Supposedly Fair Classification Systems and Their Impacts

Mackenzie Jorgensen[1], Elizabeth Black[1], Natalia Criado[2] and Jose Such[1,2]

[1]King's College London, London, UK
[2]Universitat Politècnica de València, Valencia, Spain

Abstract
The algorithmic fairness field has boomed with discrimination mitigation methods to make Machine Learning (ML) model predictions fairer across individuals and groups. However, recent research shows that these measures can sometimes lead to harming the very people Artificial Intelligence practitioners want to uplift. In this paper, we take this research a step further by including real ML models, multiple fairness metrics, and discrimination mitigation methods in our experiments to understand their relationship with the impact on groups being classified. We highlight how carefully selecting a fairness metric is not enough when taking into consideration later effects of a model's predictions–the ML model, discrimination mitigation method, and domain must be taken into account. Our experiments show that most of the mitigation methods, although they produce "fairer" predictions, actually do not improve the impact for the disadvantaged group, and for those methods that do improve impact, the improvement is minimal. We highlight that using mitigation methods to make models more "fair" can have unintended negative consequences, particularly on groups that are already disadvantaged.

Keywords
algorithmic fairness, machine learning, artificial intelligence, impacts

1. Introduction

Since the rise of Machine Learning (ML), using data to train models to make predictions has become customary. These models can help decide who makes it to the next stage of a job interview or who gets a loan—outcomes that, potentially, massively impact individuals' lives. Models can maintain or exacerbate already existing inequalities in society by outputting unfair predictions. For instance, Amazon scrapped an Artificial Intelligence (AI) tool that aided recruitment in sifting through resumes because it was sexist [1]. The model was trained on data from previously submitted resumes and the majority of those resumes were from men.

To mitigate unfair predictions, algorithmic fairness research has boomed in recent years but it has actually been around for over 50 years [2]. Many different fairness metrics, which can be used to measure how "fair" outcomes are, have been formalized, e.g., [3, 4, 5, 6, 7], and operationalized in techniques intended to ensure fairer ML predictions, e.g. [3, 4, 8, 7]. These techniques are a part of a larger set of methods called discrimination mitigation methods, e.g. [9, 10, 11]. Little consensus has been drawn as to which fairness metrics and methods are better than others [12], especially since there is no universally accepted fairness definition [13].

The fairness metrics and discrimination methods proposed have their flaws though. Recent research has shown that although the intuition behind using fairness metrics is valid, the application of the techniques can lead to harming the very groups they aim to protect, e.g. [14, 5, 15]. For instance, Liu et al. [15] used a loan repayment domain to showcase this phenomena–by using a fairness metric with an optimized threshold decision boundary, individuals, who might otherwise have been denied a loan are accepted for a loan. Although acceptance for the loan initially appears "fair," if the individuals ultimately are unable to pay, their credit scores might drop, arguably, a financial harm.

Liu et al. coined this financial harm a "delayed impact"–a later effect on a person classified. However, Liu et al. did not use a typical ML model, but an optimized decision threshold, which implied that an off-the-shelf discrimination mitigation method for applying a fairness metric could not be used on top of existing ML models. In addition, they only considered two fairness metrics, Demographic Parity and Equality of Opportunity.

In this paper, we provide the first empirical evaluation of delayed impact using actual ML models and considering different off-the-shelf discrimination mitigation methods and different fairness metrics. Through our comprehensive empirical study, we show the complex relationships that exist between real ML models, methods for discrimination mitigation, and fairness metrics, and the resulting delayed impacts on groups of users.

Through our results, we highlight that the fairness metric and discrimination mitigation method affect the delayed impact substantially more than the ML model itself. We find that with the usage of discrimination mitigation methods and fairness metrics the advantaged group benefited the majority of the time, while the disad-

AIofAI '22: 2nd Workshop on Adverse Impacts and Collateral Effects of Artificial Intelligence Technologies, Vienna, Austria
mackenzie.jorgensen@kcl.ac.uk (M. Jorgensen); elizabeth.black@kcl.ac.uk (E. Black); ncriado@upv.es (N. Criado); jose.such@kcl.ac.uk (J. Such)

vantaged group was actually worse off than without the mitigation techniques. The choice of metric is especially important. In general, our results suggest that the combination of domain, ML model choice, fairness metric, and discrimination mitigation method determines the interplay with the delayed impact.

The rest of this paper is structured as follows: the literature review in Section 2, the problem formulation and experimental design in Section 3, our experimental results in Section 4, a discussion of the results in Section 5, and our conclusion in Section 6.

2. Literature Review

In this section, we will cover fairness metrics and discrimination mitigation methods. In addition, we will outline attempts to tackle the Discriminatory Impact Problem (DIP) and subproblems that arise from it. The "Discriminatory Impact Problem," coined by Kusner et al. asks: "How can we reduce discrimination arising from *the real-world impact of decisions*?" [16].

2.1. Fairness Metrics

More than 20 fairness metrics have been proposed [17]. The definition of fairness has been a philosophical debate for centuries and, actually, it is *essentially contested* [18, 19]. Different values and assumptions (often implicit) are behind different fairness definitions and metrics [20, 19, 21]. Also, a model cannot satisfy more than one fairness metric at once, unless the case is very constrained [22, 23]. Kleinberg et al. highlight how fairness metrics can be incompatible with one another and they present a framework for understanding the trade-offs of different metrics. They reveal how calibration and balancing the two classes are incompatible. Thus, the context of a classification system is crucial in understanding what metrics should be used [23].

