ProxiMix: Enhancing Fairness with Proximity Samples in Subgroups

arxiv.org/abs/2410.01145

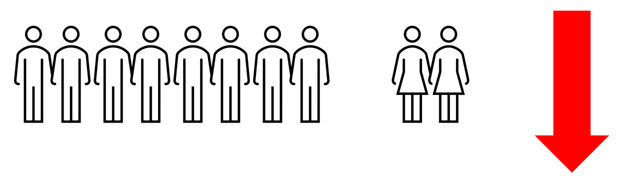
Jingyu Hu+, Jun Hong*, Mengnan Du♤, Weiru Liu+

+University of Bristol *University of the West of England ♤New Jersey Institute of Technology

TLDR: While linear Mixup (a data augmentation technique) helps with bias in machine learning, it can still retain biases present in dataset labels. This paper introduces ProxiMix, a new pre-processing method that combines existing Mixup with proximity relationships for fairer data augmentation.

1 Background & Problem Statement

What is bias? Here we refer to unfair treatment to a particular subgroup based on sensitive factors (e.g. gender, race, religion, etc.)
How bias arises? Many reasons, here we discuss **insufficient** data from the under-represented group.



Bias mitigation: Generate more samples for data **augmentation.**

Mixup: Blending pairs of data to create new synthetic data.

$$\textbf{Mixed } X = \lambda * X_1 + (1-\lambda) * X_2 \text{ ; } \textbf{Mixed } Y = \lambda * Y_1 + (1-\lambda) * Y_2$$

⚠ **But… sometimes generated data can even deepen bias**

Samples	Gender	Captial Gain	Occupation	Age	Income
M1	Male	8200	Officer	34	>50K
M2	Male	7800	Officer	35	>50K
F1	Female	8200	Officer	34	<=50K

Toy example: the female (F1) and two males (M1, M2) have similar feature profiles, but F1 is low-income (initial bias).

When using Mixup to generate new samples from F1 and M1:
-If the ratio $\boldsymbol{\lambda}$ favors F1: new **low-income female** samples.
-If the ratio $\boldsymbol{\lambda}$ favors M1: new **high-income male** samples.
Continued generation of such samples retain the initial bias, further reinforcing the model to associate high income with males and low income with females.

Our Solution: change the way of assigning Mixed Y-labels

2 Motivation & Methodology

Proposed ProxiMix: consider both Mixup labels and proximity samples labels, balancing the two with a certain degree $\boldsymbol{d}$.

