NYU

DS-GA 3001.009
Responsible Data Science
Lab 2

Center for Data Science
Udita Gupta | Tandon School of Engineering

AI Fairness 360

What is AI Fairness 360?

- **AI Fairness 360 (AIF360) is a comprehensive open-source toolkit**
    - >30 metrics: to check for unwanted bias in datasets and machine learning models
    - 10 state-of-the-art algorithms: to mitigate such bias
- **Launched by IBM**
- **It’s python package includes**
    - Metrics for datasets and models to test for bias
    - Explanations of these metrics in TEXT and JSON
    - Algorithms to mitigate bias in datasets and models
    - Some standard example datasets

Fairness: Building and Deploying models

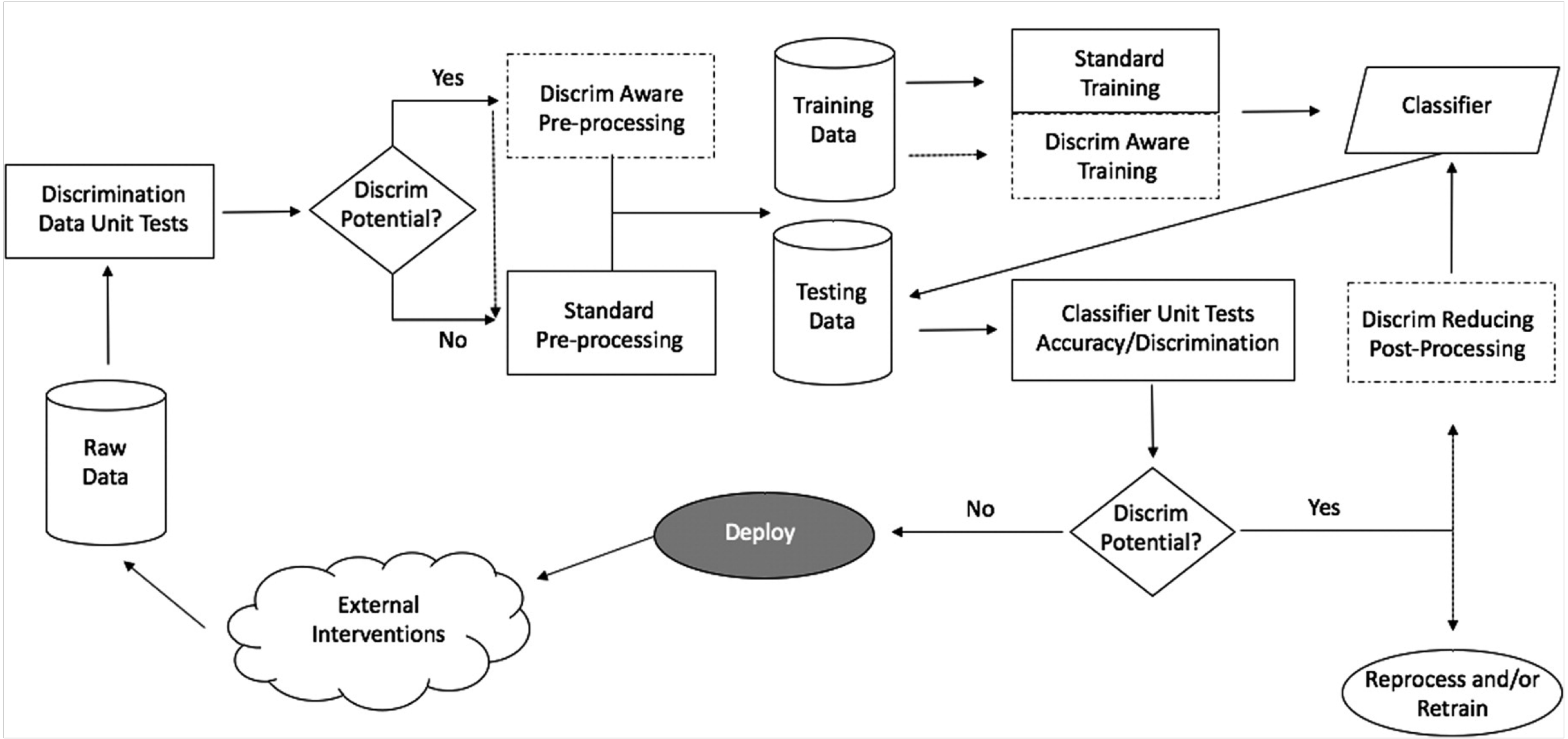

Fig 1. Discrimination Aware Classifier Build Process. [1]