Measures are incredibly important because they *create society* through classification systems [24, 19]. In a more general sense, Watkins et al. argue that these systems are inherently proposals for how the world should be [25]. Friedler et al. ascertain every model has some inherent values behind its decision making [21]. They also argue that every fairness metric also has inherent values, assumptions, and aims encoded. Mitchell et al. explicitly define such assumptions for different metrics [20].

Wachter et al. highlight a comprehensive list of fairness metrics, originally from [26], and whether or not those metrics are Bias Preserving (BP) or Bias Transforming (BT) [27]. BP metrics typically keep outcomes in relation to society's status quo; often times, this means matching group error rates. BT metrics change the status quo by usually matching group outcome rates.

2.2. Discrimination Mitigation Methods

The three key discrimination mitigation method types are pre-processing (changing the data before it is used for training a model), in-processing (applying a fairness metric as a constraint or adapting the models' learning), and post-processing (updating a model's output to make it more fair) [12]. They are used at different points along the ML pipeline. In this paper, we focus on an post-processing method because we wanted to see how different ML models, fairness metrics, and discrimination mitigation methods all worked together to affect the delayed impact, so we will cover those methods here.

Calders and Verwer developed a modified naive bayes classifier that is independent of the protected attribute to make a discrimination-free model [28]. Taking another ML model, decision tree classifiers, Kamiran et al. crafted discrimination-aware tree construction and relabeling methods [29]. Kamishima et al. compared their method, an indirect prejudice remover through regularization in the objective function, to the two previously described methods [30]. Zemel et al. improved upon Dwork et al.'s theoretical framework [3] and showed that models could learn group and individual fairness [31].

Zafar et al. focused on mitigating disparate mistreatment by using a decision boundary around a fairness metric–they used a convex-concave program to solve the problem [32]. Goel et al. utilized a weighted sum of logs technique to make a model non-discriminatory [33]. They essentially defined and solved a non-convex optimization problem. Similarly, Cotter et al. solved a non-convex optimization problem through an approximate bayesian optimization oracle which focused on the Equality of Opportunity (EOO) fairness metric[34].

Celis et al. developed a meta-algorithm that achieves near-perfect fairness on multiple metrics with more than one protected attribute [35]. Agarwal et al. adopted a reduction approach with the Exponentiated Gradient and Grid Search algorithms to guarantee the most accurate fair model [8]. We use these two reduction algorithms for our discrimination mitigation method.

2.3. Discriminatory Impact Problem

Much of the algorithmic fairness research that deals with delayed or long-term impacts is inspired by economics and its principles, such as social welfare. Hu and Chen developed a model focused on the labor market, considering fairness and long-term outcomes. They showcased that by using Demographic Parity (DP) in a short-term labor market that farther down the line, an equitable long-term equilibrium could be reached in a permanent labor market [36].

Liu et al. developed an outcome curve for a loan allocation problem to measure the delayed impact on different

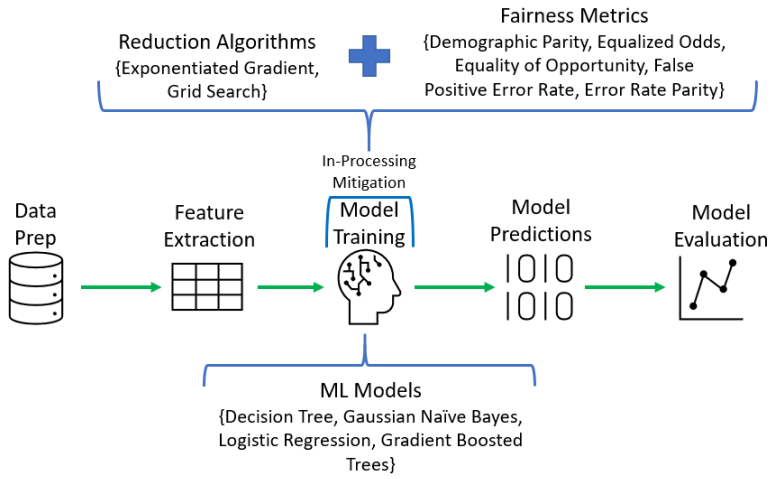

Figure 1: The ML pipeline for the experiments is here with the fairness metrics, ML models, and discrimination mitigation methods (Reduction Algorithms).

groups given DP and EOO as thresholds a bank would use to decide who gets a loan [15]. The outcome curve modeled at what point a given group would experience active harm, relative harm or relative improvement. Through their proofs, they articulate that EOO and DP both have the potential to benefit or harm groups, that DP, under certain conditions, might harm groups by overaccepting but that EOO does not, and that EOO might cause harm by under-accepting but that DP will not under-accept under a certain assumption [15]. They conducted simulations with a credit score dataset and optimized a decision threshold constrained by a fairness metric, but did not use an actual ML model nor a typical discrimination mitigation method.

In a related thread, Speicher et al. implemented a benefit function where certain outcomes have certain benefits [37]. But, unlike Liu et al., they did not consider benefit later in the future. Also, Fuster et al. ascertained that using newer prediction models led to an increase in credit provision, but that the disparities between and within-groups were exacerbated [38]. Other researchers, in the affirmative action context, developed a dynamic model where the selection rates changed over time [39], not just over one time step like in Liu et al's model.

Kusner et al. took a different approach through causal methods [16]. They proposed a constraint that reduces the discrimination coming from the impact and showcased how to efficiently solve this as a constrained optimization problem. The previous approaches focused on specific instances of the DIP; meanwhile, we inspect the problem from a different angle with more practical considerations by using multiple ML models, fairness metrics, and reduction algorithms.

