Assessing and mitigating unfairness in credit models with the Fairlearn toolkit

Version: September 22, 2020

Miroslav Dudík*, William Chen†, Solon Barocas*, Mario Inchiosa*, Nick Lewins*, Miruna Oprescu*, Joy Qiao*, Mehrnoosh Sameki*, Mario Schlener†, Jason Tuo†, Hanna Wallach*

**Microsoft*, †*EY*

Executive summary

As AI plays an increasing role in the financial services industry, it is essential that financial services organizations anticipate and mitigate unintended consequences, including fairness-related harms, such as denying people services, initiating predatory lending, amplifying gender or racial biases, or violating laws such as the United States' Equal Credit Opportunity Act (ECOA). To address these kinds of harms, fairness must be explicitly prioritized throughout the AI development and deployment lifecycle.

To help organizations prioritizing fairness in AI systems, Microsoft has released an open-source toolkit called Fairlearn. This toolkit focuses on the assessment and mitigation of fairness-related harms that affect groups of people, such as those defined in terms of race, sex, age, or disability status.

Using a dataset of loan applications, we illustrate how a machine learning model trained with standard algorithms can lead to unfairness in a loan adjudication scenario, and how Fairlearn can be used to assess and mitigate this unfairness. The model, which is obtained by thresholding the predictions of probability of default (PD), leads to an uneven distribution of adverse events for the "male" group compared to the "female" group even though this model does not use sex as one of its inputs. Fairlearn's mitigation algorithms reduce this disparity from 8 percentage points to 1 percentage point without any (statistically significant) impact on the cost to the financial services organization.

We emphasize that fairness in AI is a sociotechnical challenge, so no software toolkit will "solve" fairness in all AI systems. However, software toolkits like Fairlearn can still play a valuable role in developing fairer AI systems—as long as they are precise and targeted, embedded within a holistic risk management framework, and supplemented with additional resources and processes.[1]

[1] Although this white paper touches on compliance with antidiscrimination and related laws, none of the guidance or recommendations expressed in this white paper should be taken as legal advice or as a substitute for working with legal professionals to ensure that AI systems comply with applicable laws.

1. Introduction

Financial services organizations play a central role in the financial well-being of individuals, communities, and businesses. Every day, these organizations make decisions that impact people's lives, such as approving loan applications, foreclosing on mortgages, or paying out life insurance claims.

Adoption of AI systems has spread rapidly across the financial services industry. A 2019 survey by UK regulators found that two-thirds of industry participants rely on AI today and that the median firm in banking and insurance is planning to triple their deployment of AI systems in the next three years.

This growth in adoption is accompanied by growing recognition from financial services organizations, regulators, policymakers, consumer protection advocates, and community groups that AI systems may introduce risks of unfairness, discrimination, and lack of transparency into decision-making processes. Within the industry, these concerns are most prevalent in lending and insurance practices.

In his circular letter, sent in January 2019 to all insurers authorized to write life insurance in the state of New York, James Regalbuto, Deputy Superintendent for Life Insurance, stated that an insurer "should not use an algorithm or predictive model for underwriting or rating purposes unless the insurer can establish that the data source does not use and is not based in any way on race, color, creed, national origin, status as a victim of domestic violence, past lawful travel, or sexual orientation in any manner, or any other protected class. Moreover, an insurer should also not use an external data source for underwriting or rating purposes unless the use of the external data source is not unfairly discriminatory and complies with all other requirements in the Insurance Law and Insurance Regulations."

The U.S. Equal Credit Opportunity Act (ECOA), 15 U.S.C. 1691 *et seq.* also prohibits creditors from discriminating against credit applicants on the basis of race, color, religion, national origin, sex, marital status, or age; because an applicant receives income from a public assistance program; or because an applicant has in good faith exercised any right under the Consumer Credit Protection Act.

Any economic activity carries with it the risk of fairness-related harms. Human decision-making is not immune to unfairness and neither are the statistical or actuarial models that have been used in the financial services industry for decades. However, AI systems require an additional level of scrutiny and governance because they are often deployed at scale and tend to be opaque due to their use of sophisticated algorithms. As AI plays an increasing role in the financial services industry, it is therefore essential that organizations prioritize fairness throughout the AI development and deployment lifecycle.

