# Team 4: Fairness Labels for Machine Learning Pipelines

Ke Yang, Tin Luu, Hong Gong
New York University
{ky630,tbl245,hg1153}@nyu.edu

## ABSTRACT

With machine learning (ML) systems extensively applied across different applications involving massive amounts of data on characteristics of individuals, researchers have realized the importance of incorporating responsibility into these ML systems. While most issues can be traced back to the quality of the data employed in these systems, limited works have been done to provide system-level supports for users to develop, evaluate, and manage their built ML pipelines considering issues, such as fairness and transparency, which have consequential impact on the ML-assisted decisions for historically underrepresented groups.

Our goal is to fill in this gap, by providing a system for any user-uploaded tabular data set that supports operations, from upstream data pre-processing to downstream model learning and from multiple types of fairness interventions, and tracks important fairness-related statistics through this system, visually presented to users as fairness labels. Through this system, users are able to evaluate the impact of various operations, like upstream data operations and fairness interventions, on sub-populations, especially the impact on the sub-populations considering intersectional effects.

## 1 INTRODUCTION

With machine learning (ML) systems extensively applied across different applications that involve massive amounts of data on characteristics of individuals, the need of incorporating responsibility into these ML systems has never been more urgent. Several lines of recent work have pointed out the importance of earlier data life stages in evaluating and managing the data quality issues [4, 7, 8]. However, limited works have been done to provide system-level supports for users, especially those with limited ML background, to develop, evaluate, and manage their built ML pipelines considering issues, such as fairness and transparency, which have consequential impacts on the ML-assisted decisions for historically minority groups.

More importantly, because most lines of fairness research are on issues in various different domains and address them in ad-hoc manners, it becomes problematic for users to evaluate the fitness of different fairness measures and interventions for their own data sets with which they want to build a responsible ML method. Another problem with the existing fairness interventions is their focus and analysis on only a single sensitive attribute. Users, however, often face decisions involving more than one sensitive attribute in real life and need to understand how a fairness intervention on one sensitive attribute, to what extent, affects groups with another sensitive attribute. In social science, this phenomenon is known as intersectionality [2] or interlocking oppression [5], which aims to understand how various aspects of users' social and political identities, like gender and race, might combine to create unique modes of discrimination. Even though there are a few methods working with more than one sensitive attributes [1, 3, 6, 9], they focus on combining all the sensitive attributes together to ensure fairness for all of them, and their settings are not able to capture intersectional effects arising from the intersection among multiple sensitive attributes. We use intersectional effects and intersectionality interchangeably in the rest of paper.

In this paper, our goal is to fill this gap and to assist in minimizing such problems, by building a system for user-uploaded tabular data sets that supports operations, from upstream data pre-processing to downstream model learning and from multiple types of fairness interventions, and tracks important fairness-related statistics through this system, visually presented to users as fairness labels. Specifically, we provide two types of fairness labels for a given data set: *static labels* and *performance labels* (for data set including predictions from a ML model) to capture the distribution of target variable and performance metrics of different sub-populations on provided characteristics. With this system, users can evaluate the impact of various operations on sub-populations with considerations of intersectional effects, by comparing the generated fairness labels.

This paper makes the following contributions:

- A system that, first, supports multiple operations from earlier life stages of data to downstream ML models and, second, integrates most of existing fairness interventions.
- The proposal of two types of fairness labels that can be tracked through the system, providing interpretable explanations of fairness issues.
- Experimental results from the applications of our system to various data sets often used as benchmarks

in fariness literature, and a new data set, with the discussion on the observed intersectional effects on sub-populations shown by the fairness labels.

## 2 PROBLEM STATEMENT & APPROACH

In this section, we specify the problems we aim to solve and explain the ideas behind our fairness labels.

### 2.1 Problem Statement

Our goals are to 1) build a system that, in a consistent interface, supports the standard upstream operations in earlier-life stages of data, downstream ML models, and fairness interventions, 2) design the tracking of important fairness-related statistics for various populations constraint by various sensitive attributes, and 3) visually present these labels to users in an informative manner.

### 2.2 Fairness Labels

We define two types of fairness labels, static and performance, to track important fairness-related statistics.

**A static label** gives the distribution a sub-population (defined by sensitive attributes) for the binary categories of the user-specified target variable. This label type is generated from a tabular data set prior to running any steps in a ML pipeline. When there are more than one sensitive attribute, static labels for the sub-populations are generated as a linear combinations of all the sensitive attributes. We show an example of static label below.