3. Methods

In this section, we outline the problem formulation and the important aspects of our experiments, including the fairness metrics, ML models, discrimination mitigation methods, and delayed impact measurements used.

3.1. Problem Formulation

In our paper, we consider a binary supervised learning setting. We leave multi-class classification problems for future work. We intend to focus on the simpler binary case in our initial exploration, where we assume access to data or features, X, a binary protected attribute consisting of two demographic groups, A, and the true labels, Y. We also assume access to a model, h, trained on (X, A, Y) assuming the protected attribute, A, was used in training. The model's predictions are $\hat{Y}$.

To analyze a model's performance, confusion matrices are commonly used. Confusion matrix cells include the True Positives (TP), False Positives (FP or Type I Error), True Negatives (TN), and False Negatives (FN or Type II Error), when looking at the predicted and true classes. These terms are the building blocks for model performance metrics (e.g. accuracy, precision, recall, and F-1 score). Most fairness metrics can also be explained by TP, FP, TN, and FN [17]. From this point on, we will refer to TP, FP, TN, and FN as model outcomes.

We aim to investigate whether the findings in Liu et al.'s work (see Section 2.3) hold true in an empirical setting with actual ML models and discrimination mitigation methods. Specifically, we explore the complex relationships between ML models, discrimination mitigation

Table 1
The fairness metrics considered are listed here and where $y \in \{0, 1\}$. They are categorized as Bias Transforming (BT) or Bias Preserving (BP) metrics.

Name	BT/BP	Expression
Demographic Parity [3]	BT	$P(\hat{Y} = 1\|A = 0) = P(\hat{Y} = 1\|A = 1)$
Equalized Odds [4]	BP	$P(\hat{Y} = 1\|Y = y, A = 0) = P(\hat{Y} = 1\|Y = y, A = 1)$
Equality of Opportunity [4]	BP	$P(\hat{Y} = 1\|Y = 1, A = 0) = P(\hat{Y} = 1\|Y = 1, A = 1)$
False Positive Error Rate [5]	BP	$P(\hat{Y} = 1\|Y = 0, A = 0) = P(\hat{Y} = 1\|Y = 0, A = 1)$
Error Rate Parity [7]	BP	$P(\hat{Y} = y\|Y \neq y, A = 0) = P(\hat{Y} = y\|Y \neq y, A = 1)$

methods, and fairness metrics, and the delayed impacts on those being classified within the loan repayment domain. To find these answers, we undertake an experimental study which is defined in the next subsection.

We consider a classification problem where an ML model predicts the likelihood that a loan applicant will repay the bank if given a loan. We have different delayed impact assumptions, depending on what the benefits or losses are for different model outcomes (i.e. TP, FP, TN, and FN). In our scenario, we assume that being a FP has a higher negative weight than a TP has a positive weight with regard to the delayed impact. We will dive deeper into these ideas in Section 3.2.4.

3.2. Experiments

Through this paper, we conduct a systematic study of fairness metrics paired with discrimination mitigation methods and ML models within a single domain of focus–loan repayment prediction. For a visualization of our ML pipeline for our experiments, see Figure 1.

3.2.1. Fairness Metrics

In this section, we delve into the fairness metrics considered for our initial assessment. Our fairness goal, as Friedler et al. impresses on researchers to state, is nondiscrimination and we use group fairness metrics as the mechanism to work towards that goal [21]. We use the same formalizations from Section 3.1. The conditional probability that the hypothesis model, h, outputs a given prediction, $\hat{y}$, given a protected attribute, a, is defined as $P(\hat{y}|a)$, where $\hat{y} \in \{0, 1\}$ and $a \in \{0, 1\}$. The fairness metrics chosen are defined in Table 1. These metrics are among the most used ones (being available in the majority of fairness toolkits and libraries) and do not require expert knowledge to be used.

DP also goes by other names including Statistical Parity and Acceptance Rate. Equalized Odds (EO) has been referred to as Disparate Mistreatment but we will call it EO here. EOO and False Negative Error Rate balance [5] are the mathematically equivalent because if a model has equal True Positive Rate for both groups it will also, in turn, have an equal False Negative Rate. For the rest of this paper, we will say EOO. False Positive Error Rate (FPER) balance is also called Predictive Equality and is related to the True Negative Rate. We highlight that Liu et al. use DP and EOO in their delayed impact research. We use those two metrics again here for comparison and include three others: EO, FPER, and Error Rate Parity (ERP). All metrics we consider are Bias Preserving, except for DP which is Bias Transforming [27].

3.2.2. ML Models and Reduction Algorithms

For the purposes of creating a replicable experiment, we utilize off-the-shelf ML models from *sklearn*[1]. The models we use include: Decision Tree (DT), Gaussian Naive Bayes (GNB), Logistic Regression (LGR), and Gradient Boosted Tree (GBT) classifiers. We also used these models because their fit functions all included a sample weights parameter which was needed for using the mitigation methods from *Fairlearn*[2].

We employ the two reduction algorithms, Exponentiated Gradient (EG) and Grid Search (GS) [8], which are implemented in *Fairlearn* for our discrimination mitigation method. These algorithms take a trained ML model and a fairness constraint as parameters and reduce the binary classification to weighted classification problems. The goal of the reduction algorithms is to optimize the tradeoff between accuracy and the fairness constraint. The fairness constraints are used as Lagrange multipliers in the method[3]. One of the strengths of using the reduction algorithms as our discrimination mitigation method is that we can utilize multiple ML models in our experiments, unlike other discrimination mitigation methods which tend to be ML model specific.

