{(i)} \rangle, \mathcal{L}, C_{t-1}$
$sum = 0$
for $k = 0$ to $K-1$ do
    train $C_t^{i,k}$ using $\mathcal{L} \cup \{\langle \mathbf{x}^{(i)}, s^{(i)}, y_k \rangle\}$
    compute $\mathcal{F}(C_t^{i,k})$, using $I\mathcal{U}$
    $sum = sum + \mathcal{F}(C_t^{i,k})\mathbb{P}(y = k|\mathbf{x}^{(i)})$
end for
return $sum$
```

Since the points to be labeled are selective samples from $\mathcal{U}$, and moved from $\mathcal{U}$ to $\mathcal{L}$, after the process starts, neither $\mathcal{U}$ nor $\mathcal{L}$ can be seen as i.i.d. samples of the actual underlying distribution, and therefore, cannot be used to estimate the fairness. However, $\mathcal{U}$ initially follows the underlying distribution. Therefore, to create a dataset for evaluating the fairness of the model for sample selection, we utilize the initial unlabeled pool $\mathcal{U}$ (referred as $I\mathcal{U}$) and use it in different FAL iterations. Following the standard AL, at every iteration, for every possible label for a point $\langle \mathbf{x}^{(i)}, s^{(i)} \rangle$, SSU uses the current model $C_{t-1}$ for calculating $\mathbb{P}(y = k|\mathbf{x}^{(i)})$. Note that each of the $K$ models is used only to compute $\mathcal{F}(C_t^{i,k})$ in Eq. (2).

Algorithm 2 shows the pseudo-code of for computing the expected unfairness. In order to compute the expected unfairness, Algorithm 2 requires to train $K$ models, each for a possible label $k$ for the point $\langle \mathbf{x}^{(i)}, s^{(i)} \rangle$, to compute $\mathcal{F}(C_t^{i,k})$, which makes it inefficient. In Section 6, we propose a *constant-time* approximation alternative that enables the *same* asymptotic time complexity as traditional active learning. We shall then provide the details of extending the fairness block for other measures of fairness beyond DP in Section 7.

Having discussed the FAL framework and the unfairness estimation block, we will next describe the accuracy–fairness optimization block used by SSU for selecting the next sample to be labeled.

## 5. Accuracy–fairness optimization

Having introduced the FAL framework, in this section, we propose multiple optimizations to balance the trade-off between accuracy and

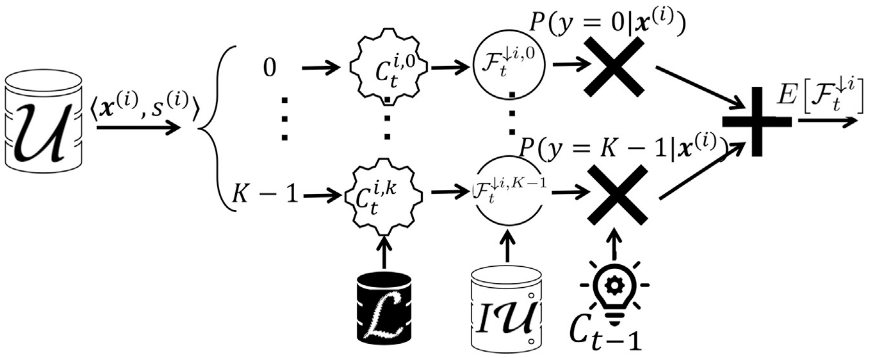

$\mathcal{U}$
$\langle x^{(i)}, s^{(i)}\rangle$
0
$K-1$
$C_t^{i,0}$
$C_t^{i,k}$
$\mathcal{F}_t^{\downarrow i,0}$
$\mathcal{F}_t^{\downarrow i,K-1}$
$P(y=0|x^{(i)})$
$P(y=K-1|x^{(i)})$
$E[\mathcal{F}_t^{\downarrow i}]$
$\mathcal{L}$
$I\mathcal{U}$
$C_{t-1}$

**Fig. 2.** Computing expected unfairness for $\langle \mathbf{x}^{(i)}, s^{(i)}\rangle \in \mathcal{U}$.

fairness. We treat the active learning optimizer for accuracy as a black box. That is, we assume the existence of a metric (such as entropy) that quantifies the desirability of a data point for traditional active learning. Nevertheless, to simplify the explanations, we use uncertainty sampling as a representative active learning method for the accuracy optimizer. To adapt for other accuracy optimizers, one should simply replace the accuracy term (entropy) with the proper alternative.

### 5.1. FAL α-aggregate

Similar to AL, FAL is also an iterative process that selects a sample from the unlabeled pool $\mathcal{U}$ to be labeled and added to the labeled pool $\mathcal{L}$. However, FAL $\alpha$-aggregate considers a combination of unfairness and misclassification error reduction as the optimization objective for the sampling step. Specifically, for a sample point $\langle \mathbf{x}^{(i)}, s^{(i)}\rangle \in \mathcal{U}$, we consider the Shannon entropy measure $\mathcal{H}_{t-1}(y^{(i)})$ for misclassification error, while considering demographic disparity $\mathcal{F}(C_t^i)$ for unfairness — $C_t^i$ is the classifier trained on $\mathcal{L}$ at iteration $t$, after labeling the point $\langle \mathbf{x}^{(i)}, s^{(i)}\rangle$ and $\mathcal{H}_{t-1}(y^{(i)})$ is the entropy of the $y^{(i)}$ based on the current model $C_{t-1}$. The formulation can be viewed as a multi-objective optimization for fairness and misclassification error. Another perspective is to view the fairness as a regularization term to the optimization. Eq. (3) is consistent with both of these views and is therefore considered in our framework.

$$\underset{\langle \mathbf{x}^{(i)}, S^{(i)}\rangle \in \mathcal{U}}{\text{argmax}} \ \alpha\, \mathcal{H}_{t-1}(y^{(i)}) + (1-\alpha)\big(\mathcal{F}(C_{t-1}) - \mathcal{F}(C_t^i)\big) \tag{3}$$

$(\mathcal{F}(C_{t-1}) - \mathcal{F}(C_t^i))$ is the unfairness reduction (fairness improvement) term and the coefficient $\alpha \in [0,1]$ is the user-provided parameter that determines the trade-off between the model fairness and model performance. Values closer to 1 put greater emphasize on model performance, while smaller values of $\alpha$ put greater importance on fairness. As we shall elaborate in Section 8, entropy and fairness values are standardized to the same scale before being combined in Eq. (3).

As we discussed earlier, computing the fairness for each unlabeled sample points, we consider the expected fairness $E\big[\mathcal{F}_t^i\big]$, which is returned by Algorithm 2. Hence, in Eq. (3), the second component of the objective function is modified to $(1-\alpha)\big(\mathcal{F}(C_{t-1}) - E\big[\mathcal{F}_t^i\big]\big)$.

While $\alpha$ controls the trade-off between the accuracy (entropy) and the fairness terms, selecting an appropriate $\alpha$ value might not be clear for the user. More importantly, FAL might find it challenging to use a fixed learning strategy based on $\alpha$ in different iterations. To avoid parameter tuning in active learning instead of using a fixed $\alpha$ for all iterations, one can use a *decay function* that begins with a large value of $\alpha$, which improves the accuracy of the model. As the model becomes more stable, the value of $\alpha$ gets dropped, putting more weight on improving fairness.

In initial iterations, the model is not accurate because it was trained only on a few labeled instances, resulting in possibly inaccurate estimates of the label probabilities for a given unlabeled instance from $\mathcal{U}$. Computing the expected fairness values relies heavily on the probabilities of the label. The miscalculation of these probabilities leads to an inaccurate estimation of fairness; such erroneous values contribute to the selection of points that do not support (and may even deteriorate) the fairness of the model and may not be good for model accuracy. In the later iterations, the model may already be stable and accurate, and new labeled points may not significantly impact its accuracy. However, the model can provide better estimations of the label probabilities, which results in more robust estimations of the expected fairness.

**Algorithm 3** FAL α-aggregate

```
for $t = 1$ to $B$ do
    $max = 0$
    for $i = 1$ to $|\mathcal{U}|$ do
        $H = -\sum_{k=0}^{K-1} \mathbb{P}(y=k|\mathbf{x}^{(i)}) \log \mathbb{P}(y=k|\mathbf{x}^{(i)})$
        $E\big[\mathcal{F}_t^i\big] = \mathbf{ExpF}\big(\langle \mathbf{x}^{(i)}, s^{(i)}\rangle, \mathcal{L}, C_{t-1}\big)$
        $obj = \alpha(i)H + (1-\alpha(i))(\mathcal{F}(C_{t-1}) - E\big[\mathcal{F}_t^i\big])$
        if $obj > max$ then
            $max = obj$
            $\langle \mathbf{x}^*, s^*\rangle = \langle \mathbf{x}^{(i)}, s^{(i)}\rangle$
        end if
    end for
    $y^*$ = label $\mathbf{x}^*$ using the labeling oracle
    move $\langle \mathbf{x}^*, s^*, y^*\rangle$ to $\mathcal{L}$
    train the classifier $C_t$ using $\mathcal{L}$
end for
return $C_t$
```

This concept is applied in different context, such as assigning learning rate (Schaul, Zhang, & LeCun, 2013), where a larger value is used initially that gradually decreases over time. Our approach is agnostic to the decay function used. In the experiments, we use a function that linearly interpolates between the range [0,1].