**A performance label** gives the distribution of performance metrics of all the sub-populations defined by the sensitive attributes. These metrics are typically used to evaluate classification tasks and define fairness measures. Performance label is generated for the dataset in which the predictions of the user-specified target variable exist, usually after applying some model-involved operations. The performance metrics supported in our system are the standard evaluation metrics, for example, False Positive Rate (FPR), False Negative Rate (FNR), True Positive Rate (TPR), and True Negative Rate (TNR), and Accuracy (ACC).

Example 1. *Consider the benchmark dataset Adult Income and its two sensitive attributes gender and race, we use it to illustrate the generation of static and performance label. The target variable used frequently in this data is a binary variable - "income", of which the positive or favorable outcome is "income >50K" and the other value represents the negative outcome. An example of the generated static label of this dataset before applying any ML pipeline is listed as follow.*

$$SL(race) = \{(white) : \{+ : 0.95, - : 0.88\}, \\ (black) : \{+ : 0.05, - : 0.11\}\} \tag{1}$$

$$SL(race, gender) = \\ \{(white, men) : \{+ : 0.86, - : 0.63\}, \\ (white, women) : \{+ : 0.14, - : 0.37\}, \\ (Black, men) : \{+ : 0.77, - : 0.46\}, \\ (Black, women) : \{+ : 0.23, - : 0.54\}\} \tag{2}$$

Where each value ranges in $[0, 1]$ representing the population ratio of a specific sub-population normalized by its previous parents. + and − represent the positive and negative outcomes respectively. Thus, the values of a specific target (positive or negative) from the sub-populations (white men and white women) within a specific previous parent group (white) sums up to 1.

An example of the generated performance label of this dataset after applying a Logistic Regression model to predict the target variable is listed as follow:

$$PL(race, gender) = \\ \{(white, men) : \{TPR : 0.9, FPR : 0.9, \ldots, ACC : 0.9\}, \\ (white, women) : \{TPR : 0.8, FPR : 0.7, \ldots, ACC : 0.6\}, \\ (Black, men) : \{TPR : 0.8, FPR : 0.95, \ldots, ACC : 0.9\}, \\ (Black, women) : \{TPR : 0.7, FPR : 0.9, \ldots, ACC : 0.8\}\} \tag{3}$$

Where each value inside a performance label represents the value of a specific metric ranging in $[0, 1]$.

Through these two types of fairness labels, users can track 1) the representation of groups in the ground truth and how it is affected by various upstream data operations and fairness interventions through static label, 2) the group-level performance of a ML model and how it is impacted by the model selection and various fairness interventions through performance label. For example, users are able to track how the fairness interventions on one sensitive attribute (gender) affect groups from another sensitive attribute (race) by comparing the static labels and performance labels of the corresponding sub-populations.

## 3 IMPLEMENTATION

To support most standard operations of a ML pipeline, we build our system on the basis of *scikit-learn* and *AI Fairness 360*, including modules from *sklearn.preprocessing* and *sklearn.model_selection*, classification modules supported by *scikit-learn*, and various fairness interventions implemented in *AI Fairness 360*. We split a ML pipeline into three stages: pre-process, model, and post-process stages. Pre-process stage includes all the standard upstream data operations, from data cleaning to fairness pre-process interventions. Model stage includes the typical classification models, from Decision Tree to fair classifiers. Post-process stage includes

all the fairness post-process interventions. In the following, we list all the modules implemented in our system at each stage.

- Pre-process stage:
  - Splitter: *random* and **balance target** splitter
  - Sampler*: *random* and balance population sampler
  - Imputer: *drop missing values*, *mode*, and *datawig* imputer
  - Scaler: *min-max* and *standard* scaler
  - Categorizer*: *discretizer* and *binarizor*
  - Encoder: *one-hot*, **ordinal**, and **customized** encoder
  - Sensitive attribute encoder: **customized** encoder
  - Fair pre-processor*: *reweighing*, *learning fair representation*, and *disparate impact remover*
  - Filter*: **row and column filter**
- Model stage:
  - Classifier: *logistic regression*, *decision tree* and its optimized version with hyper-parameter tuning
  - Fair classifier*: *adversarial de-biasing learning* and *meta fair classifier*
- Post-process stage:
  - Fair post-processor*: *equal odds* and *calibrated equal odds post-processing*

Where all the operations with symbol $*$ are optional to build a ML system and all the bold operations are operations we implemented by ourselves. All the italicized operations are implemented using the corresponding *scikit-learn* and *AI Fairness 360* modules and integrated into our system with a consistent interface. Readers are encouraged to refer their documentation for more information. We implement fairness label based on pandas groupby function and visualize it using *seaborn* library.