3.2.3. Credit Score Data

We utilize the same dataset as Liu et al. but we transformed it into a tabular dataset that can be used by ML

[1] https://scikit-learn.org/stable/
[2] https://fairlearn.org/
[3] For more information on how the reduction algorithms work, we recommend reading Agarwal et al.'s paper for indepth explanations [8].

models [15]. The original data includes FICO scores (often times used for showing credit worthiness) preprocessed by Hardt et al., which were collected from 301,536 TransUnion TransRisk scores from 2003 [4]. Further, the data includes FICO score distributions by race, cummulative distribution functions which tell us the fraction of the group by race that falls under or below a given score, and probability mass functions which tell us what the probability of an individual (by race) repaying is given their score. The scores range from 300 to 850. For the score distributions by race, see Figure 2.

To generate a tabular dataset from the FICO score data, we randomly sample scores and their probabilities by race from the data such that the demographic racial proportions remain the same (approximately 12% Black and 88% White). We take the Black group as disadvantaged and the White group as advantaged. We created a dataset with 100k rows or 100k individuals. To generate the y label for our binary classification problem, we follow the probabilities of repayment by credit score and race.

For a visualization of the repayment indices distributions by race, see Figure 3. Our tabular dataset has two columns as features–it includes the credit score and the race of an individual as the X for training and the label as to whether they repaid or not as y. For more information about how we transformed the initial FICO score dataset, please see our Github[4].

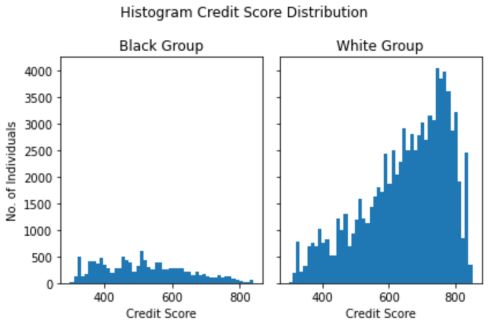

Figure 2: The histograms of the credit score distribution by race from the generated dataset.

3.2.4. Measuring Delayed Impact

We follow Liu et al.'s function of delayed impact change such that if an individual is a FP then their credit score drops by 150 points and if an individual is a TP then their credit score increases by 75 points [15]. To formalize this, before the model prediction, let us consider we have an individual's credit score s. After the model's prediction, we update that individual's score such that they have a new credit score $s*$. To calculate this change, we follow this equation: $s* = s + u$, where u is the update value, 75 for a TP individual and -150 for a FP individual. When quantifying the delayed impact from the groups, we take the average score difference of each racial group–we will refer to this as the "impact change" hereafter.

[4]https://github.com/mjorgen1/delayedimpact

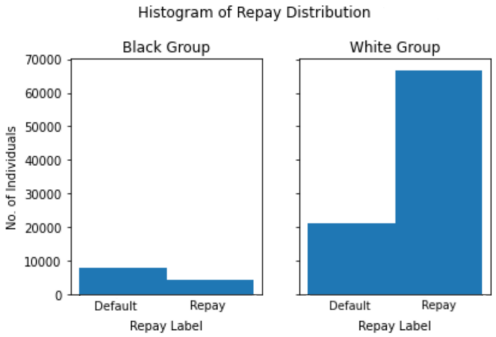

Figure 3: The histograms of the repayment indices by race which are used as the label in the classification system.

When considering the delayed impact, we only focus on TP and FP outcomes. These outcomes are the two positive class cases where an individual would then be given a loan, since they were predicted to repay the bank. In time, their credit scores would change. This scenario is especially interesting because typically being classified as the positive class has a positive or neutral benefit, but in this scenario, for individuals classified as FPs, that is not the case. Our focus on the negative delayed impact for FPs highlights the issue of predatory lending.

4. Results

For our results, we compare how the different fairness metric, reduction algorithm, and ML model combinations performed in relation to the impact change by group. To understand how those combinations affected the delayed impact, we track if the delayed impact (by group) improves (credit score increases in the case of TPs) or falls (credit score drops for FPs) in comparison to the delayed impact of the unmitigated model. We will cover the results with respect to the fairness metrics, ML models, and reduction algorithms used.

For results showing the percentage increase or decrease of the impact change in comparison to the unmitigated model, see Table 2 for the White group results and Table 4 for the Black group results. For the raw impact change results, which are key to understand the magnitude changes beyond the percentage changes, see Tables 3 and 5, where we show the raw impact for the ML models without any mitigation method for reference.

Table 2
The average impact change results for the White group for all the different fairness metric (along the left column) and reduction algorithm and ML model combinations (along the top row) are here. The percentage change (with the decimal rounded to the nearest hundredth) in impact is calculated from the unmitigated model results to the mitigated model results.

Fairness Metric	EG+DT	EG+GNB	EG+LGR	EG+GBT	GS+DT	GS+GNB	GS+LGR	GS+GBT
DP	0.61%	3.27%	1.62%	0.61%	0.61%	7.21%	1.52%	0.61%
EO	-2.7%	-0.92%	-3.24%	-3.21%	0.61%	4.11%	0.9%	0.61%
EOO	0.92%	6.13%	1.85%	1.04%	0%	3.11%	0.51%	0%
FPER	0.61%	-26.74%	1.52%	0.61%	0%	0%	0%	0%
ERP	-6.14%	-6.65%	-5.47%	-6.21%	0%	0.95%	0%	0%

Table 3
The average impact change for the White group for the unmitigated models, together with all the different fairness metric, reduction algorithm and ML model combinations. *Note the unmitigated means that no metric nor reduction algorithm were applied.