Given the scarcity of labels in the active learning paradigm, it is not possible to identify an appropriate value of $\alpha$. In other words, even though $\alpha$ is an important hyper-parameter, it cannot be tuned due to the scarcity of the labels. Instead, we advocate for an adaptive setting where the value of $\alpha$ is graduated varied so as to trade-off both accuracy and fairness. In Section 8, we conduct extensive experiments to study the impact of $\alpha$ fixing it to a value withing a wide range of [0.1,0.9], as well as adaptive $\alpha$ approach. Besides, to propose more effective solutions, after proposing the $\alpha$-aggregate approach, we introduced FAL-Nested (and later FAL-Nested-Append).

The pseudo-code of FAL with $\alpha$-aggregate is provided in Algorithm 3.

### 5.2. FAL nested

Accurately estimating the expected unfairness reduction is critical for the performance of FAL. Looking at Eq. (2), computing the expected

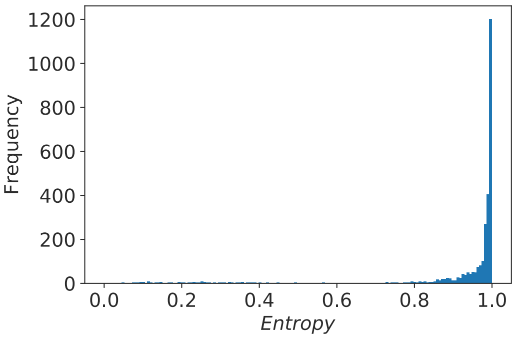


**Fig. 3.** Distribution of Entropy for COMPAS dataset.

fairness directly depends on the probability estimations of the current model $C_{t-1}$ for different class labels $k$. The miscalculation of these probabilities leads to an inaccurate estimation of fairness. Sacrificing accuracy for fairness in previous steps will affect the estimation of expected fairness in subsequent steps. Such erroneous estimation of the expected unfairness improvement as a result, contribute to the selection of points that do not support (and may even deteriorate) the fairness of the model and may not be good for model accuracy. In order to prevent this phenomena, it is necessary to always maintain an accurate intermediate model – i.e., to ensure that the selected points to be labeled will not negatively impact the accuracy of the model.

Fortunately, we make an observation in practice that helps us keeping the intermediate models accurate while achieving fairness. As we shall further explain in the following, our observation, also helps us to even reduce the computation cost of the model by only focusing on a subset of $\mathcal{U}$, instead of the full set.

It turns out in practice the distribution of the entropy of the data points in $\mathcal{U}$ is left-skewed, having a large number of points with entropy close to the point with maximum entropy. Therefore, all these points are almost equally good candidates from the accuracy perspective. We observed this in our real experiments, including the one on COMPAS dataset shown in Fig. 3. In the figure, one can observe that the last bucket contains a relatively large pool of points with entropy close to the maximum value. As a result, the top bucket of high-entropy samples provide "good" alternatives for the point with highest entropy, selected by AL (Eq. (1)). Hence, the "regret" of selecting either of the alternatives compared to the top-1 point should be small. On the other hand, the relatively large size of the set (as observed in Fig. 3) increases the chance that it contains a point with a high potential for reducing unfairness. This is because the expected unfairness reduction monotonically increases by the size of the set. To see why, consider two sets $S' \subset S$. Let $p_j$ be the point with maximum expected unfairness reduction in $S'$ and $p_i$ be the one in $S$. Since every point in $S'$ also belongs to $S$, $\left(\mathcal{F}(C_{t-1}) - E\left[\mathcal{F}_t^i\right]\right) \geq \left(\mathcal{F}(C_{t-1}) - E\left[\mathcal{F}_t^j\right]\right)$.

We use the above observation to design a *nested optimization* for accuracy and fairness. In particular, instead of computing a score by linearly combining the two terms (Eq. (3)), we apply a nested optimization where in the first level, we select a subset of $\mathcal{U}_A \subset \mathcal{U}$ of the top-$\ell$ points[3] that maximize the entropy $\mathcal{H}(y^{(i)})$ using the current classifier $C_{t-1}$:

$$\mathcal{U}_A = \ell\text{-}\underset{\langle \mathbf{x}^{(i)}, s^{(i)} \rangle \in \mathcal{U}}{\operatorname{argmax}} \mathcal{H}(y|\mathbf{x}^{(i)}, \mathcal{L}) \tag{4}$$

where the function $\ell$-argmax returns $\ell$ samples with maximum values. The first optimization level only optimizes for accuracy, to ensure maintaining a high accuracy for the intermediate model. Next, changing the

**Algorithm 4** FAL Nested

```
for $t = 1$ to $B$ do
    $\mathcal{U}_A = \ell\text{-}\operatorname{argmax}_{\langle \mathbf{x}^{(i)}, s^{(i)} \rangle \in \mathcal{U}} \mathcal{H}(y|\mathbf{x}^{(i)}, \mathcal{L})$
    $min = \infty$
    for $\langle \mathbf{x}^{(i)}, s^{(i)} \rangle \in \mathcal{U}_A$ do
        $E\left[\mathcal{F}_t^i\right] = \mathbf{ExpF}\left(\langle \mathbf{x}^{(i)}, s^{(i)} \rangle, \mathcal{L}, C_{t-1}\right)$
        if $E\left[\mathcal{F}_t^i\right] < min$ then
            $min = E\left[\mathcal{F}_t^i\right]$
            $\langle \mathbf{x}^*, s^* \rangle = \langle \mathbf{x}^{(i)}, s^{(i)} \rangle$
        end if
    end for
    $y^*$ = label $\mathbf{x}^*$ using the labeling oracle
    move $\langle \mathbf{x}^*, s^*, y^* \rangle$ to $\mathcal{L}$
    train the classifier $C_t$ using $\mathcal{L}$
end for
return $C_t$
```

optimization criteria to fairness in the second level, SSU selects a point from $\mathcal{U}_A$ that maximizes the expected unfairness improvement (Eq. (5)) and pass it to the labeling oracle.

$$\underset{\langle \mathbf{x}^{(i)}, s^{(i)} \rangle \in \mathcal{U}_A}{\operatorname{argmax}} \left(\mathcal{F}(C_{t-1}) - E\left[\mathcal{F}_t^i\right]\right) = \underset{\langle \mathbf{x}^{(i)}, s^{(i)} \rangle \in \mathcal{U}_A}{\operatorname{argmin}} E\left[\mathcal{F}_t^i\right] \tag{5}$$

Besides assuring the accurate estimations of expected unfairness reduction, the nested optimization helps to reduce the time-complexity of FAL. This is especially important as it reduces the number of points to be evaluated for fairness from $|\mathcal{U}|$ to $\ell$. Note that, in order to estimate the unfairness reduction for every sample (Fig. 2), Algorithm 2 requires retraining of $k$ models. As a result, the computation time to run the framework is dominated by the total time to compute fairness values for the points in the unlabeled pool, before selecting one to be labeled. Hence, reducing the number of points to be evaluated for fairness significantly reduces the computation cost by a factor of $\frac{\ell}{|\mathcal{U}|}$.

The pseudo-code of FAL Nested is provided in Algorithm 4.

Before concluding this section, we would like to discuss the impact of small approximation errors on the performance of FAL-Nested. FAL-Nested considers highly uncertain points in the second stage. But there still is an approximation, since instead of selecting the point that maximizes the entropy (the best point from the accuracy perspective), it selects a point with the entropy close to maximum. Let the difference between the maximum entropy and the entropy of the selected point be $\epsilon$. Indeed, a small $\epsilon$ difference does not significantly impact the trained model. However, this small error will get propagated to the subsequent iterations, resulting in a drop in the accuracy, observed in our experiments in Section 8. Note that at every iteration, we use the model $C_{t-1}$ built in the previous iterations to estimate (i) entropy values and (ii) expected fairness improvement. A slightly less accurate intermediate model causes less accurate estimation of these values, hence making larger errors in selection of next iteration samples. As a result, the impact of error gets propagated to the subsequent iterations. Still, as we shall demonstrate in our experiments, the aggregated impact of error propagation was minimal, maintaining a high accuracy, while resolving the unfairness issues.

### 5.3. FAL nested-append

Since the sample points in $\mathcal{U}$ are unlabeled, FAL has no choice but to estimate the expected unfairness reduction after labeling a point, according to how likely it will take each possible label. But whether unfairness reduces depends on the actual label after adding the point to $\mathcal{L}$. To further clarify this, let us consider a toy example highlighted in Fig. 4. To simplify the explanation, we use 6 triangle and 6 circle points, each representing a demographic group, to evaluate the fairness of a linear binary classifier. Suppose the decision boundary of current classifier ($C_{t-1}$) is the one shown in solid line in the figure, and the

[3] Instead of a fixed set cardinality, one could consider selecting the points that their values are close to the maximum (e.g. have distance less than 0.001).