Algorithm 1 ProxiMix Algorithm

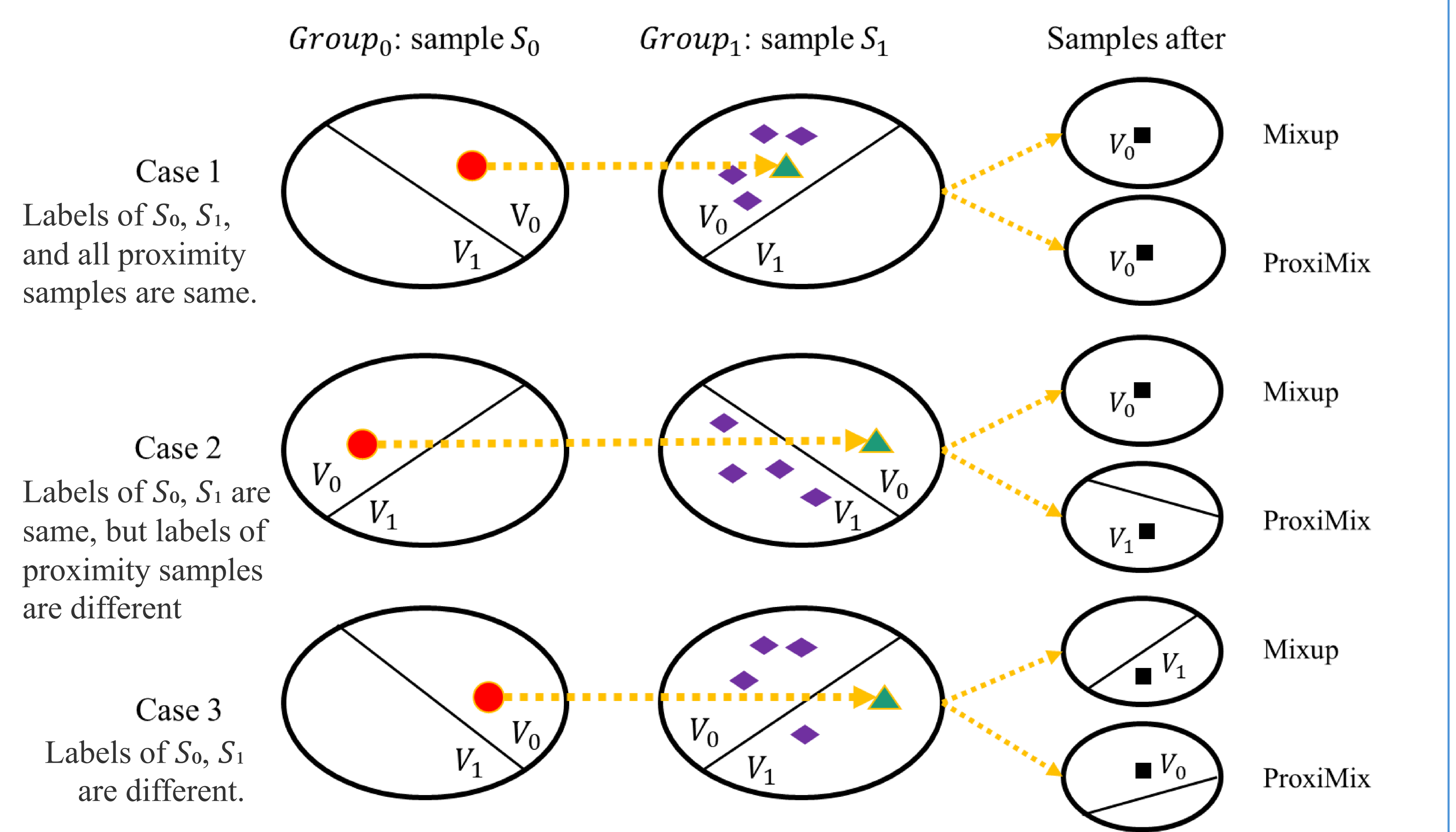
```
Input S_0(x_0, y_0, z_0) ~ D_train(Z = 0), S_1(x_1, y_1, z_1) ~ D_train(Z = 1)
procedure ProxiMix(S_0, S_1, D_train, d)
    procedure Proximity-Based-Mixed(S_0, S_1, D_train)
        ProxiSet = [] .
        P_dis = ||x_0 − x_1||
        for each sample S_i(z_i, y_i, z_i) in D_train(Z = 1) do
            P_cur = ||x_i − x_1||
            if P_cur ≤ P_dis then
                Add S_i to ProxiSet.
            end if
        end for
        NewSet = S_0 ∪ ProxiSet
        Y_Sim = Label_Counts(Y ∈ NewSet)/size(NewSet)
    end procedure
    procedure Lambda-Based-Mix(S_0, S_1)
        λ = Beta(α, α)
        Y_λ = λ ∗ y_0 + (1 − λ) ∗ y_1
    end procedure
    Ỹ = d ∗ Y_λ + (1 − d) ∗ Y_Sim, d ∈ [0, 1]
    Return Ỹ
end procedure
```

Proximity samples: samples that are within the calculated distance between two Mixup samples.

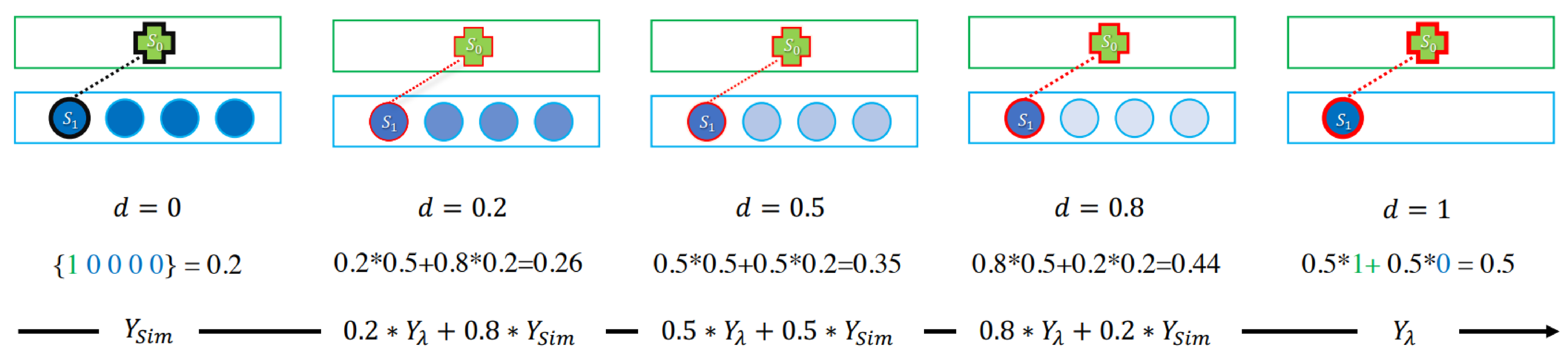
$\boldsymbol{Y_{Sim}}$: Proximity-aware
$\boldsymbol{Y_{\lambda}}$: Labels of Mixup samples