Unfairness in AI systems

AI systems can behave unfairly for a variety of reasons, some societal, some technical, and some a combination of both societal and technical. For example, some AI systems behave unfairly because of societal biases that are reflected in the datasets used to train them. Other AI systems behave unfairly because of societal biases in the assumptions and decisions made by teams during the AI development and deployment lifecycle. Others yet behave unfairly not because of societal biases, but because of other system characteristics, such as datasets that contain too few data points about some group of people. Because it can be difficult to distinguish between these reasons, we define whether an AI system is behaving unfairly in terms of its impacts on people—i.e., in terms of fairness-related harms—and not in terms of specific causes, such as societal biases. This framework of unfairness is distinct, but intertwined with various concepts in antidiscrimination law and, in particular, with the legal doctrine of

disparate impact. However, as we mention earlier, the guidance and recommendations expressed in this white paper should not be taken as legal advice or as a substitute for working with legal professionals. Unfairness (as considered here) and discrimination (in the legal sense) should be treated as distinct issues that any organization that develops or leverages AI systems will need to address.

AI systems can cause several kinds of fairness-related harms, including harms involving people's experiences with AI systems or the ways that AI systems represent the groups to which they belong. For example, AI systems can unfairly allocate opportunities, resources, or information; as another example, AI systems can also fail to provide the same quality of service to some people as they do to others.

Because there are many sources of unfairness, it is not possible to "debias" an AI system or to guarantee fairness; the goal is to mitigate fairness-related harms as much as possible. Prioritizing fairness in AI often means making trade-offs based on competing priorities and the specifics of a given situation, so there are seldom objectively "right" answers. It is therefore important to be explicit and transparent about those priorities and specifics. Moreover, there is no single definition of fairness that will apply equally well to all AI systems, and different definitions of fairness are often in tension with one another.

Allocational harms in loan adjudication

In this white paper, we focus on the unfair allocation of resources (loans) by an AI system deployed in a loan adjudication scenario. We consider a machine learning model that predicts the probability of default (PD) for loan applicants: applicants with a PD above a certain threshold are screened out, while applicants with a PD below the threshold are further reviewed and considered for the loan. Because this system limits access to loans, it can lead to allocational harms for loan applicants. We show how the Fairlearn toolkit, an open-source project started by Microsoft, can be used to assess and mitigate allocational harms in this setting. We note that this machine learning model might exhibit additional fairness-related harms (e.g., regarding the accuracy of predictions experienced by different branches of a financial services organization), but we only focus here on unfair allocation of loans to applicants.

2. Fairlearn

Fairlearn is an open-source Python toolkit for assessing and improving the fairness of AI systems. The design of Fairlearn reflects the understanding that there is no single definition of fairness and that prioritizing fairness in AI often means making trade-offs based on competing priorities. Fairlearn enables data scientists and developers to select an appropriate fairness metric, to navigate trade-offs between fairness and model performance, and to select an unfairness mitigation algorithm that fits their needs.

Fairlearn focuses on fairness-related harms that affect groups of people, such as those defined in terms of race, sex, age, or disability status. Fairlearn supports a wide range of fairness metrics for assessing a model's effects on groups of people, covering both classification and regression tasks. These fairness metrics can be evaluated using an interactive visualization dashboard, which also helps with navigating trade-offs between fairness and model performance. Besides the assessment component, Fairlearn also provides a range of unfairness mitigation algorithms that are appropriate for a wide range of contexts.

Fairness metrics

Fairness metrics quantify the extent to which a model satisfies a given definition of fairness. Fairlearn covers several standard definitions of fairness for binary classification, as well as definitions that are

appropriate for regression. These definitions either require parity in model performance (e.g., accuracy rate, error rate, precision, recall) or parity in selection rate (e.g., loan approval rate) between different groups defined in terms of a sensitive feature like "sex" or "age." We note that the sensitive feature need not be used as an input to the model, it is only required when evaluating fairness metrics.

For example, in classification settings where a more accurate prediction corresponds to a better user experience (e.g., spam detection or fraud detection), the following definitions might be appropriate:

- *Bounded group loss*: The accuracy rate for each group should be above some level that leads to an acceptable quality of service. The corresponding fairness metric is the difference between the worst-case accuracy rate (i.e., lowest accuracy rate across all groups) and the desired level.
- *Accuracy-rate parity*: The accuracy rates across all groups should be equal. The corresponding fairness metric is the difference between the groups' largest and smallest accuracy rates.