Fairness Metric	EG+DT	EG+GNB	EG+LGR	EG+GBT	GS+DT	GS+GNB	GS+LGR	GS+GBT
Unmitigated*	39.28	37.02	38.93	39.28	39.28	37.02	38.93	39.28
DP	39.52	38.23	39.56	39.52	39.52	39.69	39.52	39.52
EO	38.22	36.68	37.67	38.02	39.52	38.54	39.28	39.52
EOO	39.64	39.29	39.65	39.69	39.28	38.17	39.13	39.28
FPER	39.52	27.12	39.52	39.52	39.28	37.02	38.93	39.28
ERP	36.87	34.56	36.8	36.84	39.28	37.37	38.93	39.28

In addition, in Appendix A, we include the full breakdown of our ML model specific performance results (e.g. accuracy) in Tables 6, 7, 8, and 9.

In all the Tables, we use the following notation, where "A+M" refers to a reduction algorithm A paired with a ML model M. Recall, "EG" is the Exponentiated Gradient reduction algorithm, "GS" is the Grid Search reduction algorithm, "DT" is a Decision Tree classifier, "GNB" is a Gaussian Naive Bayes classifier, "LGR" is a Logistic Regression classifier, and "GBT" is a Gradient Boosted Tree classifier.

4.1. Fairness Metric-Focused Results

Here, we will delve into how the fairness metrics affected the two groups' impact changes. For the White group, we see that for all DP model results led to a positive impact change; meanwhile, for the Black group, all model results with DP led to a negative impact change. Our results highlight a limitation when using DP as a fairness constraint; while selection rates are forced to be as equal as possible across the groups (when the groups have different true outcomes), one group will be less qualified in comparison to the other [7]. Thus, on face value, the less qualified group could continue or begin a negative record for their demographic group. Dwork et al. coined this problem the "self-fulfilling prophecy" [3].

When EO was used with the EG reduction algorithm, we see that the White group's impact change drops slightly; contrastingly, when the GS reduction algorithm was used, the White group's impact change remains about the same or increases very slightly. With regards to the Black group's runs with EO, the impact change drops significantly, except for one run when it rises (GS+GNB).

For the runs with EOO, the White group's impact change is stagnant or minimally rises; meanwhile, the Black group's impact change drops substantially for all but one run (GS+GNB). In that single case for the Black group, the impact improves, particularly when considering the percentage increase, but when looking at the raw impact figures, one can easily see that GNB unmitigated has almost neutral impact. Further, the raw impact achieved by applying EOO with GS, while one of the highest, is still very far from the raw impact for the White group, and not that far from the highest impact for the Black group with unmitigated model (LGR).

When the FPER metric constrains the models, the White group's impact change increases slightly or remains stagnant and there is one run where the impact change falls (EG+GNB). When FPER was paired with the EG reduction algorithm, we highlight that the Black group's impact change drops significantly; surprisingly, when GS was the reduction algorithm with FPER, the impact change is stagnant.

When ERP constrains the models, the White group's impact change either drops slightly or remains the same, and in one instance, it increases extremely slightly. The Black group's impact change either increases, remains stagnant, or drops (in one case, GS+GBT). In fact, ERP

Table 4
The average impact change results for the Black group for all the different fairness metric (along the left column) and reduction algorithm and ML model combinations (along the top row) are here. The percentage change (with the decimal rounded to the nearest hundredth) in impact is calculated from the unmitigated model results to the mitigated model results. ∅ is used for the values when the unmitigated model impact was 0 to begin with, making percent changes not possible to calculate.

Fairness Metric	EG+DT	EG+GNB	EG+LGR	EG+GBT	GS+DT	GS+GNB	GS+LGR	GS+GBT
DP	-712.78%	∅	-613.22%	-694.76%	-469.97%	∅	-412.28%	-484.59%
EO	-166.61%	∅	-177.7%	-139.14%	-131.46%	∅	-230.71%	-129.43%
EOO	-325.4%	∅	-301.34%	-311.86%	-114.54%	∅	-163.02%	-129.43%
FPER	-171.25%	∅	-164.35%	-168.1%	0%	∅	0%	0%
ERP	0%	∅	0.27%	0.92%	0%	∅	0%	-1.23%

Table 5
The average impact change for the Black group for the unmitigated models, together with all the different fairness metric, reduction algorithm and ML model combinations. *Note the unmitigated means that no metric nor reduction algorithm were applied.

Fairness Metric	EG+DT	EG+GNB	EG+LGR	EG+GBT	GS+DT	GS+GNB	GS+LGR	GS+GBT
Unmitigated*	6.26	0	7.49	6.49	6.26	0	7.49	6.49
DP	-38.36	-26.62	-38.44	-38.6	-23.16	-35.72	-23.39	-24.96
EO	-4.17	-11.48	-5.82	-2.54	-1.97	6.41	-9.79	-1.91
EOO	-14.11	-16.86	-15.08	-13.75	-0.91	7.49	-4.72	-1.91
FPER	-4.46	-15.92	-4.82	-4.42	6.26	0	7.49	6.49
ERP	6.26	8.1	7.51	6.55	6.26	8.31	7.49	6.41

seems to be the only metric that has the least negative impact on the Black group from all the metrics; even though, most of the time, there is no improvement or only marginal improvement, and in one case, there is a small drop in impact. We note, however, that when ERP is used together with GNB, regardless of the reduction method, there is a significant improvement in terms of proportion for the Black group. Though in terms of total improvement, GNB constrained by ERP only offers a slight improvement for the Black group over the best impact for the unmitigated models.