Degree $\boldsymbol{d}$: Balance Y_{Sim}, Y_λ
- $\boldsymbol{d = 0}$: Proximity labels only
- $\boldsymbol{d = 1}$: Mixup labels only
- $\boldsymbol{d \in (0,1)}$: Both proximity and Mixup labels

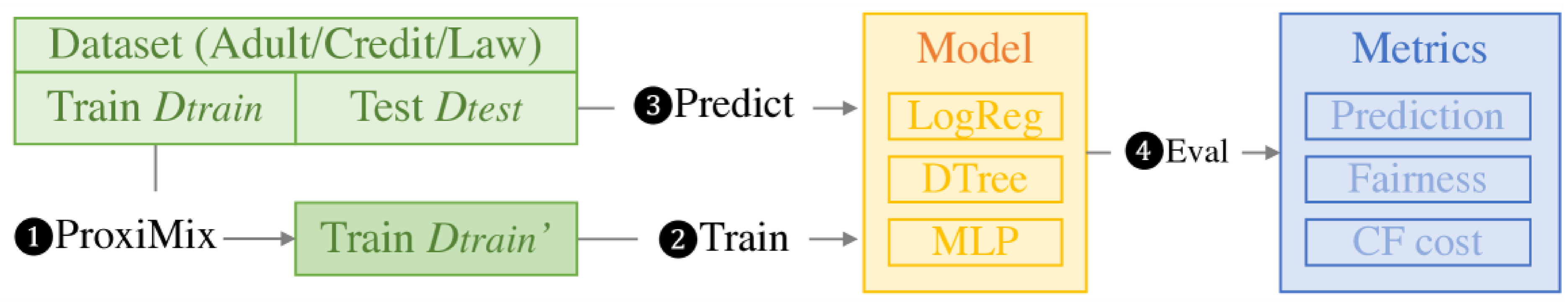
Comparison between ProxiMix and Mixup (λ = 0.5)



The Intuitive Demo of ProxiMix (Different d, λ = 0.5)



3 Experiments & Discussion



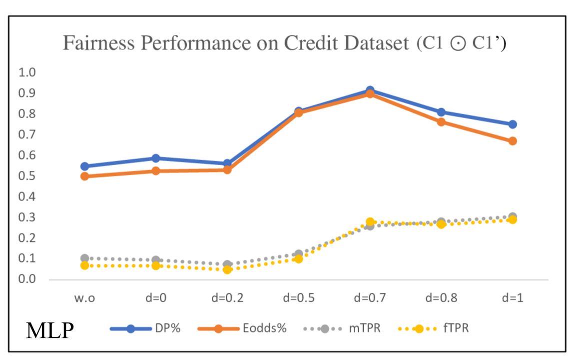
C_i: 1st Sample from C1/C2/C3/C4; ⊙: ProxiMix; C_j: 2nd Sample form C1'/C3'