On the other hand, in classification settings where being classified as "positive" results into an allocation of an opportunity or resource (e.g., a loan) and having a positive label in a dataset means that the corresponding individual is "qualified," the following fairness definitions might be appropriate:

- *Demographic parity*: Individuals within each group should be classified as positive at equal rates. Equivalently, the selection rates across all groups should be equal.
- *True-positive-rate parity*: The qualified individuals in each group should be classified as positive at equal rates. Equivalently, the true-positive rates across all groups should be equal.
- *Equalized odds*: The qualified individuals within each group should be classified as positive at equal rates; the unqualified individuals within each group should also be classified as positive at equal rates. Equivalently, the true-positive rates across all groups should be equal and the false-positive rates across all groups should be equal.

Interactive visualization dashboard

Fairlearn's interactive visualization dashboard can help users to (a) assess which groups of people might be negatively impacted by a model and (b) compare the fairness and performance of multiple models.

When setting up the dashboard for fairness assessment, the user selects (a) the sensitive feature (e.g., "sex" or "age") that will be used to assess the fairness of one or multiple models and (b) the performance metric (e.g., accuracy rate) that will be used to assess model performance. These selections are then used to generate visualizations of a model's impact on groups defined in terms of the sensitive feature (e.g., accuracy rate for "female" and accuracy rate for "male," as defined in terms of the "sex" feature). The dashboard also allows users to compare the fairness and performance of multiple models, enabling them to navigate trade-offs and find a model that fits their needs.

Unfairness mitigation algorithms

Fairlearn includes two types of unfairness mitigation algorithms—postprocessing algorithms and reduction algorithms—that are intended to help users improve the fairness of their AI systems. Both types operate as "wrappers" around any standard classification or regression algorithm.

Fairlearn's postprocessing algorithms take an already-trained model and transform its predictions so that they satisfy the constraints implied by the selected fairness metric (e.g., demographic parity) while optimizing model performance (e.g., accuracy rate); there is no need to retrain the model. For example,

given a model that predicts the probability of default, a postprocessing algorithm will try to find a threshold above which an applicant should be rejected. This threshold typically needs to be different for each group of people (defined in terms of the selected sensitive feature). We emphasize that this limits the scope of postprocessing algorithms, because sensitive features may not be available to use at deployment time, may be inappropriate to use, or (in some domains) may be prohibited by law.[2]

Fairlearn's reduction algorithms wrap around any standard classification or regression algorithm, and iteratively (a) re-weight the training data points and (b) retrain the model after each re-weighting. After 10 to 20 iterations, this process results in a model that satisfies the constraints implied by the selected fairness metric while optimizing model performance. (The fact that it is possible to find such a model by merely re-weighting the data and retraining a standard algorithm is, at first glance, surprising, but this approach is backed by mathematical theory.) We note that reduction algorithms do not need access to sensitive features at deployment time, and work with many different fairness metrics. These algorithms also allow for training multiple models that make different trade-offs between fairness and performance, which users can compare using Fairlearn's interactive visualization dashboard.

3. Fairer loan adjudication models with Fairlearn: Case study

When making a decision to approve or decline a loan, financial services organizations gather data from the applicant, third parties, and internal sources to assess the applicant's creditworthiness. Several models contribute to the decision, including a model that predicts the applicant's probability of default (or PD), referred to as a PD model. The probability of default (PD) is defined as the probability that the applicant will fall behind on payments by more than 90 days during the coming year.

A large portion of retail lending decisions are made automatically, based on what is referred to as an adjudication strategy. For example, an adjudication strategy might state that the loan is approved automatically if the applicant's PD is below a certain threshold and a series of "review rules," defined by the financial organization, are met. Conversely, the adjudication strategy might state that the loan is rejected automatically if the applicant's PD is above a certain threshold. In this case, the applicant has a right to an explanation, usually provided as a list of reasons, codified as "reason codes," that are automatically obtained from the underlying PD model. Recourse to an automatic rejection is only available through manual review—which is rare—or by resubmitting the loan application.