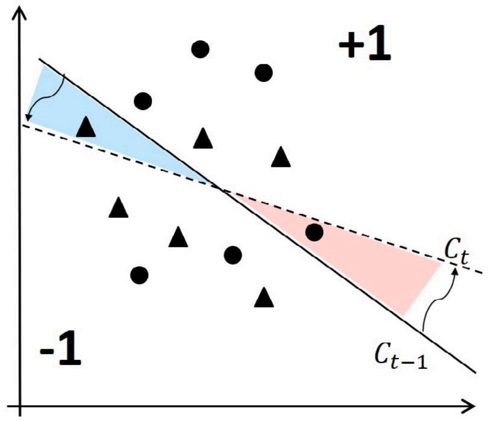


**Fig. 4.** Motivation example for Nested-Append approach.

**Algorithm 5** FAL Nested-Append

```
for $t = 1$ to $B$ do
    $\mathcal{U}_A = \ell\text{-argmax}_{\langle \mathbf{x}^{(i)}, s^{(i)}\rangle \in \mathcal{U}} \mathcal{H}(y|\mathbf{x}^{(i)}, \mathcal{L})$
    $min = \infty$
    for $\langle \mathbf{x}^{(i)}, s^{(i)}\rangle \in \mathcal{U}_A$ do
        $E\left[\mathcal{F}_t^i\right] = \textbf{ExpF}\left(\langle \mathbf{x}^{(i)}, s^{(i)}\rangle, \mathcal{L}, C_{t-1}\right)$
        if $E\left[\mathcal{F}_t^i\right] < min$ then
            $min = E\left[\mathcal{F}_t^i\right]$
            $\langle \mathbf{x}^*, s^*\rangle = \langle \mathbf{x}^{(i)}, s^{(i)}\rangle$
        end if
    end for
    $y^*$ = label $\mathbf{x}^*$ using the labeling oracle
    move $\langle \mathbf{x}^*, s^*, y^*\rangle$ to $\mathcal{L}$
    train the classifier $C_t$ using $\mathcal{L}$
    compute $\mathcal{F}(C_t)$
    if $\mathcal{F}(C_t) - \mathcal{F}(C_{t-1}) > 0$ then
        Replicate $\langle \mathbf{x}^*, s^*, y^*\rangle$ in $\mathcal{L}$
    end if
end for
return $C_t$
```

points in the top-right of the line are classified as +1. $C_{t-1}$ classifies two third (4 out of 6) of circles, but only one third (2 out of 6) of triangles as +1, hence is not fair according to DP.

Looking at the figure, to make the classifier fair, FAL needs to rotate the border toward the dashed line. Consider the two angles highlighted in the figure in the intersection of the two lines. To make the rotation, FAL needs to find points in $\mathcal{U}$ with the true +1 label that belong to the top-left angle, or the ones with the true −1 label in the bottom-right angle. Note that such points are misclassified by the current classifier. As a result, it is less likely to find the points with proper labels needed for reducing unfairness. On the other hand, if the label is not as hoped, adding the new point will not help to reduce unfairness. Therefore, to boost FAL for improving fairness, our next optimization, FAL Nested-Append, replicates the points that get labeled in a way that unfairness gets reduced. Let $\tilde{\mathbf{p}}_i = \langle \tilde{\mathbf{x}}^{(i)}, \tilde{s}^{(i)}\rangle \in \mathcal{U}_A$ be the point selected using FAL Nested. Let $\mathcal{F}(C_t)$ be the unfairness of the model after collecting the true label of $\tilde{\mathbf{p}}_i$ and adding it to $\mathcal{L}$. If $\mathcal{F}(C_t) - \mathcal{F}(C_{t-1}) > 0$, the algorithm replicates $\tilde{\mathbf{p}}_i$ in $\mathcal{L}$, further boosting its impact for unfairness reduction. In particular, since FAL Nested puts more emphasize on accuracy, FAL Nested-Append helps to account more for fairness. As we shall show in Section 8, boosting the performance of FAL Nested for fairness, FAL Nested-Append on average had a better performance across different experiments.

The pseudo-code of FAL Nested is provided in Algorithm 5.

## 6. Efficient FAL by covariance

Our proposed FAL framework is agnostic to the choice of classifier $C$. In this section, we show that it is possible to design efficient algorithm for the special case of generalized linear models. An appealing property of our algorithm is that, using the efficient computation provided in Section 6.1, it has *same* asymptotic time complexity as traditional active learning. We achieve this by avoiding the model retraining step for calculating the expected fairness of unlabeled samples, in Algorithm 2.

Consider a generalized linear model in form of $\hat{y} = \theta^\top \mathbf{x}$.[4] The covariance between the model and a sensitive attribute $s$ should be zero under model independence (demographic parity). We make a key observation in Lemma 1 that shows this covariance, $cov(s, \hat{y})$, *only depends on* $cov(s, \mathbf{x})$ *and* $\theta$.

**Lemma 1.** *For a generalized linear model in form of* $\hat{y} = \theta^\top \mathbf{x}$, $cov(S, \hat{y}) = \theta^\top cov(s, \mathbf{x})$.

**Proof.**

$$\begin{aligned}
cov(s,\hat{y}) &= E[s\,\hat{y}] - E[s]E[\hat{y}]\\
E[s]E[\hat{y}] &= \mu_s E\left[\sum \theta_i x_i\right] = \mu_s \sum \theta_i \mu_{x_i}\\
&= \theta_1 \mu_s \mu_{x_1} + \theta_2 \mu_s \mu_{x_2} + \cdots + \theta_d \mu_s \mu_{x_d}\\
E[s\hat{y}] &= E\left[s \sum \theta_i x_i\right] = E\left[\sum s\,\theta_i x_i\right]\\
&= E\left[s\,\theta_1 x_1 + s\,\theta_2 x_2 + \cdots + s\,\theta_d x_d\right]\\
&= E\left[s\,\theta_1 x_1\right] + E\left[s\,\theta_2 x_2\right] + \cdots + E\left[s\,\theta_d x_d\right]\\
&= \theta_1 E\left[s\,x_1\right] + \cdots + \theta_d E\left[s\,x_d\right]
\end{aligned}$$

$$\begin{aligned}
\Rightarrow cov(s,\hat{y}) &= \theta_1 E\left[s\,x_1\right] + \cdots + \theta_d E\left[s\,x_d\right] -\\
&\quad (\theta_1 \mu_s \mu_{x_1} + \cdots + \theta_d \mu_s \mu_{x_d})\\
&= \theta_1 (E\left[s\,x_1\right] - \mu_s \mu_{x_1}) + \cdots + \theta_d (E\left[s\,x_d\right] - \mu_s \mu_{x_d})\\
&= \sum_{i=1}^{d} \theta_i cov(s, x_i) = \theta^\top cov(s, \mathbf{x}) \quad \square
\end{aligned}$$

According to Lemma 1, the covariance of the model with the sensitive attribute (that results in unfairness) depends only on the weight vector $\theta$ and the underlying covariance of features $\mathbf{x}$ with $s$. We can reduce the model unfairness by ensuring that the model does not assign high weights to the problematic features (the features with high covariance with $s$). This observation allows us to indirectly optimize for fairness through covariance instead of computing expected unfairness reduction.

Consider a feature $x_i$ that is highly correlated with the sensitive attribute (i.e., $cov(x_i, s)$ is high) and also has a high weight $\theta_i$ in the current model. Our objective is to reduce the weight assigned to such features. The reason the model has assigned a large weight to $x_i$ is that *$x_i$ is highly predictive of $y$ in $\mathcal{L}$*. Therefore, in order to reduce the weight $\theta_i$, we need to reduce $cov_{\mathcal{L}}(x_i, y)$ in the labeled pool $\mathcal{L}$ to make it less predictive of $y$ in $\mathcal{L}$. Now, consider a point $P_j = \langle \mathbf{x}^{(j)}, s^{(j)}\rangle \in \mathcal{U}$ and its value $x_i^{(j)}$ on feature $\mathbf{x}_i$. Depending on $x_i^{(j)}$ and its label $y^{(j)}$ (after labeling), the point $P_j$ can impact $cov(x_i, y)$ in $\mathcal{L}$. Indeed, we do not know $y^{(j)}$ during the sample selection step. Still, similar to Section 5, we can consider the probability distribution over $y$ and calculate the expected improvement in covariance. Let $cov_i = cov_{\mathcal{L}}(x_i, y)$ be the covariance of $x_i$ and $y$ in $\mathcal{L}$ and $cov_{j,i,k} = cov_{\mathcal{L}\cup\{\langle \mathbf{x}^{(j)}, s^{(j)}, k\rangle\}}(x_i, y)$ the covariance of $x_i$ and $y$ after adding $\langle \mathbf{x}^{(j)}, s^{(j)}, k\rangle$ to $\mathcal{L}$. The expected covariance improvement for $x_i$ after adding $P_j$ to $\mathcal{L}$ is

$$E\left[cov_{j,i}^{\downarrow}\right] = \sum_{k=0}^{K-1} (|cov_i| - |cov_{j,i,k}|)\mathbb{P}(y = k|\mathbf{x}^{(j)}) \tag{6}$$

[4] The decision boundary of the classifier is viewed as a threshold value on $\hat{y}$ that separate different classes, using, for example, a sign function.