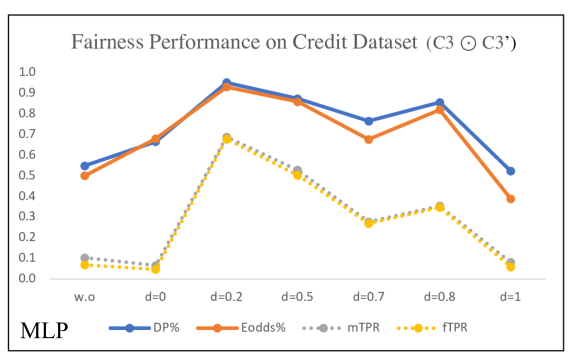
	C1 (z,y)	C2 (z,y)	C3 (z,y)	C4 (z,y)
Adult	female, low-income	female, high-income	male, low-income	male, high-income
Law	female, failed	female, passed	male, failed	male, passed
Credit	female, on-time	female, overdue	male, on-time	male, overdue
	C1'($\bar{y}$)	C1'($\bar{y}$)	C3'($\bar{y}$)	C3'($\bar{y}$)
Adult	male group	male group	female group	female group
Law	male group	male group	female group	female group
Credit	male group	male group	female group	female group

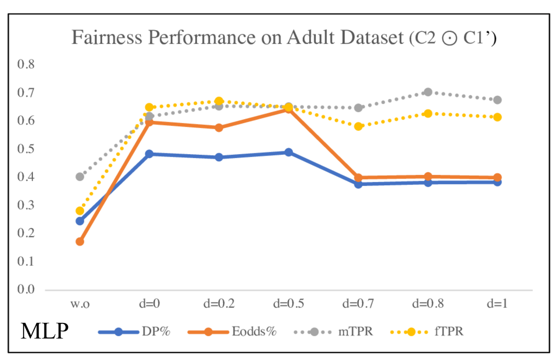
The overall workflow. There are many parameters for ProxiMix, here we fixed: Generated size, Proximity size.
Discussed: Sample combinations (which pairs to mix) $\boldsymbol{C_i \odot C_j}$, Balancing degree $\boldsymbol{d}$ (between Proximity-aware and Mixup)

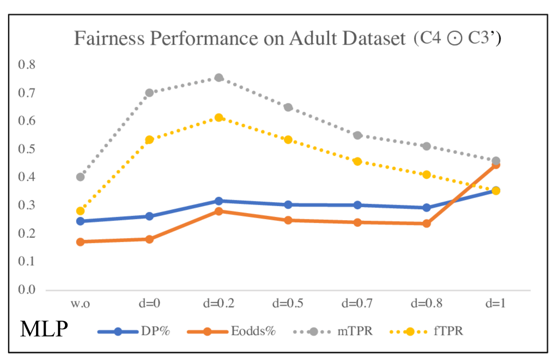
Dataset	Adult Income				Law School			
Model	LogReg		DT		LogReg		DT	
d=0.5	F1 Score	DP%	F1 Score	DP%	F1 Score	DP%	F1 Score	DP%
Baseline	0.7791	0.2892	0.7782	0.2847	0.6408	<u>0.9856</u>	0.6146	**0.9935**
$C1 \odot C1'$	0.7758	0.2439	0.7749	0.2792	0.6680	0.9261	0.6336	0.9831
$C2 \odot C1'$	0.7820	**0.4730**	0.7729	**0.3698**	0.6279	**0.9948**	0.6428	<u>0.9925</u>
$C3 \odot C3'$	0.7705	0.2625	0.7721	0.2971	0.6696	0.9619	0.6309	0.9837
$C4 \odot C3'$	0.7884	<u>0.2889</u>	0.7780	<u>0.2988</u>	0.6251	0.9840	0.6369	0.9921

- ProxiMix performs **better** on datasets where the **initial bias is obvious** (Adult, Credit), which aligns with our expectations. As ProxiMix is specifically designed to address the issue where directly applying Mixup can deepen the initial bias. Using Mixup directly is sufficient if the data is sufficiently unbiased/fair.









- It has trade-off between prediction performance and fairness performance with different $\boldsymbol{d}$. Though some $\boldsymbol{d}$ improve fairness (relative error between groups), their absolute prediction performance within groups decreases (fTPR, mTPR).

Summary We identified an intuitive research gap: using Mixup for data augmentation can potentially deepen biases. This paper presents a straightforward solution called ProxiMix, which uses proximity samples as references when assigning mixed labels. There is much room for future discussion on how different settings can benefit this gap, such as defining more tailored proximity samples and analyzing the influence of generated and proximity sizes.

Full Paper arxiv.org/abs/2410.01145

ECAI 50th ANNIVERSARY · AEQUITAS Fairness and Bias in AI · University of BRISTOL · UWE Bristol · NJIT New Jersey Institute of Technology