AIF360: Metrics, Algorithms

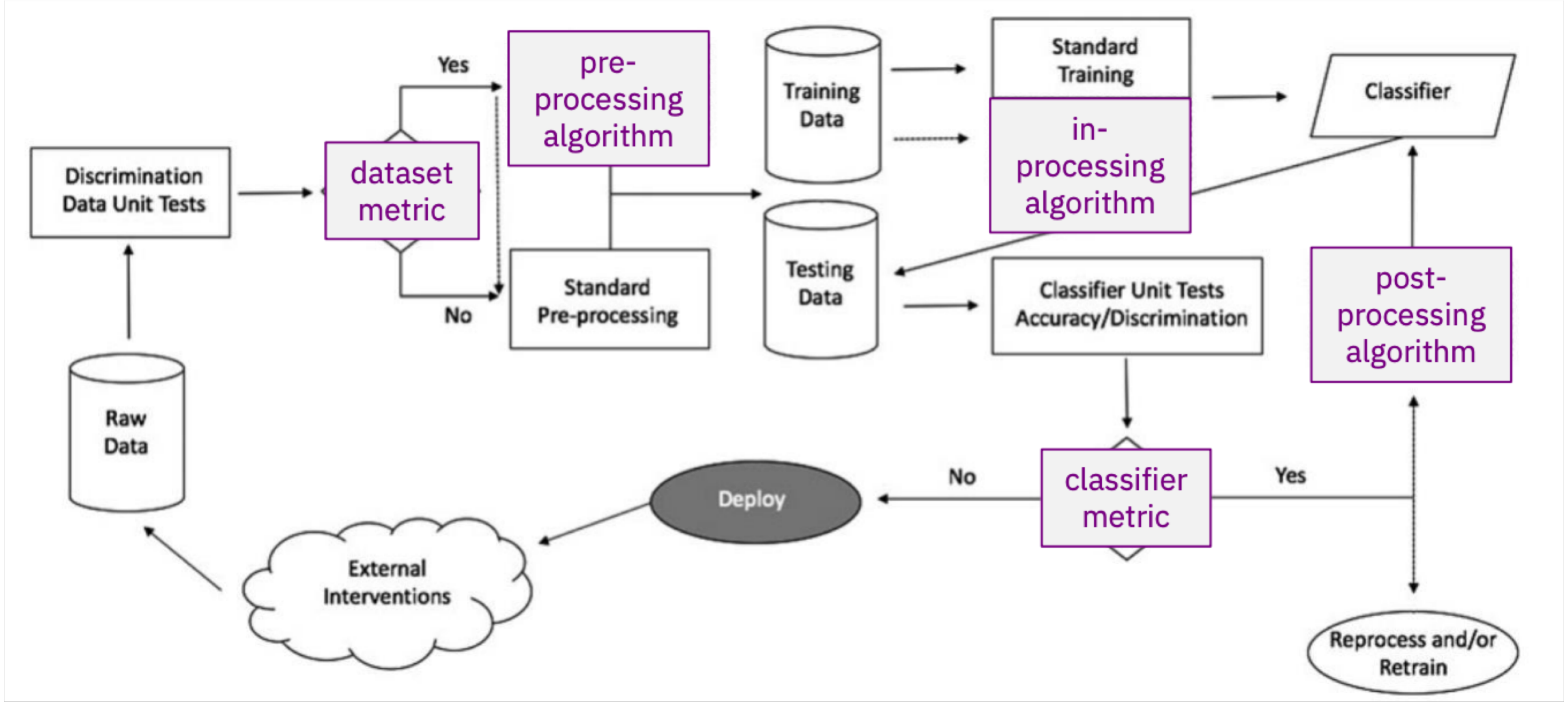

Fig 2. AIF360 Metrics and Algorithms. [2]