Finally, we now focus only on DP and EOO to compare our results with those of Liu et al. [15], as they only used these two metrics. Our results reveal that although their findings were theoretically valid given their assumptions and the way they modelled the problem without actual ML models; in practice, with actual ML models, reduction algorithms, and different metrics, we found both similar and different results with regard to fairness metric behavior. As aforementioned, a motivation for our work was to overcome the limitations of their study and to analyse to what extent their fairness metric and delayed impact results hold when an off-the-shelf ML model and reduction algorithm are used.

In particular, our results support Liu et al.'s claim that EOO and DP both have the potential to positively and negatively impact groups. In contrast to their finding that under certain conditions EOO will not overaccept and negatively impact groups, we highlight that EOO actually over-accepts and causes a drop in impact change for the Black group when using DT and LGR models with both reduction algorithms and the GNB model with the EG reduction algorithm (see Selection Rates in Tables 6, 8, and 7 and the decrease in impact change in Table 4). The majority of our results support their claim that given an assumption, EOO might negatively impact groups by under-accepting, while DP will not. The only exception in our results in contradiction to this was DP constraining the GNB model with EG where we see DP underaccepting for the Black group (see Selection Rates in Table 7).

4.2. ML Model and Reduction Algorithm-Focused Results

Now that we have covered the impact changes for the two groups by fairness metric, let us consider how the different ML models affected the impact change. When looking column wise by race and ML model, we notice that the impact changes do not vary too much. For instance, when looking at the DT model impact change results for the Black group, we see that when EG was used, the impact change dropped for 4 out of 5 runs and when GS was used, 3 out of 5 runs had an impact change drop. When looking at the White group model results for DT, we see ML model column-wise numbers are not drastically different either; for EG, 2 runs had a slight

drop in impact while the other 3 had a slight impact increase and for GS, 3 runs had no changes to the impact while two slightly increased it.

Concerning impact change stagnating, we highlight that that happens more when using GS than when using EG as the reduction algorithm. We also note that, when comparing the two reduction algorithms (and their respective ML model pairs) for the Black group impact change, results reveal that the impact changes are less severe when GS was used for the majority of 4 out of 5 metric runs (see the first four rows in Table 4). For the White group, all GS runs either kept the impact change stagnant or increased it slightly; meanwhile, there were more fluctuations and drop in impact change when EG was used (see Table 2).

Focusing on each reduction algorithm, when using the EG reduction algorithm with DP, EO, EOO, and FPER, the Black group had a decrease in impact in comparison to the unmitigated model. We find similar but slightly different results for the GS reduction algorithm results for the Black group. We only see increases in the GS+GNB model paired with the fairness metrics: EO, EOO, and ERP. All other combinations with GS leave the Black group with no change in impact or a decrease in it.

5. Discussion

We now discuss more in detail the implications of the results we obtained. One key aspect was that we were intrigued that the delayed impact changes were not universal across the two reduction algorithms and fairness metrics. There was a great deal of variation. We present a few of our key findings below.

"Fair predictions" can result in worse impacts for both the advantaged and disadvantaged groups. As shown in our results, by just applying a fairness metric with a reduction algorithm, we have by no means a guarantee that the delayed impact on any of the groups considered will improve. In fact, we have seen that some combinations of metrics and reduction algorithms can lead to worse impacts for both groups considered.

Most "fair predictions" fail to improve impact for the disadvantaged group. While the drops in impact for the White group are rather small, and, in general, we see that the White group's impact change is not substantially different when using discrimination mitigation methods, the Black group's impact change drops significantly when using most fairness metric and reduction algorithm combinations. We see that overall, the White group does better except for a few cases when using fairness metrics. However, this result is not the same for the Black group. The Black group, for the majority of the runs, was worse off delayed impact wise as a result of using the fairness metric and reduction algorithm.

Improvement for the disadvantaged group, when achieved, is quite modest. For the rare cases where there was an improvement for the Black group, the rise in impact was minimal. The one metric that seemed to benefit the Black group, for 4 out of 8 model runs, was ERP. However, in two cases the improvement is negligible, and in the other two cases, while getting to the best impact for that group, it is a minimal improvement over the best unmitigated model impact result.

Fairness metrics are not the only aspect influencing impacts. All the three main aspects we considered, namely the ML model, the discrimination mitigation method (reduction algorithm,) and the fairness metric seemed to have an influence on the impacts. However, the ML model choice appeared to have less of an effect on the delayed impact than the discrimination mitigation method (reduction algorithms in our experiments) and fairness metric. Thus, we claim that both the discrimination mitigation method and fairness metric choice have a larger role in a group's delayed impact.

Fairness metrics results do not always match expectations. Our results highlight that the fairness metrics will not reveal much about the delayed impact unless the effect metrics have on the other outcomes is known in advance. For example, one could expect that in a case like our case study, where FP entails a negative impact, a metric aiming at having equal error rates (like FPER) would lead to fairer classifications. In this case, we concluded that optimization of that metric could lead to the error rate for the Black group being greater, which in turn makes it more difficult for the model to classify Black individuals correctly.

6. Conclusion

Through our study, we empirically investigated how delayed impact of a group changes depending on what fairness metric, discrimination mitigation method, and ML model is utilized. Our experiments highlight how AI Practitioners must not only consider the fairness metric but also the discrimination mitigation method used because they both matter when considering the domain specific delayed impact. We also argue that predictions made by "fair models" need to be closely scrutinised in terms of the impact of such predictions.