Following Lemma 1 the contribution of the covariance reduction for a feature $x_i$ to fairness is proportional to $|\theta_i cov(x_i, s)|$. Subsequently, it is important to reduce $cov_{\mathcal{L}}(x_i, y)$ for the features that are highly correlated with the sensitive attribute and have a high weight $\theta_i$ in the model. Therefore, the (indirect) fairness improvement by covariance for a point $P_j \in \mathcal{U}$ can be computed as following:

$$E\left[FbC_j^{\downarrow}\right] = \sum_{i=1}^{d} |\theta_i cov(s, x_i)| E\left[cov_{j,i}^{\downarrow}\right] \tag{7}$$

Now, it is enough to replace the term for expected unfairness reduction $\left(\mathcal{F}(C_{t-1}) - \mathcal{F}(C_t^i)\right)$, in Eqs. (3) and (5), with $E\left[FbC_i^{\downarrow}\right]$.

### 6.1. Efficiently computing covariance of $X$ and $y$ in $\mathcal{L}$

So far in this section, we proposed the efficient FAL by covariance method that works based on Eq. (7). Let $T$ be the time to train a classifier on $\mathcal{L}$. The time complexity of FAL (without considering the labeling cost) is $O(B \cdot T \cdot |\mathcal{U}|)$. That simply is because at every iteration FAL requires to train a constant number of classifiers per each sample in $\mathcal{U}$, while there totally are $B$ iterations. In this section we show that maintaining the aggregates from the previous steps, we can further improve the efficiency by computing $E\left[FbC_j^{\downarrow}\right]$ in *constant time*, when $d$ is constant. As a result the time complexity drops to $O\left(B \cdot (|\mathcal{U}| + T)\right)$, *the same as traditional active learning*, when FAL by covariance method is used.

First, we note that $cov(x_i, s)$, the covariance of each feature $x_i$ with $S$, does not depend on $\mathcal{L}$ and can be computed in advance, using the unlabeled samples in $\mathcal{U}$. It is computed once for every feature at the beginning of the process and the same numbers will be used in different iterations. The values of $cov_i = cov_{\mathcal{L}}(x_i, y)$ and $cov_{j,i,k} = cov_{\mathcal{L} \cup \{\langle \mathbf{x}^{(j)}, s^{(j)}, y^{(j)} = k \rangle\}}(x_i, y)$ in Eq. (6), however, depend on the set of labeled data and should get recomputed at different iterations and for different points $P_j \in \mathcal{U}$. We maintain the following aggregates for efficiently computing these values:

$$\mathcal{G}_y = \sum_{\forall \langle \mathbf{x}^{(\ell)}, s^{(\ell)}, y^{(\ell)} \rangle \in \mathcal{L}} y^{(\ell)}$$

$$\mathcal{G}_x[i] = \sum_{\forall \langle \mathbf{x}^{(\ell)}, s^{(\ell)}, y^{(\ell)} \rangle \in \mathcal{L}} x_i^{(\ell)}$$

$$\mathcal{G}_z[i] = \sum_{\forall \langle \mathbf{x}^{(\ell)}, s^{(\ell)}, y^{(\ell)} \rangle \in \mathcal{L}} x_i^{(\ell)} y^{(\ell)}$$

Note that at every iteration each of the above aggregates can be updated in constant time by adding the corresponding value from the new point to it. Now, using these aggregates:

$$cov_i = cov_{\mathcal{L}}(x_i, y) = \frac{\mathcal{G}_z[i]}{n} - \frac{\mathcal{G}_x[i]}{n} \times \frac{\mathcal{G}_y}{n}$$

Similarly, for a point $P_j = \langle \mathbf{x}^{(j)}, s^{(j)} \rangle \in \mathcal{U}$ and a label $y^{(j)} = k$:

$$cov_{j,i,k} = \frac{\mathcal{G}_z[i] + k x_i^{(j)}}{n+1} - \frac{\mathcal{G}_x[i] + x_i^{(j)}}{n+1} \times \frac{\mathcal{G}_y + k}{n+1}$$

## 7. Extension to other fairness models

So far in this paper, we considered independence ($\hat{y} \perp s$) for fairness. Next, we discuss how to extended our findings to other measures based on separation and sufficiency (Barocas et al., 2019), such as equalized odds (Hardt et al., 2016), where the prediction outcome $\hat{y}$ is independent of the sensitive feature $s$ given the true label $y$, i.e. $P(\hat{y} = 1 | y = 1, s = 0) = P(\hat{y} = 1 | y = 1, s = 1), y \in 0, 1$.

The fairness–accuracy optimizer of FAL is not limited to a specific notion of fairness for balancing accuracy and fairness. Similarly, the notion of expected unfairness reduction does not rely on a specific notion of fairness as $\mathcal{F}(.)$ in Eq. (2) can be computed using any fairness measure, besides demographic parity. As a result, at a high-level, the FAL framework should work as-is for other notions of fairness as well. However, as we shall explain in the following, computing $\mathcal{F}(.)$ would require additional information that comes at a cost of randomly labeling a subset of data.

Looking at Fig. 2, recall that we use the initial unlabeled set $I\mathcal{U}$ for estimating the fairness of a model. $I\mathcal{U}$ follows the underlying data distribution and, hence, can be used for evaluating the demographic disparity. However, this set cannot be used for estimating fairness according to separation or sufficiency since its instances are not labeled. On the other hand, the pool of labeled data is not representative of the underlying data distribution.

Our resolution is to use a small subset of $C_{\mathcal{U}} \subset I\mathcal{U}$ for fairness computation, accepting the potential error in estimations relative to the size of the set. Before starting the FAL process, we need to label $C_{\mathcal{U}}$. Once labeled, $C_{\mathcal{U}}$ will be used for calculating $\mathcal{F}(.)$ based on other notions of fairness, and FAL can be executed as-is. In Section 8, we run experiments to show the extension of FAL for equalized odds.

## 8. Experiments

The experiments were performed on a Linux machine with a Core I9 CPU and 128 GB memory. The algorithms were implemented using Python 3.7.[5]

### 8.1. Datasets

*COMPAS*[6]: published by ProPublica (Angwin et al., 2016), this dataset contains information of juvenile felonies such as `marriage status`, `race`, `age`, `prior convictions`, and the `charge degree` of the current arrest. We normalized data so that it has zero mean and unit variance. We consider `race` as sensitive attribute and filtered dataset to black and white defendants. The dataset contains 5,875 records, after filtering. Following the standard practice (Corbett-Davies et al., 2017; Mehrabi et al., 2019), we use two-year violent recidivism record as the true label of recidivism: $y^{(i)} = 1$ if the recidivism is greater than zero and $y^{(i)} = 0$ otherwise.

*Adult dataset*[7]: contains 45,222 individuals income extracted from the 1994 census data with attributes such as `age`, `occupation`, `education`, `race`, `sex`, `marital-status`, `hours-per-week`, `native country`, etc. We use `income` (a binary attribute with values $\geq 50k\$$ and $\leq 50k\$$) as the true label. We consider `sex` as the sensitive attribute. We normalized data so that it has zero mean and unit variance.

### 8.2. Algorithms evaluated

All our proposed approaches are evaluated using a regularized $\ell_2$ norm logistic regression classifier with a regularization strength of one. We trained the logistic regression with *liblinear* optimizer and with a maximum iteration of 100. Our findings are transferable to other classifiers. We begin by comparing our proposed approach against a wide variety of representative baselines. Then, we focus on understanding the effectiveness and performances of our proposed approaches under different settings.

**Baselines:** We consider four baselines in order to build a fair classifier in a limited data environment. We first start to evaluate passive methods, **RandL** and **R-FLR**, that select all the samples randomly at one shot to form a training set and fit a regular and fair logistic regression (proposed by (Zafar et al., 2017)), respectively. We then evaluate active methods, **AL** and **AL-FLR**, which iteratively select a sample point based on its informativeness (Eq. (1)) and fits a regular and fair logistic regression (Zafar et al., 2017) in each iteration, respectively.