PD models can lead to fairness-related harms for applicants with certain characteristics, including membership in legally protected classes defined by ECOA. Model creators must examine such models for evidence of unfairness and apply mitigation strategies to achieve a model that is deemed acceptable. Model validators and independent auditors challenge model assessments, and management must grant approval before models are deployed. The organizations' risk-management practices in model development are overseen by regulatory bodies, which can impose fines or restrictions on operations.[3] Civil society places high expectations on financial services organizations to play a role in community building, including that they do not unfairly withhold opportunities or resources. Consumer rights

[2] In the context of ECOA, there is a narrow carve-out in Regulation B for different age-specific thresholds, but not for thresholds specific to other sensitive features, which correspond to "protected classes."

[3] See SR 11-7 and E23 for examples of regulatory requirements for model risk management in the U.S. and Canada.

advocacy groups argue that systems that distribute such important opportunities and resources should be transparent and fair, and that customers have a right to an explanation when a loan is declined. Financial services organizations need to take all of these considerations into account when they design their policies and procedures for managing and monitoring risks, including the risk of unfairness.

In this white paper, we focus on the use of a PD model for automatic loan rejection—a scenario that can be viewed through the lens of binary classification. We first show how a PD model trained with a standard machine learning algorithm (specifically, LightGBM) can lead to unfairness that affects groups defined in terms of the sensitive feature "sex," even though "sex" is not used as an input to the model. We then show how the Fairlearn toolkit can be used to assess and mitigate this unfairness.

We note that there are other important use cases for PD models, which we do not consider here and which do not take the form of binary classification. For example, instead of an automatic loan approval or loan rejection, the PD model might be used to decide what level of approval authority is required to approve the loan or what loan conditions, such as the interest rate, are offered to the applicant.

Next we describe the dataset used to develop the PD model and used to assess its fairness. We then discuss the choice of performance metrics, based on the business objectives of financial services organizations, and the choice of fairness metrics, based on the potential for allocational harms.

Dataset

The dataset consists of over 300,000 loan applications. Each application is represented as 121 input features that reflect personal information, existing credit product information, credit bureau information, and real-estate information. Personal information includes sensitive features such as sex, age, postal code, and marital status. Each application is labeled according to whether the applicant defaulted on the loan (label 1) or did not default on the loan (label 0). The dataset is severely imbalanced with respect to the labels: only 8% data points are labeled as 1 (i.e., "default"). We use 70% of the dataset for training (i.e., the "train" portion) and 30% for assessing fairness and model performance (i.e., the "test" portion); the train and test portions are stratified according to the labels.

Machine learning task and performance metrics

We use the loan application dataset to train a PD model to predict the probability of default; the probability is then thresholded and a "positive" prediction (i.e., PD above the threshold) corresponds to an automatic loan rejection, whereas a "negative" prediction (i.e., PD below the threshold) corresponds to an automatic loan approval (for simplicity, we assume that the "review rules" are always met). The downstream effects of the thresholded PD model are fully determined by this binary decision, so we analyze the thresholded PD model as a binary classifier. In the mitigation stage, we also consider other binary classifiers, which are not necessarily based on thresholding the probability of default.

From the perspective of a financial services organization, there are two kinds of adverse events that can occur: **False positives** are rejections of applicants that would not have defaulted, which reduce the organization's profits. **False negatives** are approvals of applicants that do ultimately default, which increase the organization's default risk. The costs of these two kinds of adverse events are not equal.

To assess the performance of the PD model, we will therefore use **false positive rate** and **false negative rate**. We will also use **cost rate** and **weighted error rate** to measure the cost impact on the organization.