In this paper, we emphasized why the interplay between delayed impact, fairness metric, discrimination mitigation method, and ML models needs to be considered to avoid undesired delayed impacts, particularly on

disadvantaged groups. For future work, we plan to analyze the statistical significance of our results and study more datasets from other domains to better understand the relationships that we have outlined and the type of generalizations that could be made across different types of domains. We also aim to explore what other delayed impacts can arise from other outcomes such as FN ones and how to formalize them.

Acknowledgments

The PhD studentship for the first author is funded by the Engineering and Physical Sciences Research Council through the EP/S023356/1 grant.

References

[1] J. Dastin, Amazon scraps secret ai recruiting tool that showed bias against women, https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G, 2018. Accessed: 2022-01-12.

[2] B. Hutchinson, M. Mitchell, 50 years of test (un) fairness: Lessons for machine learning, in: Proceedings of the Conference on Fairness, Accountability, and Transparency, 2019, pp. 49–58.

[3] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, R. Zemel, Fairness through awareness, in: Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, 2012, pp. 214–226.

[4] M. Hardt, E. Price, E. Price, N. Srebro, Equality of opportunity in supervised learning, in: D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 29, Curran Associates, Inc., 2016.

[5] A. Chouldechova, Fair prediction with disparate impact: A study of bias in recidivism prediction instruments, Big data 5 (2017) 153–163.

[6] S. Verma, J. Rubin, Fairness definitions explained, in: 2018 IEEE/ACM International Workshop on Software Fairness (FairWare), IEEE, 2018, pp. 1–7.

[7] S. Barocas, M. Hardt, A. Narayanan, Fairness and Machine Learning, fairmlbook.org, 2019. http://www.fairmlbook.org.

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

[9] T. Calders, F. Kamiran, M. Pechenizkiy, Building classifiers with independency constraints, in: 2009 IEEE International Conference on Data Mining Workshops, 2009, pp. 13–18.

[10] S. Hajian, J. Domingo-Ferrer, A methodology for direct and indirect discrimination prevention in data mining, IEEE Transactions on Knowledge and Data Engineering 25 (2013) 1445–1459.

[11] J. Metcalf, E. Moss, et al., Owning ethics: Corporate logics, silicon valley, and the institutionalization of ethics, Social Research: An International Quarterly 86 (2019) 449–476.

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

[13] N. A. Saxena, Perceptions of fairness, in: Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 537–538.

[14] S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, A. Huq, Algorithmic decision making and the cost of fairness, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 797–806.

[15] L. T. Liu, S. Dean, E. Rolf, M. Simchowitz, M. Hardt, Delayed impact of fair machine learning, in: J. Dy, A. Krause (Eds.), Proceedings of the 35th International Conference on Machine Learning, volume 80 of *Proceedings of Machine Learning Research*, PMLR, Stockholmsmässan, Stockholm, Sweden, 2018, pp. 3150–3158.

[16] M. Kusner, C. Russell, J. Loftus, R. Silva, Making decisions that reduce discriminatory impacts, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, volume 97 of *Proceedings of Machine Learning Research*, PMLR, 2019, pp. 3591–3600.

[17] A. Narayanan, 21 fairness definitions and their politics, https://www.youtube.com/watch?v=jIXIuYdnyyk, 2018. FAT* 2018 Tutorial.

[18] W. B. Gallie, Essentially contested concepts, Proceedings of the Aristotelian Society 56 (1955) 167–198.

[19] A. Z. Jacobs, H. Wallach, Measurement and fairness, in: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, FAccT '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 375–385.

[20] S. Mitchell, E. Potash, S. Barocas, A. D'Amour, K. Lum, Prediction-based decisions and fairness: A catalogue of choices, assumptions, and definitions, arXiv preprint arXiv:1811.07867 (2018).

[21] S. A. Friedler, C. Scheidegger, S. Venkatasubrama-

nian, The (im)possibility of fairness: Different value systems require different mechanisms for fair decision making 64 (2021) 136–143.

[22] J. M. Kleinberg, S. Mullainathan, M. Raghavan, Inherent trade-offs in the fair determination of risk scores, in: C. H. Papadimitriou (Ed.), 8th Innovations in Theoretical Computer Science Conference, ITCS 2017, January 9-11, 2017, Berkeley, CA, USA, volume 67 of *LIPIcs*, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017, pp. 43:1–43:23.

[23] A. D. Selbst, D. Boyd, S. A. Friedler, S. Venkatasubramanian, J. Vertesi, Fairness and abstraction in sociotechnical systems, in: Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 59–68.

[24] G. C. Bowker, S. L. Star, Sorting things out: Classification and its consequences, MIT press, 2000.

[25] E. A. Watkins, E. Moss, J. Metcalf, R. Singh, M. C. Elish, Governing algorithmic systems with impact assessments: Six observations, in: Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society, AIES '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 1010–1022. URL: https://doi.org/10.1145/3461702.3462580. doi:10.1145/3461702.3462580.

[26] S. Verma, J. Rubin, Fairness definitions explained, in: 2018 IEEE/ACM International Workshop on Software Fairness (Fairware), IEEE, 2018, pp. 1–7.

[27] S. Wachter, B. Mittelstadt, C. Russell, Bias preservation in machine learning: The legality of fairness metrics under eu non-discrimination law, West Virginia Law Review, Forthcoming (2021).

[28] T. Calders, S. Verwer, Three naive bayes approaches for discrimination-free classification, Data Mining and Knowledge Discovery 21 (2010) 277–292.

[29] F. Kamiran, T. Calders, M. Pechenizkiy, Discrimination aware decision tree learning, in: 2010 IEEE International Conference on Data Mining, IEEE, 2010, pp. 869–874.