[5] Our codes are publicly available: https://github.com/anahideh/FAL--Fair-Active-Learning

[6] ProPublica, https://bit.ly/35pzGFj

[7] UCI repository, https://bit.ly/2GTWz9Z

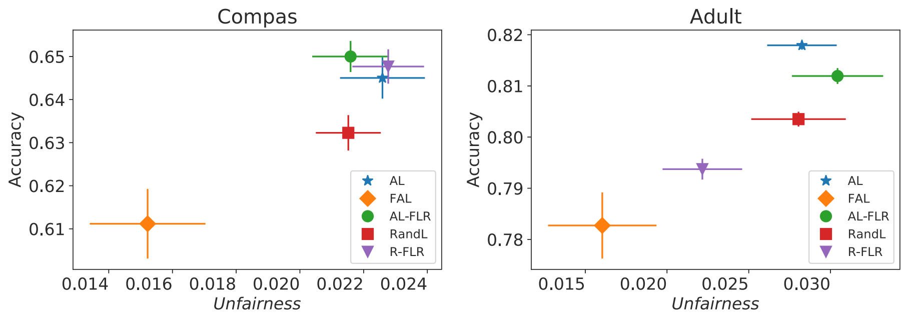
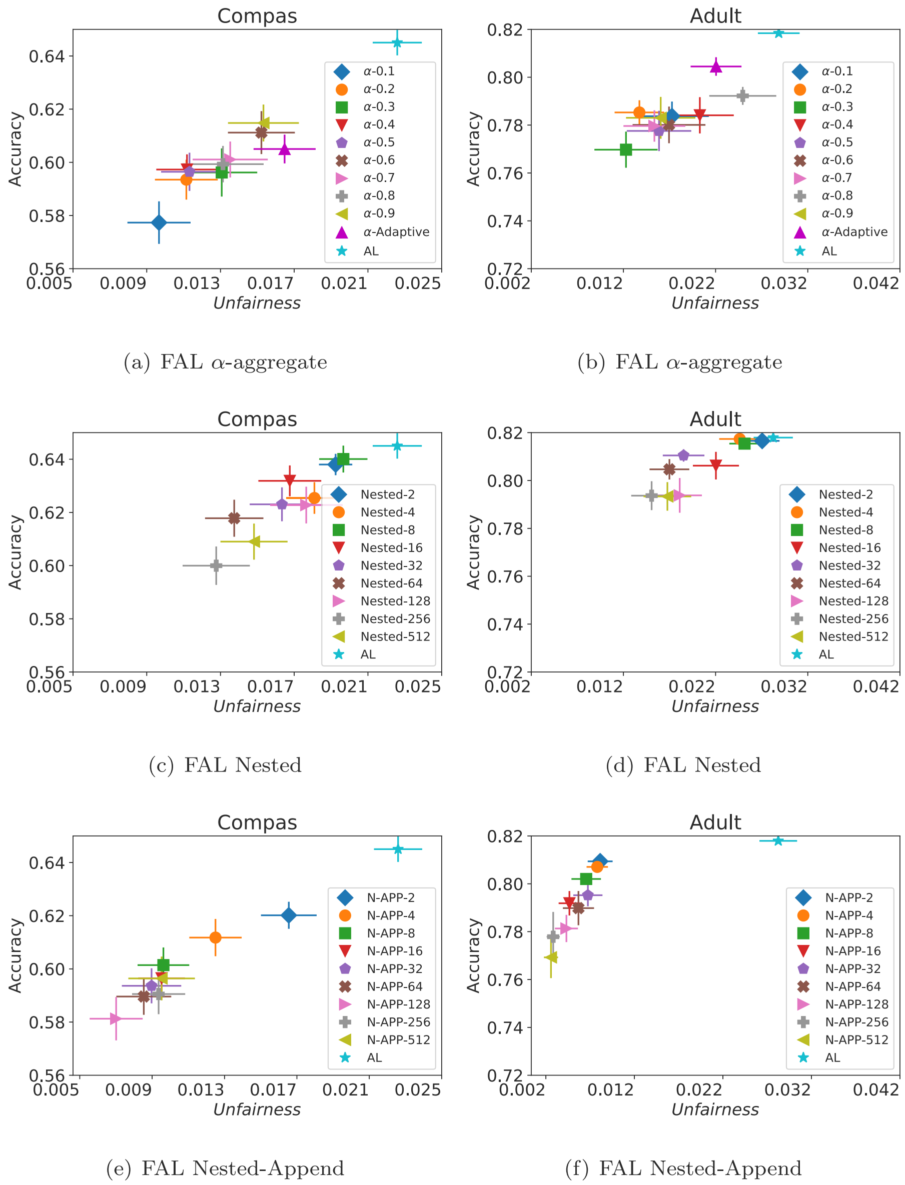


**Fig. 5.** Comparison with baselines.



(a) FAL $\alpha$-aggregate

(b) FAL $\alpha$-aggregate

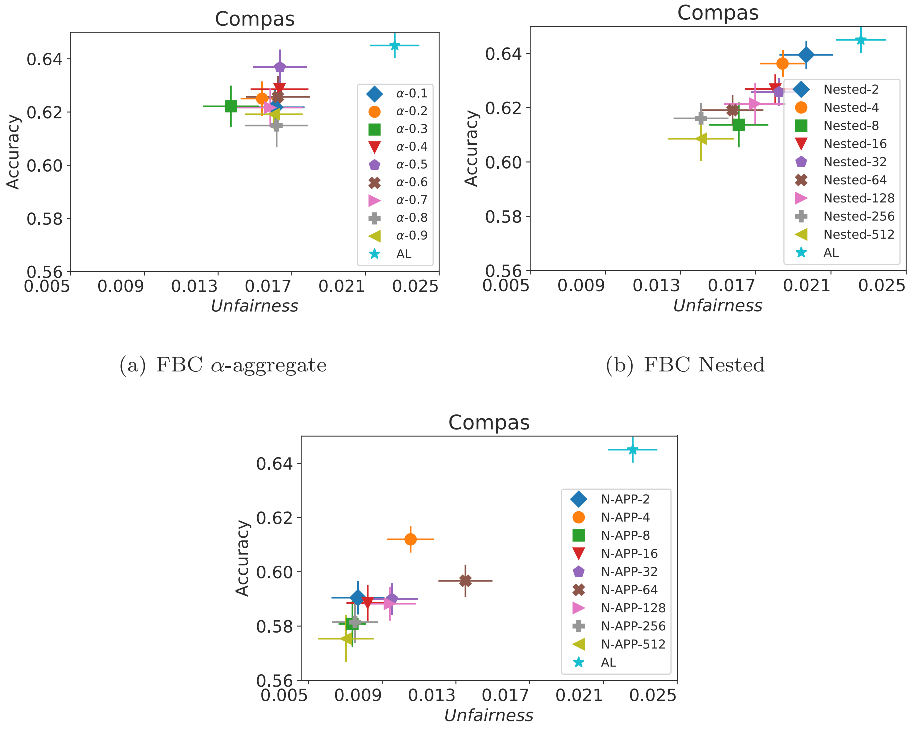
(c) FAL Nested

(d) FAL Nested

(e) FAL Nested-Append

(f) FAL Nested-Append

**Fig. 6.** The average DP and accuracy of different FAL approaches.



(a) FBC $\alpha$-aggregate

(b) FBC Nested

(c) FBC Nested-Append

**Fig. 7.** The average DP and accuracy of different FBC approaches, COMPAS Dataset.

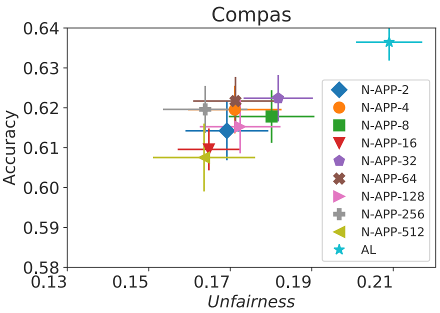


**Fig. 8.** FAL with Equalized odds, COMPAS Dataset.

**Our Algorithms:** We evaluate our fairness–accuracy optimization algorithms proposed in Section 5, namely FAL $\alpha$-aggregate ( FAL-$\alpha$), FAL Nested (**Nested**), and FAL Nested-Append (**N-App**). For FAL-$\alpha$, we normalize the accuracy and fairness improvement values as ($v$-min)/(max–min) before combining them in Eq. (3). Besides fixed values of $\alpha$, we also consider an adaptive $\alpha$ parameter, using a decay function that, starting from $\alpha = 1$ to $\alpha = 0$, the value $\alpha$ drops by 0.1 every $\lfloor B/11 \rfloor$ iterations. For **Nested**, and **N-App**, we consider different exponents of 2 as the value of $\ell$ (in Eq. (4)) from 2 to 512.

Our default choice for computing unfairness reduction is Algorithm 2. The efficient FAL by covariance (**FBC**), proposed in Section 6, is also evaluated to show the computation time improvement. We evaluate **FBC** with the three optimization approaches FAL-$\alpha$, **Nested**, and **N-App**. Finally, in order to show the extension of our proposal for other fairness models, we run FAL using Equalized Odds as the fairness measure.

### 8.3. Performance evaluation

We perform the experiments using 30 random splits of the datasets into training $\mathcal{U}$ (60% of the examples) and testing (40% of the examples). We consider the mean and variance over the 30 random splits. We specify the maximum labeling budget to 200 after which performance leveled off. In each FAL and AL scenarios, we start with six labeled points and sequentially select points to label, until the budget is exhausted. Mutual information is our default measure of demographic parity.

We first evaluate the performance of FAL-$\alpha$ versus the passive and active baselines **RandL**, **R-FLR**, **AL**, and **AL-FLR**, using accuracy and fairness measures to show the deficiency of these approaches in construction of a final fair classifier. Fig. 5 illustrates the performance of baselines and FAL-$\alpha$ where $\alpha = 0.6$ for COMPAS and Adult datasets. The bars indicate the standard deviation on 30 random split of data. We observe that the baselines had similar performances on fairness. Even applying a fair classifier (**FLR**) fails to improve the fairness. FAL-$\alpha$, on the other hand, significantly reduces unfairness while sacrificing small amount of accuracy. The results indicate a significant drop in the unfairness of the model (almost 30% reduction) versus the accuracy drop of 4% when we use FAL.