To define these metrics, we use the following notation:

n	**#samples**	
N	#actual negatives	(true="no default" in data)
P	#actual positives	(true="default" in data)
FP	#false positives	(true="no default", predicted="default")
FN	#false negatives	(true="default", predicted="no default")
cost_{FP} and cost_{FN}	cost of a false positive and a false negative	

The performance metrics are then defined as follows:

- **False positive rate (FPR):** the fraction of applicants that do not default (i.e., are labeled as 0 in the dataset), but are incorrectly predicted to default, $\text{FPR} = \text{FP}/\text{N}$.
- **False negative rate (FNR):** the fraction of applicants that default (i.e., are labeled as 1 in the dataset), but are incorrectly predicted to not default, $\text{FNR} = \text{FN}/\text{P}$.
- **Cost rate:** the average per-decision cost,

$$\frac{\text{FP} \cdot \text{cost}_{\text{FP}} + \text{FN} \cdot \text{cost}_{\text{FN}}}{\text{n}}.$$

  The cost rate can be rewritten in terms of FPR and FNR as

$$\text{w}_{\text{FPR}} \cdot \text{FPR} + \text{w}_{\text{FNR}} \cdot \text{FNR}$$

  where $\text{w}_{\text{FPR}} = \frac{\text{N} \cdot \text{cost}_{\text{FP}}}{\text{n}}, \text{w}_{\text{FNR}} = \frac{\text{P} \cdot \text{cost}_{\text{FN}}}{\text{n}}$.

Instead of working with the cost rate directly, we will work with its scaled version, which is obtained by scaling costs so that $\text{w}_{\text{FPR}} + \text{w}_{\text{FNR}} = 1$, while keeping the ratio of the costs unchanged. We assume that the cost of a false negative (i.e., approving an applicant that ultimately defaults) is 11.5 times larger than the cost of a false positive (i.e., rejecting an applicant that would not have defaulted), which yields $\text{w}_{\text{FPR}} = \text{w}_{\text{FNR}} = 0.5$. The chosen cost ratio (i.e., 11.5) is in line with the cost asymmetry we would expect in this scenario, but it has an additional benefit of yielding a simple performance metric. We refer to this scaled metric as the **weighted error rate**, because it is a weighted average of FPR and FNR:

- **Weighted error rate:** $0.5 \cdot \text{FPR} + 0.5 \cdot \text{FNR}$

Although FPR and FNR are weighted equally, there are only 8% of actual positives (defaulting applicants) in the data, so this weighted error rate indeed up-weights false negatives relative to false positives compared to the standard (i.e., unweighted) error rate. We do not use the standard (i.e., unweighted) error rate, because it does not take into account the highly asymmetric costs for the adverse events.

Fairness tasks and fairness metrics

We consider two fairness tasks: fairness assessment and unfairness mitigation. We focus on fairness-related harms that affect the "male" and "female" groups defined in terms of the "sex" feature.

We consider the same two adverse events, but now from the perspective of applicants: **False positives** (i.e., rejections of applicants that would not have defaulted) are harmful, because they withhold resources (loans) from applicants. **False negatives** (i.e., approvals of applicants that do ultimately

default) are also harmful, because applicants that default on their loans have their credit scores lowered, their names passed on to collection agencies, and may potentially go bankrupt.

To assess between-group differences in the occurrence of these adverse events, we evaluate:

- **FPR difference**: the absolute difference between false positive rates for the "male" group and the "female" group, defined as $|\mathrm{FPR}(\text{"male"}) - \mathrm{FPR}(\text{"female"})|$.
- **FNR difference:** the absolute difference between false negative rates for the "male" group and the "female" group, defined as $|\mathrm{FNR}(\text{"male"}) - \mathrm{FNR}(\text{"female"})|$.
- **Equalized odds difference:** the maximum of the FPR difference and the FNR difference.

When the equalized odds difference equals zero, the two groups (i.e., "male" and "female") have equal false positive rates and equal false negative rates. This property corresponds to the standard quantitative fairness definition, which we introduced earlier, called *equalized odds*, hence the name. These fairness metrics, alongside many other metrics, are part of the Fairlearn module **fairlearn.metrics.**

Initial model

We obtain the initial PD model by fitting an ensemble of decision trees to the train portion of the dataset using the LightGBM library with the default parameters listed below:

boosting_type	"gbdt"
learning_rate	0.1
n_estimators	100

num_leaves	31
min_child_samples	20
min_child_weight	0.001

In words, we train an ensemble of 100 trees using the *gradient boosted decision tree* algorithm with a learning rate of 0.1. The parameters limit overfitting by restricting the trees to have a maximum of 31 leaves, a minimum of 20 data points in each leaf, and the sum of Hessians of at least 0.001 in each leaf.