# 4 EVALUATION

## 4.1 Experimental Setup

In this section, we discuss the experiments to evaluate our system for its robustness and effectiveness to inspect the intersectional effects caused by interaction across multiple sensitive attributes. We simulate users' interaction with our system using jupyter notebook.

## 4.2 Datasets

We evaluate our system on four datasets shown in Table 1. The first three of them are frequently used in the literature of fairness inside which different level of disparity has been observed between groups defined by the sensitive attributes that are also listed in the table.

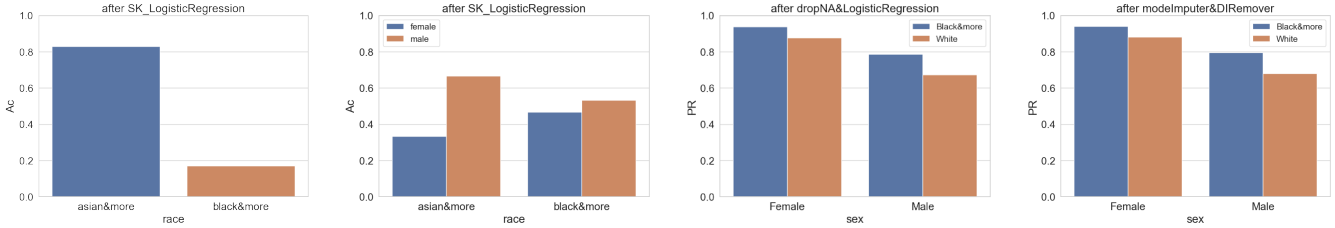
## 4.3 Results

***System Robustness.*** We evaluate our system on a new dataset: Law School Admissions for robustness. We observe that there are various abnormal values inside this dataset such as typos and abbreviations. For example, *race* column includes values like whjte, caucasion, caucasian, W, and C, which we can assume the indication of the correct value white. Since our system does not support the fine-grained clean of above abnormal values at this moment, we cleaned this dataset before feed it into our system. Data clean, especially clean for the above abnormal values, can be difficult in practice. As a part of future work, we plan to add a data clean operation into pre-processing stage to support common data clean operation like automatically replacing abnormal values. In order to apply fairness interventions, we categorize *race* into 2 groups: protected group including Black, Hispanic, and native American, and the other group including white and Asian.

In Figure 1a and 1b, we showcase the generated labels of Law School Admissions using "application status" as the target variable to be predicted. We observed a disparity between the acceptance rate of two gender groups while there are less female students accepted inside this data. And the disparity is not distributed similarly inside different racial groups. This skewed distribution of positive outcomes, i.e., being accepted, can lead to distinct performance of a model built on this dataset. Users are able to efficiently evaluate whether certain disparity is observed against a minority group inside the model predictions using our system. We built a Logistic Regression classifier to predict the target variable using all features that are related with the acceptance. However, even with hyper-parameter tuning, the learned model only have accuracy around 50%. The set of features inside this dataset is relatively less informative to build a good model to predict the target variable.

***Intersectional Effects.*** We evaluate intersectional effects through applying fairness interventions on one sensitive attribute and observe generated fairness labels for two sensitive attributes. We want to learn whether the applied fairness intervention impacts the sub-populations from another sensitive attribute in a different manner. For brevity, we only include results from Adult Income dataset in Figure 1c and 1d. We recommend readers to find all the experiment results in our GitHub repository.

For Adult Income, we train a Logistic Regression classifier using the same set of features as in *AI Fairness 360* but include another feature *workclass* with missing values. The performance label measured by selection rate is shown in Figure 1c. We then apply a fairness preprocessing intervention — *disparate impact remover* on gender using 1 as repair level to evaluate intersectional effects for different racial groups. We




**(a) Static label of Law School Admissions for race**

**(b) Static label of Law School Admissions for race and gender**

**(c) Performance label of Adult Income**

**(d) Performance label of Adult Income after applying Disparate Impact Remover**

**Figure 1: Fairness labels of Adult Income and Law School Admissions for `gender` and `race` as sensitive attributes. Applied pipeline for Adult Income includes removing rows with empty values, disparate impact remover (only for Figure 1d), train a Logistic Regression classifier to predict the target variable: annual income. Applied pipeline for Law School Admissions includes sampling 10,000 items, removing empty values, and train a Logistic Regression model to predict the target variable: application status.**

| dataset | size | target | 1st sensitive att | 2nd sensitive att | used in other papers |
|---|---|---|---|---|---|
| Adult Income | 32,561 | annual income | race | gender | Yes |
| German Credit | 1000 | credit worthiness | binned age | gender | Yes |
| COMPAS Score | 7,214 | whether recidivated | race | gender | Yes |
| Law School Admissions | 438,487 | application status | race | gender | No |