AIF360: Metrics, Algorithms, Explainers

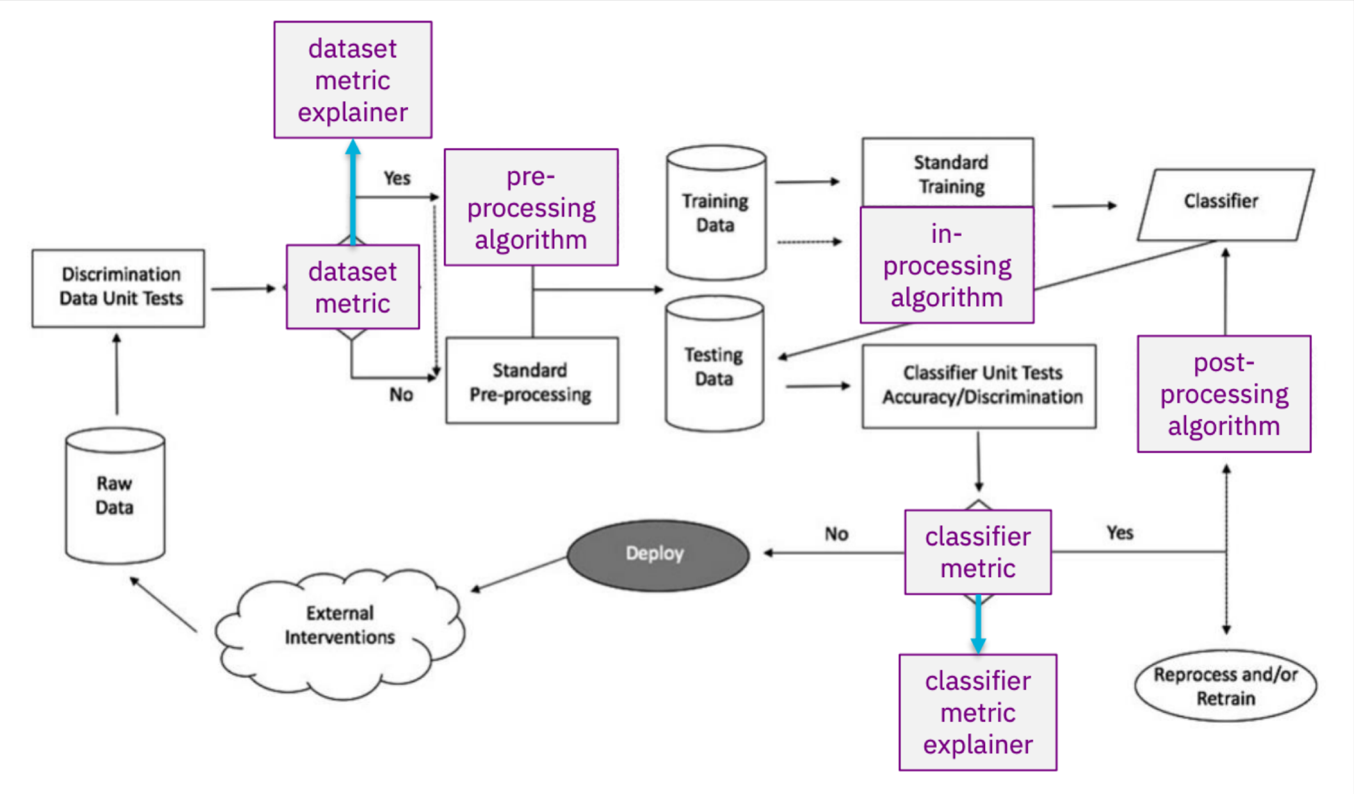

Fig 3. AIF360 Metrics and Algorithms. [2]

AIF360 Algorithms

- **Pre-processing**
  - Disparate Impact Remover
  - Learning Fair Representations
  - Optimized Preprocessing
  - Reweighing
- **In-processing**
  - Adversarial Debiasing
  - ART Classifier
  - Prejudice Remover
- **Post-processing**
  - Calibrated Equality of Odds
  - Equality of Odds
  - Reject Option Classification

AIF360 Metrics

- Statistical Parity Difference
- Equal Opportunity Difference
- Average Odds Difference
- Disparate Impact
- Mean Difference
- And many more..!

Reweighing

- **Pre-processing technique[3]**
- **Groundwork:**
  - Quality of the classifier is measured by its accuracy and discrimination; the more accurate, the better, and the less discriminatory, the better.
  - Let's restrict ourselves to one binary sensitive attribute S with domain {b,w} and a binary classification problem with target attribute Class with domain {−, +}.
  - "+" is the desirable class for the data subjects and the objects satisfying S = b and S = w represent, respectively, the deprived and the favored community.
  - The discrimination of a classifier C is defined as
    - $disc_{S=b} := P(C(X) = +|\ X(S) = w) - P(C(X) = +|\ X(S) = b)$, where X is a random unlabeled data object.
  - A discrimination larger than 0 reflects that a tuple for which S is w has a higher chance of being assigned the positive label by the classifier C than one where S is b.

Reweighing

- **The tuples in the training dataset are assigned weights.**
- **By carefully choosing the weights, the training dataset can be made discrimination-free w.r.t. S without having to change any of the labels.**
  - For example, objects with X(S) = b and X(Class) = + will get higher weights than objects with X(S) = b and X(Class) = − and objects with X(S) = w and X(Class) = + will get lower weights than objects with X(S) = w and X(Class) = −.