One can observe that **AL-FLR** has higher unfairness compared to the proposed **FAL**. To justify let us first review how **AL-FLR** algorithm works. At every iteration of the active learning process, **AL-FLR** trains a FairML model on the collected samples instead of a regular ML model. Given the known trade-off between fairness and model accuracy in a FairML model, the intermediate models then are less accurate (than if regular ML was used instead). The intermediate models are used to select the next sample points to be labeled. However, due to the low accuracy of an intermediate model in **AL-FLR**, it poorly estimates the entropy values that introduces error in estimating expected unfairness values, resulting in selecting sub-optimal points for being labeled next. The erroneous estimation of values, causing a poor selection of points for sampling that is propagated to the subsequent iterations, further

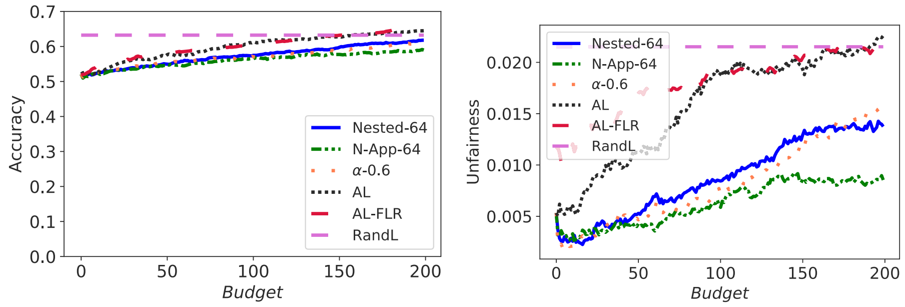


**Fig. 9.** Performance evaluation over budget, COMPAS Dataset.

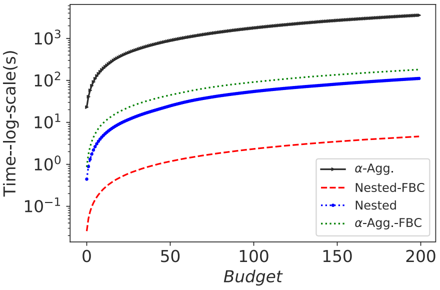


**Fig. 10.** Computation time of proposed approaches, COMPAS Dataset.

**Table 2**
Average replication percentage over 30 random runs of N-App.

| $l$ | Adult-Eodds | Compas-Eodds | Adult-Dparity | Compas-Dparity |
|---|---|---|---|---|
| 2 | 0.31 | 0.35 | 0.30 | 0.33 |
| 4 | 0.31 | 0.35 | 0.31 | 0.35 |
| 8 | 0.33 | 0.38 | 0.31 | 0.37 |
| 16 | 0.33 | 0.36 | 0.31 | 0.36 |
| 32 | 0.34 | 0.36 | 0.31 | 0.36 |
| 64 | 0.34 | 0.38 | 0.32 | 0.37 |
| 128 | 0.36 | 0.38 | 0.32 | 0.37 |
| 256 | 0.37 | 0.37 | 0.33 | 0.37 |
| 512 | 0.36 | 0.37 | 0.34 | 0.38 |

escalating its negative impact. In summary, the poor performance of the intermediate models built by **AL-FLR** introduces errors that their impact gets propagated in all subsequent iterations, resulting in models with both low accuracy and fairness.

In our next experiment illustrated in Fig. 6, we evaluate the average performance of three different optimizers  FAL-$\alpha$, Nested, and **N-App**, the efficient algorithms proposed in Section 5 and compare it against AL. Fig. 6(a) presents a comprehensive comparison of FAL with different user-defined $\alpha$ and adaptive alpha, versus AL on COMPAS dataset. We can observe that FAL-$\alpha$ achieved a good level of fairness across different $\alpha$ values. Fig. 6(c) corresponds to the performance of **Nested**, which focuses on the upper percentile of the entropy distribution, to ensure that the selected points are improving accuracy, and not only the unfairness. As expected, the results indicate that this approach nudges up the accuracy of FAL-$\alpha$. Finally, in Fig. 6(e), we evaluate the average performance of **N-App** on COMPAS dataset. Compared to both FAL-$\alpha$ and **Nested**, the unfairness level of the model dramatically improved by appending the points two times when they truly improve the unfairness level in each iteration. Similarly, we replicated the results for Adult dataset as in Fig. 6(b), 6(d), and 6(f). It can be seen that the effective **N-App** approach significantly improves the unfairness level of the model while maintaining its accuracy.

We also evaluate the average performance of **FBC** approach as proposed in Section 6. Fig. 7 includes the results of **FBC** with the three proposed optimizers on COMPAS dataset. The results are fairly consistent with the results we observed in Fig. 6 for COMPAS dataset. Note that **FBC** is an approximation of the expected unfairness reduction and is computationally more efficient to be used in FAL algorithm (Fig. 10).

As we discussed in Section 7, our proposed approaches can be extended to use other fairness measures. Fig. 8 corresponds to the experimental setup where he Equalized Odds notion of fairness as proposed by Hardt et al. (2016) is used in FAL-$\alpha$ for measuring fairness. The results indicate the effectiveness of FAL compared to AL using different $\alpha$ values. The results of the same setting for

Fig. 9 corresponds to the average (un)fairness and average accuracy score on 30 random runs using COMPAS dataset. Looking at the figure, we can observe that **Nested-64** enforces the accuracy while considering the fairness in the sample selection. Hence, with a higher accuracy and lower unfairness it outperforms FAL-$\alpha$-0.6. Overall, **N-App-64**, outperforms both FAL-$\alpha$-0.6 and **Nested-64**, significantly reducing unfairness while dropping the accuracy by a small amount. Note that since Nested-Append replicates some of the sample points (but not others) it pays a price in accuracy reduction. However, our experiments demonstrate in practice the accuracy drop is negligible. In particular, looking at Fig. 9, we can observe that Nested-append has lower accuracy than Nested. This indicates the impact of replicating points, significantly reducing the unfairness. Although N-App-64 sacrifices accuracy to *cut half of the unfairness of Nested-64*, the accuracy is minimally dropped as it achieved almost $.58/.6 = 96\%$ of the accuracy of Nested-64.

Table 2 demonstrates the average percentage of observations that are replicated in the Nested Append approach. Overall, 30% of the observations are replicated, that confirms the effectiveness of the replication on the final model performance compared to the Nested approach, as shown in Fig. 9.

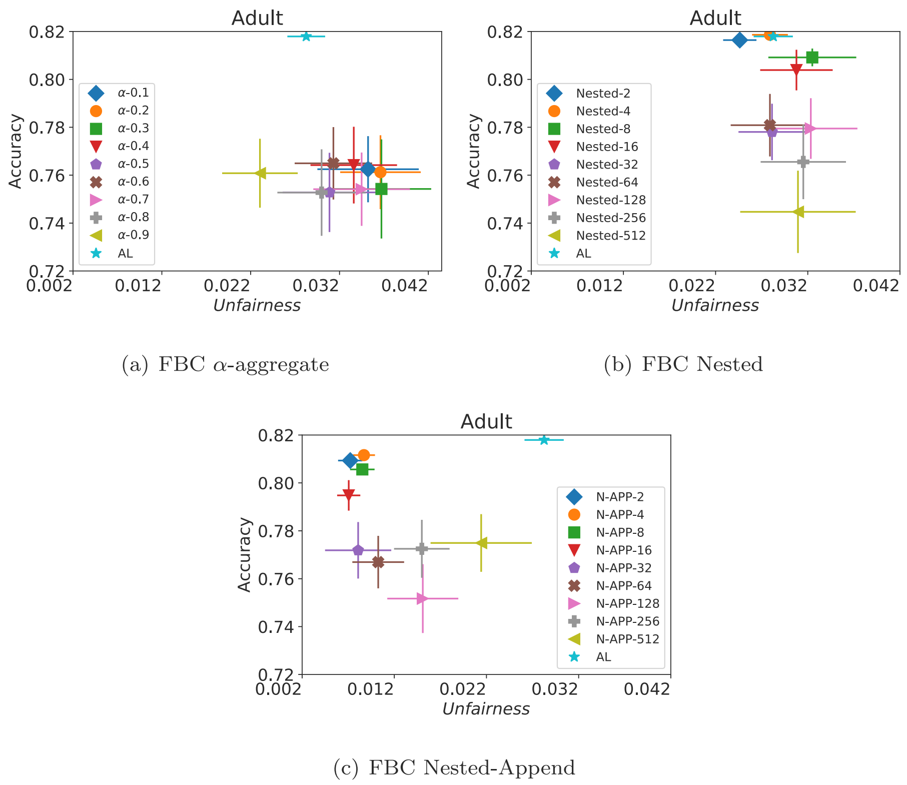
Fig. 10 shows the computation time of each sampling iteration for different accuracy–fairness optimizers compared to the original FAL on COMPAS dataset. FBC is orders of magnitude faster than FAL as it avoids the need to compute expected fairness. On the other hand, since it indirectly optimizes for fairness, FAL outperforms it on fairness.