Fairness assessment with Fairlearn

We threshold the PD model using a threshold that yields the optimal performance according to the weighted error rate. This yields the following weighted error rate and equalized odds difference:

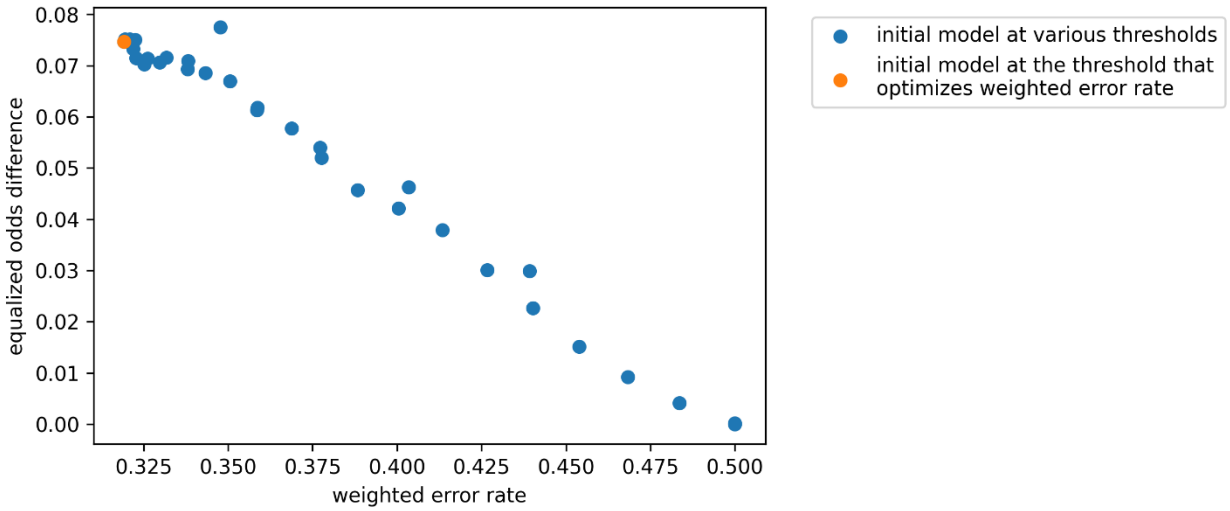
Weighted error rate	0.32 ±0.006
Equalized odds difference	0.08 ±0.007

	False positive rate (true=no default predicted=default)	**False negative rate** (true=default, predicted=no default)
female	0.27 ±0.004	0.37 ±0.014
male	0.35 ±0.006	0.31 ±0.016
overall	0.30 ±0.003	0.34 ±0.011
difference	0.08 ±0.007	0.06 ±0.021

The above table shows that adverse events are distributed unevenly between the two groups (i.e., "male" and "female"). Defaulting female applicants are more likely to be classified as non-defaulting compared to defaulting male applicants (by 6 percentage points), while non-defaulting male applicants

are more likely to be classified as defaulting compared to non-defaulting female applicants (by 8 percentage points). This means that female applicants are more likely to default when given a loan, while male applicants are more likely to have their loan rejected when they would be able to pay it off.

In the plot below, we show how fairness and performance change as we vary the threshold that is applied to PD model. For example, reducing the equalized odds difference to 0.04 means increasing the weighted error rate to 0.40. In contrast, the mitigation algorithms from Fairlearn can reduce the equalized odds difference to 0.01 without a statistically significant increase in the weighted error rate.



Unfairness mitigation with Fairlearn: Postprocessing

The unfairness described above can be mitigated without much of an impact on the weighted error rate so long as we allow a separate threshold for each group (i.e., "male" or "female"). The approach of picking such group-specific thresholds was proposed by Hardt, Price, and Srebro (2016) and is available in Fairlearn as **fairlearn.postprocessing.ThresholdOptimizer**. This algorithm takes as its input a scoring function that underlies an existing classifier (in this case, the initial LightGBM model) and identifies a separate threshold for each group in order to optimize the performance metric (in this case, weighted error rate) while simultaneously satisfying the constraints implied by the selected fairness metric (in this case, equalized odds difference). The classifier obtained using this algorithm successfully mitigates the unfairness described above, without very much of an impact on the model's performance:

Weighted error rate	0.32 ±0.006
Equalized odds difference	0.01 ±0.022

	False positive rate (true=no default predicted=default)	**False negative rate** (true=default, predicted=no default)
Female	0.31 ±0.004	0.34 ±0.014
Male	0.31 ±0.005	0.35 ±0.017
overall	0.31 ±0.003	0.34 ±0.011
difference	0.00 ±0.006	0.01 ±0.022

However, we note that the resulting classifier uses group-specific thresholds, and therefore requires access to the "sex" feature at deployment time, which is, for instance, illegal under ECOA.