Reweighing - Idea

- **Idea behind weight calculation[3]:**
  - If the dataset D is unbiased, i.e., S and Class are statistically independent, the expected probability $P_{exp}(S = b \wedge Class = +)$ would be:
    - $P_{exp}(S = b \wedge Class = +) \coloneqq \frac{|\{X \in D \mid X(S)=b\}|}{|D|} \times \frac{|\{X \in D \mid X(Class) = +\}|}{|D|}$
  - In reality, however, the observed probability in D,
    - $P_{obs}(S = b \wedge Class = +) \coloneqq \frac{|\{X \in D \mid X(S)=b \wedge X(Class) = +\}|}{|D|}$ might be different.
  - If the expected probability is higher than the observed probability value, it shows the bias toward class − for those objects X with X(S) = b.
  - To compensate for the bias, we will assign lower weights to objects that have been deprived or favored.
  - Every object X will be assigned weight:
    - $W(X) \coloneqq \frac{P_{exp}(S = X(S) \wedge Class = X(Class))}{P_{obs}(S = X(S) \wedge Class = X(Class))}$
    - i.e., the weight of an object will be the expected probability to see an instance with its sensitive attribute value and class given independence, divided by its observed probability.

Reweighing - Algorithm

Algorithm 3: *Reweighing*

Input: $(D, S, Class)$
Output: Classifier learned on reweighed D

for $s \in \{b, w\}$ **do**
  for $c \in \{-, +\}$ **do**
    Let $W(s, c) := \dfrac{|\{X \in D \mid X(S) = s\}| \times |\{X \in D \mid X(Class) = c\}|}{|D| \times |\{X \in D \mid X(Class) = c \text{ and } X(S) = s\}|}$
  end for
end for
$D_W := \{\}$
for X in D **do**
  Add $(X, W(X(S), X(Class)))$ to D_W
end for
Train a classifier C on training set D_W, taking onto account the weights
return Classifier C

Fig 4. Reweighing Algorithm. [3]

Reweighing - Example

Sex	Ethnicity	Highest degree	Job type	Class
M	Native	H. school	Board	+
M	Native	Univ.	Board	+
M	Native	H. school	Board	+
M	Non-nat.	H. school	Healthcare	+
M	Non-nat.	Univ.	Healthcare	–
F	Non-nat.	Univ.	Education	–
F	Native	H. school	Education	–
F	Native	None	Healthcare	+
F	Non-nat.	Univ.	Education	–
F	Native	H. school	Board	+

Table 1. Example Dataset. [3]

Reweighing - Example

- **Consider the dataset in Table 1[3].**
  - We will calculate the weights for each data object (aka tuple) according to it's S and class values.
- **Let's calculate the weight for X(S) = f and X(Class) = +.**
  - We can see that 50% of the objects have *X(S)* = *f* and 60% of the objects have *X(Class)* = +
  - So, the expected probability:
    - $P_{exp}(Sex = f \wedge X(Class) = +) = 0.5 \times 0.6 = 30\%$
    - But it's actual probability is 20%
    - $W(X) = \frac{0.5 \times 0.6}{0.2} = 1.5$
- **Similarly, the weights of all other combinations are as follows:**

$$W(X) := \begin{cases} 1.5 & \text{if } X(Sex) = f \text{ and } X(Class) = + \\ 0.67 & \text{if } X(Sex) = f \text{ and } X(Class) = - \\ 0.75 & \text{if } X(Sex) = m \text{ and } X(Class) = + \\ 2 & \text{if } X(Sex) = m \text{ and } X(Class) = - \end{cases}$$

Reweighing - Example

Sex	Ethnicity	Highest degree	Job type	Cl.	Weight
M	Native	H. school	Board	+	0.75
M	Native	Univ.	Board	+	0.75
M	Native	H. school	Board	+	0.75
M	Non-nat.	H. school	Healthcare	+	0.75
M	Non-nat.	Univ.	Healthcare	–	2
F	Non-nat.	Univ.	Education	–	0.67
F	Native	H. school	Education	–	0.67
F	Native	None	Healthcare	+	1.5
F	Non-nat.	Univ.	Education	–	0.67
F	Native	H. school	Board	+	1.5

Table 2. Example Dataset with weights. [3]

Useful Links

- Git: https://github.com/IBM/AIF360
- Toolkit Homepage: https://aif360.mybluemix.net/
- Example Code Pattern: https://github.com/IBM/ensure-loan-fairness-aif360
- API documentation: https://aif360.readthedocs.io/en/latest/modules/algorithms.html
- AIF360 Overview Video: https://www.youtube.com/watch?v=X1NsrcaRQTE
- Reweighing paper: https://link.springer.com/content/pdf/10.1007%2Fs10115-011-0463-8.pdf

References

•[1] d'Alessandro B, O'Neil C, LaGatta T (2017) Conscientious classification: a data scientist's guide to discriminationaware classification. Big Data 5:2, 120–134, DOI: 10.1089/ big.2016.0048.

• [2] http://cognitive-science.info/wp-content/uploads/2018/09/CSIG_krv-aif360-2018-09-20-1.pdf

• [3] F. Kamiran and T. Calders, "Data Preprocessing Techniques for Classification without Discrimination," Knowledge and Information Systems, 2012.