Next, we provide additional experimental results on the performance of our proposed algorithms, using FBC and Equalized Odds.

Fig. 11 presents results for **FBC** approach using three different optimizers, FAL-$\alpha$, **Nested**, and **N-App** on Adult dataset. It can be seen that our ideal **N-App** approach outperforms other optimizers when we use the efficient FBC approach for fairness approximation.

In Fig. 8 we provided our experiment results for Equalized Odds, using **N-App**, on COMPAS dataset. Fig. 12 shows our complimentary results for the other two accuracy–fairness optimizers: FAL-$\alpha$ and **Nested**.

Finally, Fig. 13 provides results of FAL using three different optimizer for Equalized Odds on Adult dataset. The results indicate that our efficient and effective **N-App** approach outperforms other optimizers in terms of unfairness reduction while maintaining accuracy.

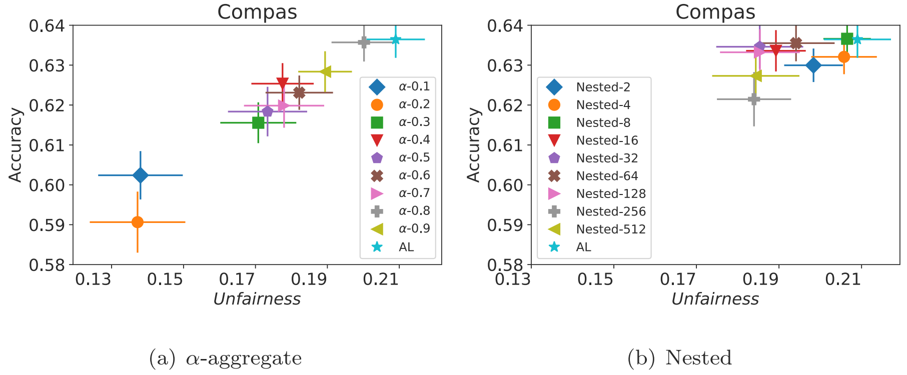


(a) FBC α-aggregate

(b) FBC Nested

(c) FBC Nested-Append

**Fig. 11.** The average DP and accuracy of different FBC approaches, Adult Dataset.



(a) α-aggregate

(b) Nested

**Fig. 12.** Equalized Odds, COMPAS Dataset.

## 9. Final remarks

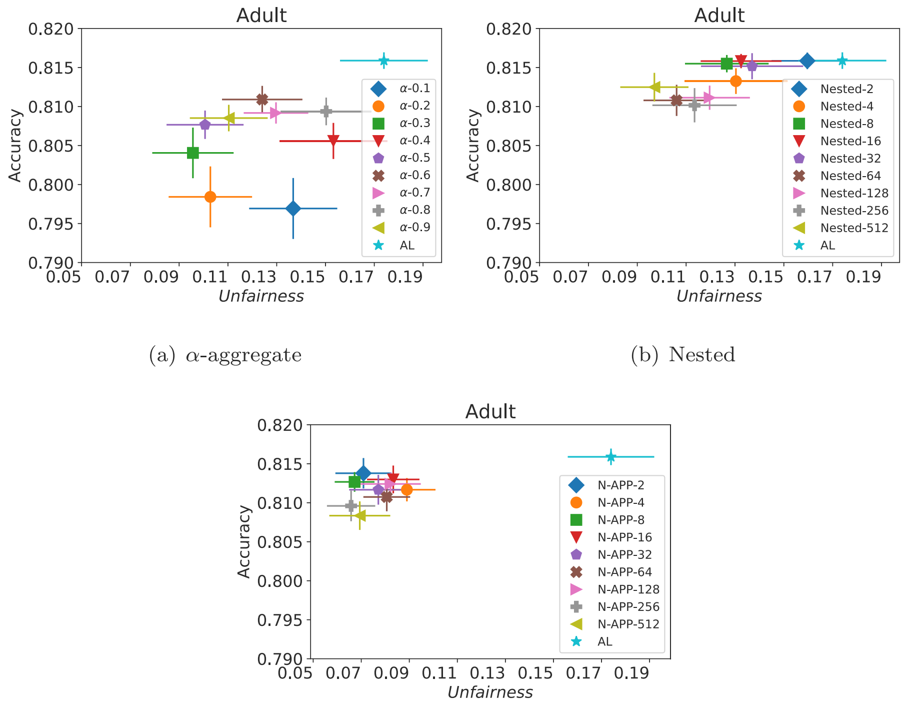
Prior works on fair classification assume the availability of sufficiently labeled data. In a number of societal applications such as recidivism prediction, the labeled data is unavailable and collecting it is expensive and time-consuming. The traditional active learning approach focuses on accuracy, often at the cost of fairness. In this paper, we proposed a framework for fairness in active learning that balances fairness and accuracy by selecting samples from the unlabeled pool that maximizes a linear combination of misclassification error reduction and improvement over expected fairness. We described a wide variety of optimizations for improving accuracy, fairness, and running time. Specifically, **FAL Nested-Append** successfully achieves a deft balance between accuracy and fairness. Our extensive experiments on real datasets confirm that our proposed approach produces a fairer model without significantly sacrificing the accuracy. We hope that our proposed approach will have a positive impact by improving 1the model fairness in a number of real-world scenarios.

## CRediT authorship contribution statement

**Hadis Anahideh:** Methodology development, Draft preparation, Implementation, and validation of our proposal. **Abolfazl Asudeh:** Methodology development, Draft preparation, Implementation, and validation of our proposal. **Saravanan Thirumuruganathan:** Methodology development, Draft preparation, Implementation, and validation of our proposal.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.



(a) $\alpha$-aggregate

(b) Nested

(c) Nested-Append

**Fig. 13.** Equalized Odds, Adult Dataset.

## References

Angluin, D. (1988). Queries and concept learning. *Machine Learning*, *2*(4), 319–342.

Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016). Machine bias: Risk assessments in criminal sentencing. *ProPublica*, URL https://bit.ly/2s0UMfA.

Asudeh, A., Jagadish, H., Stoyanovich, J., & Das, G. (2019). *Designing fair ranking schemes*. SIGMOD.

Asudeh, A., Jin, Z., & Jagadish, H. (2019). Assessing and remedying coverage for a given dataset. In *ICDE* (pp. 554–565). IEEE.

Asudeh, A., Shahbazi, N., Jin, Z., & Jagadish, H. (2021). Identifying insufficient data coverage for ordinal continuous-valued attributes. In *Proceedings of the 2021 international conference on management of data* (pp. 129–141).

Bakker, M. A., Valdés, H. R., Tu, D. P., Gummadi, K. P., Varshney, K. R., Weller, A., et al. (2020). Fair enough: Improving fairness in budget-constrained decision making using confidence thresholds. In *SafeAI@ AAAI*.

Balcan, M.-F., Broder, A., & Zhang, T. (2007). Margin based active learning. In *COLT* (pp. 35–50). Springer.

Barocas, S., Hardt, M., & Narayanan, A. (2017). Fairness in machine learning. *NIPS Tutorial*.

Barocas, S., Hardt, M., & Narayanan, A. (2019). Fairness and machine learning: Limitations and opportunities. https://fairmlbook.org.

Barocas, S., & Selbst, A. D. (2016). Big data's disparate impact. *California Law Review*, *104*, 671.

Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In *FAccT* (pp. 77–91).

Calders, T., & Verwer, S. (2010). Three naive Bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, *21*(2), 277–292.

Calmon, F., Wei, D., Vinzamuri, B., Ramamurthy, K. N., & Varshney, K. R. (2017). Optimized pre-processing for discrimination prevention. In *Advances in neural information processing systems* (pp. 3992–4001).

Celis, L. E., Huang, L., Keswani, V., & Vishnoi, N. K. (2019). Classification with fairness constraints: A meta-algorithm with provable guarantees. In *FAccT* (pp. 319–328). ACM.

Celis, L. E., Keswani, V., & Vishnoi, N. (2020). Data preprocessing to mitigate bias: A maximum entropy based approach. In *International conference on machine learning* (pp. 1349–1359). PMLR.

Cohn, D., Atlas, L., & Ladner, R. (1994). Improving generalization with active learning. *Machine Learning*, *15*(2), 201–221.

Cohn, D. A., Ghahramani, Z., & Jordan, M. I. (1996). Active learning with statistical models. *Journal of Artificial Intelligence Research*, *4*, 129–145.

Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., & Huq, A. (2017). Algorithmic decision making and the cost of fairness. In *SIGKDD* (pp. 797–806). ACM.

Dagan, I., & Engelson, S. P. (1995). Committee-based sampling for training probabilistic classifiers. In *Machine learning proceedings 1995* (pp. 150–157). Elsevier.

Dasgupta, S., Hsu, D. J., & Monteleoni, C. (2007). *A general agnostic active learning algorithm*. Citeseer.

Donmez, P., Carbonell, J. G., & Bennett, P. N. (2007). Dual strategy active learning. In *ECIR*.