Unfairness mitigation with Fairlearn: Reduction algorithms

Fairlearn also includes two reduction algorithms—**fairlearn.reductions.GridSearch** and **fairlearn.reductions.ExponentiatedGradient**—introduced by Agarwal et al. (2018, 2019). Reduction algorithms act as wrappers around any standard classification or regression algorithm, such as LightGBM. They create a sequence of reweighted datasets and retrain the model on each of them. The retraining process is guaranteed to find a trained model that satisfies the constraints implied by the selected fairness metric (in this case, equalized odds difference) while optimizing the performance metric (in this case, weighted error rate). We note that unlike postprocessing algorithms, reduction algorithms do not need access to the sensitive feature (in this case, "sex") at deployment time.

Running the grid search algorithm on the "train" portion of the dataset yields a classifier that behaves similarly (in terms of fairness and performance) to the classifier obtained using the postprocessing algorithm, but without requiring deployment-time access to the sensitive feature (i.e., "sex"):

Weighted error rate	0.32 ±0.006
Equalized odds difference	0.01 ±0.021

	False positive rate (true=no default predicted=default)	**False negative rate** (true=default, predicted=no default)
female	0.32 ±0.004	0.32 ±0.014
male	0.33 ±0.005	0.33 ±0.016
overall	0.32 ±0.003	0.32 ±0.011
difference	0.01 ±0.006	0.01 ±0.021

Navigating trade-offs between fairness and performance with Fairlearn

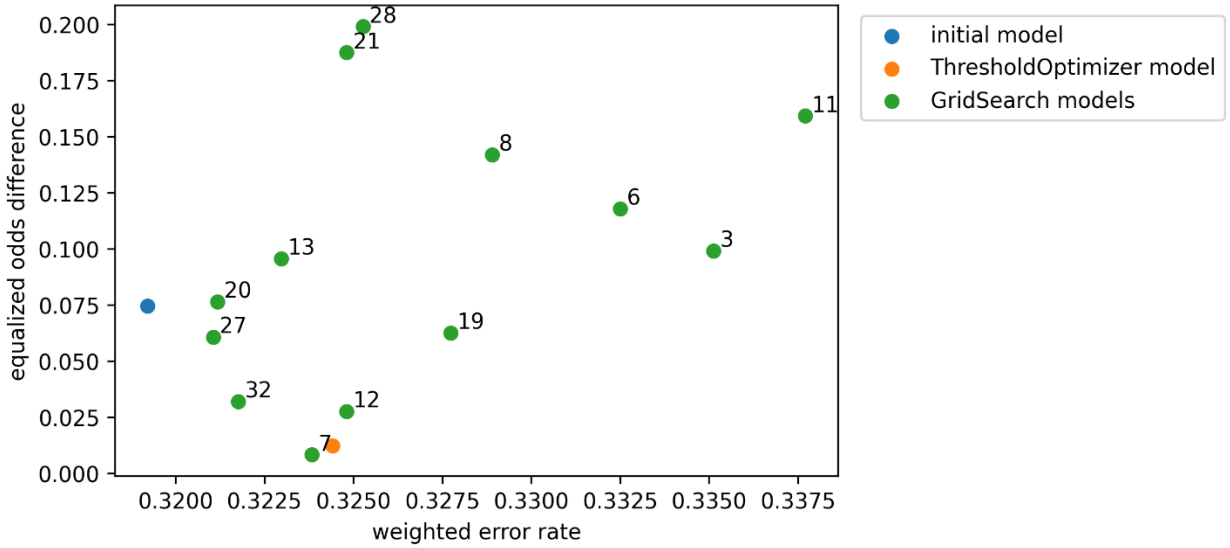
In addition to optimizing the performance metric (in this case, weighted error rate) while simultaneously satisfying the constraints implied by the selected fairness metric (in this case, equalized odds difference), reduction algorithms can also be used to navigate trade-offs between fairness and model performance. This is because the models generated by the retraining process have varying fairness (according to the fairness metric) and performance (according to the performance metric).