**Table 1: Datasets**

present the performance label after the fairness intervention in Figure 1d. We first observed that the fairness intervention does not remove all the difference between two gender groups for the input pipeline. And we found that it also slightly increase the selection rate for the privileged racial group, i.e., white inside male group in Figure 1d. Comparing to the results of applying the same fairness intervention on this dataset shown in AI Fairness 360, we notice that *disparate impact remover* can be sensitive to the input set of features and the pre-processed steps applied on it through experiments of various input pipeline.

## 5 DISCUSSION

In this paper, we implemented a system that support most of the operations from different stages in a ML pipeline and integrate the generation and visualization of fairness labels in our system.

In general, our project went well without major difficulties. Specifically, we spent most of time on designing and generating reasonable fairness labels for multiple sensitive attributes so that the labels provide meaningful yet interpretable information.

For system, the next step of our project focuses on integrating more modules at each stage of a ML pipeline and upgrading our system to support pipeline with flexible length as input. For fairness labels, the next step of our project focuses on informative representations and visualizations of linear combinations of more than two sensitive attributes.

## 6 DETAILED CONTRIBUTIONS

### 6.1 Student 1: Ke Yang

Student 1 is responsible for the implementation of the pipeline of our system and the supported modules inside each stage, the experiments with German Credit and Adult Income datasets.

### 6.2 Student 2: Tin Luu

Tin contributed to 1) the building and generating of static labels, 2) working with the label data structure to generate bar-plot visualizations for both static and performance labels, 3) cleaning and experimenting with the raw Law Schools data set for intersection effects.

### 6.3 Student 3: Hong Gong

Student 3 is contributed to 1) designing and implementing of static label and performance label 2) experiments with COMPAS Score dataset for intersection effects.

## REFERENCES


[1] Yeounoh Chung, Tim Kraska, Neoklis Polyzotis, Ki Hyun Tae, and Steven Euijong Whang. 2019. Slice Finder: Automated Data Slicing for Model Validation. In *35th IEEE International Conference on Data Engineering, ICDE 2019, Macao, China, April 8-11, 2019*. 1550–1553. https://doi.org/10.1109/ICDE.2019.00139

[2] Kimberle Crenshaw. 1989. Demarginalizing the intersection of race and sex: A black feminist critique of antidiscrimination doctrine, feminist theory and antiracist politics. *u. Chi. Legal f.* (1989), 139.

[3] Úrsula Hébert-Johnson, Michael P. Kim, Omer Reingold, and Guy N. Rothblum. 2018. Multicalibration: Calibration for the (Computationally-Identifiable) Masses. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. 1944–1953. http://proceedings.mlr.press/v80/hebert-johnson18a.html

[4] Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miroslav Dudík, and Hanna M. Wallach. 2019. Improving Fairness in Machine Learning Systems: What Do Industry Practitioners Need?. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI 2019, Glasgow, Scotland, UK, May 04-09, 2019*. 600. https://doi.org/10.1145/3290605.3300830

[5] Wendy Hulko. 2009. The time-and context-contingent nature of intersectionality and interlocking oppressions. *Affilia* 24, 1 (2009), 44–55.

[6] Michael J. Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. 2018. Preventing Fairness Gerrymandering: Auditing and Learning for Subgroup Fairness. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. 2569–2577. http://proceedings.mlr.press/v80/kearns18a.html

[7] Keith Kirkpatrick. 2017. It's not the algorithm, it's the data. *Commun. ACM* 60, 2 (2017), 21–23.

[8] Julia Stoyanovich, Bill Howe, Serge Abiteboul, Gerome Miklau, Arnaud Sahuguet, and Gerhard Weikum. 2017. Fides: Towards a Platform for Responsible Data Science. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management, Chicago, IL, USA, June 27-29, 2017*. 26:1–26:6. https://doi.org/10.1145/3085504.3085530

[9] Zhe Zhang and Daniel B. Neill. 2016. Identifying Significant Predictive Bias in Classifiers. *CoRR* abs/1611.08292 (2016). arXiv:1611.08292 http://arxiv.org/abs/1611.08292