Drosou, M., Jagadish, H., Pitoura, E., & Stoyanovich, J. (2017). Diversity in big data: A review. *Big Data*, *5*(2), 73–84.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2012). Fairness through awareness. In *ITCS* (pp. 214–226). ACM.

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., & Venkatasubramanian, S. (2015). Certifying and removing disparate impact. In *SIGKDD*.

Fish, B., Kun, J., & Lelkes, A. D. (2016). A confidence-based approach for balancing fairness and accuracy. In *SDM* (pp. 144–152). SIAM.

Freytag, A., Rodner, E., & Denzler, J. (2014). Selecting influential examples: Active learning with expected model output changes. In *European conference on computer vision* (pp. 562–577). Springer.

Gilad-Bachrach, R., Navot, A., & Tishby, N. (2006). Query by committee made real. In *Advances in neural information processing systems* (pp. 443–450).

Goh, G., Cotter, A., Gupta, M., & Friedlander, M. P. (2016). Satisfying real-world goals with dataset constraints. In *NeurIPS* (pp. 2415–2423).

Hardt, M., Price, E., Srebro, N., et al. (2016). Equality of opportunity in supervised learning. In *NeurIPS* (pp. 3315–3323).

Hébert-Johnson, U., Kim, M. P., Reingold, O., & Rothblum, G. N. (2017). Calibration for the (computationally-identifiable) masses. arXiv preprint arXiv:1711.08513.

Hoi, S. C., Jin, R., & Lyu, M. R. (2006). Large-scale text categorization by batch mode active learning. In *Proceedings of the 15th international conference on world wide web* (pp. 633–642).

Huang, S.-J., Jin, R., & Zhou, Z.-H. (2010). Active learning by querying informative and representative examples. In *NeurIPS* (pp. 892–900).

Huang, L., & Vishnoi, N. K. (2019). Stable and fair classification. arXiv preprint arXiv:1902.07823.

Jan, T. (2018). Redlining was banned 50 years ago. It's still hurting minorities today. Washington Post.

Jones, F. L. (1983). Sources of gender inequality in income: what the Australian census says. *Social Forces*, *62*(1), 134–152.

Kamiran, F., & Calders, T. (2012). Data preprocessing techniques for classification without discrimination. *KAIS*, *33*(1), 1–33.

Kearns, M., Neel, S., Roth, A., & Wu, Z. S. (2018). Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *ICML* (pp. 2564–2572).

Kim, M. P., Ghorbani, A., & Zou, J. (2018). Multiaccuracy: Black-box post-processing for fairness in classification. arXiv preprint arXiv:1805.12317.

King, R. D., Whelan, K. E., Jones, F. M., Reiser, P. G., Bryant, C. H., Muggleton, S. H., et al. (2004). Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, *427*(6971), 247–252.

Komiyama, J., Takeda, A., Honda, J., & Shimao, H. (2018). Nonconvex optimization for regression with fairness constraints. In *ICML*.

Krasanakis, E., Spyromitros-Xioufis, E., Papadopoulos, S., & Kompatsiaris, Y. (2018). Adaptive sensitive reweighting to mitigate bias in fairness-aware classification. In *WWW* (pp. 853–862).

Kumar, P., & Gupta, A. (2020). Active learning query strategies for classification, regression, and clustering: a survey. *Journal of Computer Science and Technology*, *35*(4), 913–945.

Kusner, M. J., Loftus, J., Russell, C., & Silva, R. (2017). Counterfactual fairness. In *NeurIPS* (pp. 4066–4076).

Lewis, D. D., & Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994* (pp. 148–156). Elsevier.

Lewis, D. D., & Gale, W. A. (1994). A sequential algorithm for training text classifiers. In *SIGIR'94* (pp. 3–12). Springer.

McCallumzy, A. K., & Nigamy, K. (1998). Employing EM and pool-based active learning for text classification. In *Proc. International conference on machine learning* ICML, (pp. 359–367). Citeseer.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2019). A survey on bias and fairness in machine learning. arXiv preprint arXiv:1908.09635.

Melville, P., & Mooney, R. J. (2004). Diverse ensembles for active learning. In *Proceedings of the twenty-first international conference on machine learning* (p. 74).

Menon, A. K., & Williamson, R. C. (2018). The cost of fairness in binary classification. In *FAccT* (pp. 107–118).

Mitchell, T. M. (1982). Generalization as search. *Artificial Intelligence*, *18*(2), 203–226.

Moskovitch, R., Nissim, N., Stopel, D., Feher, C., Englert, R., & Elovici, Y. (2007). Improving the detection of unknown computer worms activity using active learning. In *Annual conference on artificial intelligence* (pp. 489–493). Springer.

Narayanan, A. (2018). Translation tutorial: 21 fairness definitions and their politics. In *Proc. conf. fairness accountability transp., New York, USA*.

Noriega-Campero, A., Bakker, M. A., Garcia-Bulle, B., & Pentland, A. (2019). Active fairness in algorithmic decision making. In *AIES* (pp. 77–83).

Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., & Weinberger, K. Q. (2017). On fairness and calibration. In *NeurIPS* (pp. 5680–5689).

Roy, N., & McCallum, A. (2001). Toward optimal active learning through monte carlo estimation of error reduction. *ICML, Williamstown*, *2*, 441–448.

Salimi, B., Rodriguez, L., Howe, B., & Suciu, D. (2019). Interventional fairness: Causal database repair for algorithmic fairness. In *SIGMOD* (pp. 793–810).

Schaul, T., Zhang, S., & LeCun, Y. (2013). No more pesky learning rates. In *ICML* (pp. 343–351).

Settles, B. (2012). Active learning. In *Synthesis lectures on artificial intelligence and machine learning, vol. 18* (pp. 1–111).

Settles, B., & Craven, M. (2008). An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the 2008 conference on empirical methods in natural language processing* (pp. 1070–1079).

Settles, B., Craven, M., & Ray, S. (2007). Multiple-instance active learning. *Advances in Neural Information Processing Systems*, *20*, 1289–1296.

Seung, H. S., Opper, M., & Sompolinsky, H. (1992). Query by committee. In *Proceedings of the fifth annual workshop on computational learning theory* (pp. 287–294).

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, *27*(3), 379–423.

Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, *5*(1), 3–55.

Sharaf, A., & Daumé III, H. Promoting fairness in learned models by learning to active learn under parity constraints. In *ICML 2020 Workshop on real world experiment design and active learning*.

Simoiu, C., Corbett-Davies, S., Goel, S., et al. (2017). The problem of infra-marginality in outcome tests for discrimination. *The Annals of Applied Statistics*, *11*(3), 1193–1216.

Soen, A., Husain, H., & Nock, R. (2020). Data preprocessing to mitigate bias with boosted fair mollifiers. arXiv preprint arXiv:2012.00188.

Stoyanovich, J., Yang, K., & Jagadish, H. (2018). Online set selection with fairness and diversity constraints. In *EDBT*.

Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. *JMLR*, *2*(Nov), 45–66.

Tur, G., Hakkani-Tür, D., & Schapire, R. E. (2005). Combining active and semi-supervised learning for spoken language understanding. *Speech Communication*, *45*(2), 171–186.

Wu, X., Chen, C., Zhong, M., Wang, J., & Shi, J. (2021). COVID-AL: The diagnosis of COVID-19 with deep active learning. *Medical Image Analysis*, *68*, Article 101913.

Xu, Z., Yu, K., Tresp, V., Xu, X., & Wang, J. (2003). Representative sampling for text classification using support vector machines. In *European conference on information retrieval* (pp. 393–407). Springer.

Xu, D., Yuan, S., Zhang, L., & Wu, X. (2018). Fairgan: Fairness-aware generative adversarial networks. In *2018 IEEE international conference on big data (Big Data)* (pp. 570–575). IEEE.

Zafar, M. B., Valera, I., Gomez Rodriguez, M., & Gummadi, K. P. (2017). Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *WWW* (pp. 1171–1180).

Zafar, M. B., Valera, I., Rodriguez, M. G., & Gummadi, K. P. (2015). Fairness constraints: Mechanisms for fair classification. arXiv preprint arXiv:1507.05259.

Zehlike, M., Bonchi, F., Castillo, C., Hajian, S., Megahed, M., & Baeza-Yates, R. (2017). Fa* ir: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on conference on information and knowledge management* (pp. 1569–1578). ACM.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., & Dwork, C. (2013). Learning fair representations. In *ICML* (pp. 325–333).

Zhang, H., Chu, X., Asudeh, A., & Navathe, S. B. (2021). Omnifair: A declarative system for model-agnostic group fairness in machine learning. In *Proceedings of the 2021 international conference on management of data* (pp. 2076–2088).

Žliobaitė, I. (2017). Measuring discrimination in algorithmic decision making. *Data Mining and Knowledge Discovery*, *31*(4), 1060–1089.

Zou, J., & Schiebinger, L. (2018). *AI can be sexist and racist—it's time to make it fair*. Nature Publishing Group.

**Hadis Anahideh**

**Abolfazl Asudeh**

**Saravanan Thirumuruganathan**