The plot below shows the fairness and performance of (a) the initial model (obtained using the performance-optimizing single threshold), (b) the mitigated model obtained using the postprocessing algorithm (ThresholdOptimizer), as well as (c) several mitigated models obtained using the grid search algorithm (GridSearch). We note that the table above was generated using grid search model #7.

Models that are closer to the bottom-left corner of the plot are fairer and better performing than models that are closer to the top-right corner. The initial model achieves the best weighted error rate, but its equalized odds difference is quite high. In contrast, the models obtained using the postprocessing and grid search algorithms have much lower equalized odds differences, with only negligible (and statistically insignificant) increase in weighted error rate (note the scale on the horizontal axis). Although

we selected (and reported results from) the grid search model with the lowest equalized odds difference, there are other models (e.g., #32 and #27) that are possible candidates for deployment.

Note that the trade-offs obtained by GridSearch are substantially better than the trade-offs obtained by thresholding the initial model using a single threshold, as discussed above. Any threshold that achieves a weighted error rate under 0.34 leads to an equalized odds difference of 0.06 or more.



4. Conclusion and discussion

In this white paper, we demonstrated how the Fairlearn open-source toolkit can be used to assess and improve the fairness of a loan adjudication model. Our analysis has several limitations. First, we implicitly assumed that each group has the same cost for each adverse event (i.e., false positive or false negative), and the selected fairness metric (equalized odds difference) treated disparities in the occurrence of the two adverse events symmetrically. A more refined analysis would consider a detailed utility model that takes into account both the fact that each adverse event might have a different cost for each group and also the fact that the relative cost of the two adverse events is different. Second, although the model did not use "sex" as an input, it used other features whose use is regulated, such as features related to age or marital status. The use of these features would need to be reviewed for compliance with the law (such as ECOA). Third, even if the use of these features were to pass review, the fact that the training algorithm (e.g., grid search) was informed by sensitive features, including the "sex" feature, might be legally problematic—a topic that is currently being debated in the broader context of antidiscrimination law (e.g., by Bent, Harned and Wallach, Hellman, Nachbar).

Although we focused on the use of AI in a loan adjudication scenario, AI systems are increasingly used throughout the entire credit lifecycle. After an applicant is approved for a loan, AI systems are often used to determine the loan amount and the interest rate that are offered to the applicant. During the remaining life of a credit product, a financial services organization can also use AI systems for account management based on the account behaviour, credit bureau information, and other customer information. AI systems can support pre-approval of credit-card limit increases or assist with a collection strategy when a customer exhibits difficulty with repaying. AI systems are used extensively by financial services organizations; prioritizing fairness in AI is therefore fundamental to their success.

We focused on assessing and mitigating fairness-related harms that affect groups of people, such as those defined in terms of race, sex, age, or disability status—an approach referred to as group fairness—but we note that there are other ways of conceptualizing fairness in AI systems that may be applicable in the financial services industry. For example, individual fairness requires that similar individuals be treated similarly. In some contexts, it might be desirable to require both group fairness and individual fairness. Fairness can also be conceptualized via the lens of causal reasoning, which can, for example, help financial services organizations answer counterfactual questions relating to sensitive features.

Although this white paper focuses on the use of a software toolkit, we emphasize that fairness in AI is a sociotechnical challenge, so no software toolkit will "solve" fairness in all AI systems. That is not to say that software toolkits cannot play a role in developing fairer AI systems—simply that they need to be precise and targeted, embedded within a holistic risk management framework that considers AI systems' broader societal context, and supplemented with additional resources and processes.

We hope that this white paper has demonstrated the promise of using toolkits like Fairlearn in the financial services industry, and we hope that this use, over time, will inform Fairlearn's development.

Acknowledgements

Some of the material presented here was based on content developed by Microsoft's Aether Fairness and Inclusiveness Working Group. We would also like to thank Merisa Heu-Weller and Ben Glatstein from Microsoft for many helpful comments and Kathleen Walker for editing contributions.