[30] T. Kamishima, S. Akaho, H. Asoh, J. Sakuma, Fairness-aware classifier with prejudice remover regularizer, in: P. A. Flach, T. De Bie, N. Cristianini (Eds.), Machine Learning and Knowledge Discovery in Databases, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 35–50.

[31] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, C. Dwork, Learning fair representations, in: International conference on machine learning, PMLR, 2013, pp. 325–333.

[32] M. B. Zafar, I. Valera, M. Gomez Rodriguez, K. P. Gummadi, Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment, in: Proceedings of the 26th international conference on world wide web, 2017, pp. 1171–1180.

[33] N. Goel, M. Yaghini, B. Faltings, Non-discriminatory machine learning through convex fairness criteria, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 32, 2018.

[34] A. Cotter, H. Jiang, K. Sridharan, Two-player games for efficient non-convex constrained optimization, in: Algorithmic Learning Theory, PMLR, 2019, pp. 300–332.

[35] L. E. Celis, L. Huang, V. Keswani, N. K. Vishnoi, Classification with fairness constraints: A meta-algorithm with provable guarantees, in: Proceedings of the Conference on Fairness, Accountability, and Transparency, 2019, pp. 319–328.

[36] L. Hu, Y. Chen, A short-term intervention for long-term fairness in the labor market, in: Proceedings of the 2018 World Wide Web Conference, WWW '18, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 2018, p. 1389–1398.

[37] T. Speicher, H. Heidari, N. Grgic-Hlaca, K. P. Gummadi, A. Singla, A. Weller, M. B. Zafar, A unified approach to quantifying algorithmic unfairness: Measuring individual and group unfairness via inequality indices, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '18, Association for Computing Machinery, New York, NY, USA, 2018, p. 2239–2248. URL: https://doi.org/10.1145/3219819.3220046. doi:10.1145/3219819.3220046.

[38] A. Fuster, P. Goldsmith-Pinkham, T. Ramadorai, A. Walther, Predictably unequal? the effects of machine learning on credit markets, The Effects of Machine Learning on Credit Markets (October 1, 2020) (2020).

[39] H. Mouzannar, M. I. Ohannessian, N. Srebro, From fair decision making to social equality, in: Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 359–368.

A. Extended Results

In our performance tables for each of our ML models, we include the Accuracy (Acc), F1 weighted score (F1), and Selection Rate (SR). We utilize the F1 weighted metric so we take into consideration the proportion of labels, since our dataset is imbalanced. We also considered Accuracy since it is a common performance metric for classification systems. Further, Selection Rate is especially interesting in this scenario because of the effect of being classified as the positive class. In our model result tables below, the

"Base" results column refers to our ML model without any discrimination mitigation techniques used. Then, to the right of that column we include "E" or "G" to state what reduction algorithm was used and the other acronym, after the "+" there, represents the fairness metric, e.g. "DP" for Demographic Parity. The scores shown are out of 100. For our model specific results, see Table 6 for Decision Tree model results, Table 7 for Gaussian Naive Bayes model results, Table 8 for Logistic Regression model results, and Table 9 for the Gradient Boosted Tree model results.

Table 6
The Decision Tree Classifier model performance results for all of our different combinations of runs.

	Base	E+DP	E+EO	E+EOO	E+FPRP	E+ERP	G+DP	G+EO	G+EOO	G+FPRP	G+ERP
Acc	88.32	85.07	85.63	86.96	87.75	85.4	86.37	87.92	87.96	88.32	88.43
F1	88.2	84.46	85.65	86.71	87.54	85.4	85.97	87.74	87.75	88.2	88.2
SR	72.96	77.52	70.82	74.45	74.06	71.01	75.92	73.8	74.16	72.96	72.96

Table 7
The Gaussian Naive Bayes model performance results for all of our different combinations of runs.

	Base	E+DP	E+EO	E+EOO	E+FPRP	E+ERP	G+DP	G+EO	G+EOO	G+FPRP	G+ERP
Acc	85.72	81.89	84.62	86.83	69.32	83.82	85.29	88.39	88.25	85.72	87.65
F1	85.59	81.96	84.36	86.46	70.66	83.84	84.81	88.13	87.98	85.59	87.39
SR	72.66	70.23	74.08	75.75	56.39	70.75	76.37	74.86	74.99	72.66	74.81

Table 8
The Logistic Regression model performance results for all of our different combinations of runs.

	Base	E+DP	E+EO	E+EOO	E+FPRP	E+ERP	G+DP	G+EO	G+EOO	G+FPRP	G+ERP
Acc	88.41	85.11	85.09	86.95	87.72	85.49	86.38	87.36	87.73	88.41	88.41
F1	88.24	84.51	85.08	86.67	87.51	85.57	85.98	87.06	87.44	88.24	88.24
SR	73.68	77.43	71.13	74.71	74.08	69.91	75.92	75.16	74.99	73.68	73.68

Table 9
The Gradient Boosted Tree model performance results for all of our different combinations of runs.

	Base	E+DP	E+EO	E+EOO	E+FPRP	E+ERP	G+DP	G+EO	G+EOO	G+FPRP	G+ERP
Acc	88.32	85.06	85.19	86.97	87.77	85.42	86.26	87.94	87.91	88.32	88.39
F1	88.21	84.45	85.32	86.73	87.56	85.42	85.84	87.75	87.69	88.21	88.26
SR	72.9	77.52	69.3	74.25	74.11	71.01	76.06	73.83	74.32	72.9	73.13