Socially Fair Pairwise Learning

Bokun Wang
Department of Computer Science & Engineering
Texas A&M University

CSCE 689 Final Presentation
Dec 1, 2023

Outline

1. **Background I: Socially Fair Machine Learning**
2. Background II: Pairwise Machine Learning
3. Problem Formulation
4. Algorithmic Design
5. Convergence Analysis

Social/Min-Max Fairness

2022

J. Abernethy, P. Awasthi, M. Kleindessner, J. Morgenstern, C. Russell, J. Zhang
Active sampling for min-max fairness
ICML 2022

2023

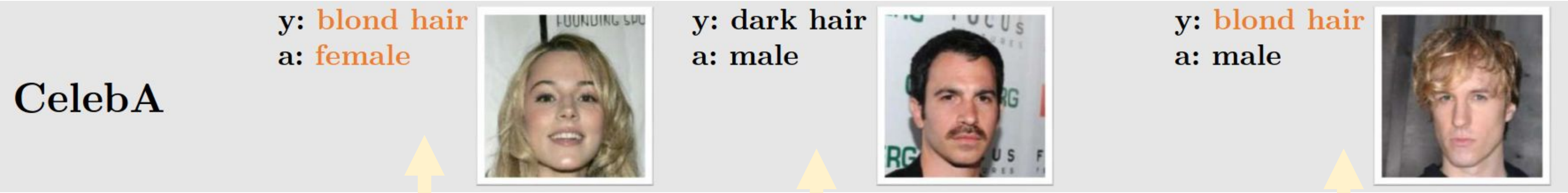
Anonymous authors
On Socially Fair Regression and Low-Rank Approximation
ICLR 2024 Submission (Under Review)

Goal: Optimize the performance of the algorithm across all sub-populations

Motivation

y: label; a: attribute

Rare group



Average accuracy: 94.6%

2019

S Sagawa, PW Koh, TB Hashimoto, P Liang

Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization

Motivation

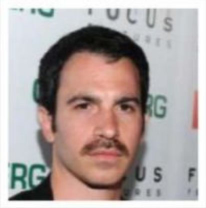
y: label; a: attribute

Rare group

CelebA

y: blond hair
a: female

y: dark hair
a: male



y: blond hair
a: male

Accuracy:
25.0%

2019

S Sagawa, PW Koh, TB Hashimoto, P Liang
Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization

Socially Fair Regression

Feature matrix (n x d) of the i-th group

Labels of the i-th group

$$\min_{\mathbf{x}\in\mathbb{R}^d} \max_{i\in[\ell]} \left\| \mathbf{A}^{(i)}\mathbf{x} - \mathbf{b}^{(i)} \right\|$$

Model

Groups

Socially Fair Regression

Feature matrix (n x d) of the i-th group

Labels of the i-th group

$$\min_{\mathbf{x} \in \mathbb{R}^d} \max_{i \in [\ell]} \left\| \mathbf{A}^{(i)} \mathbf{x} - \mathbf{b}^{(i)} \right\|$$

Model

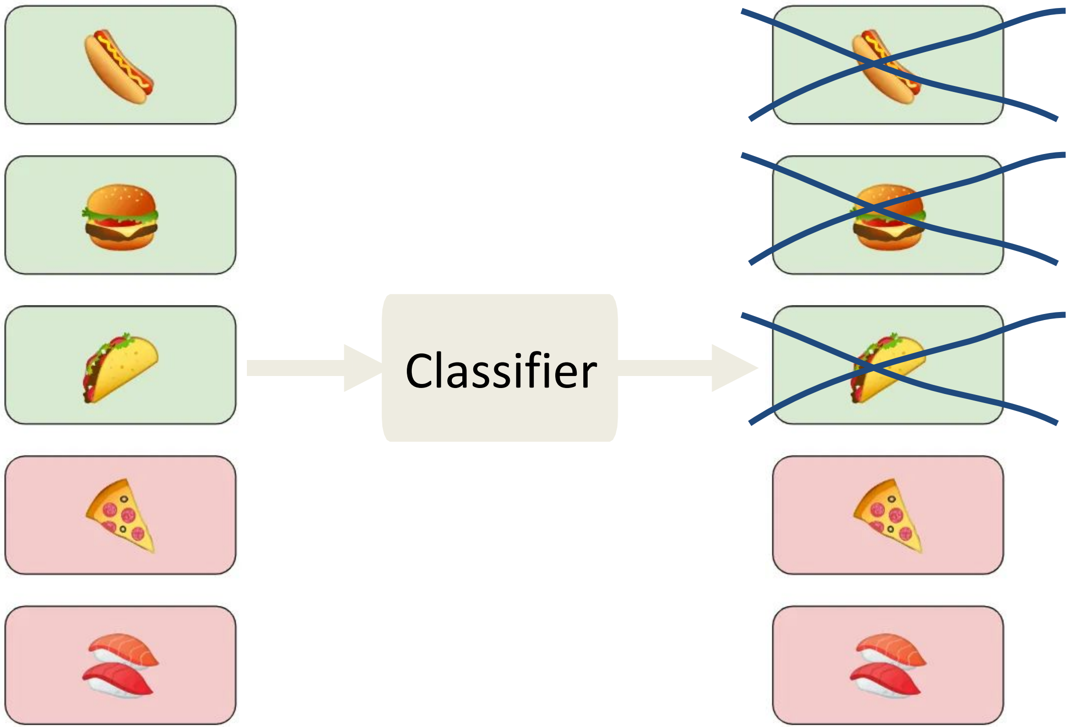
Groups

Minimizing the loss on the worst group

Outline

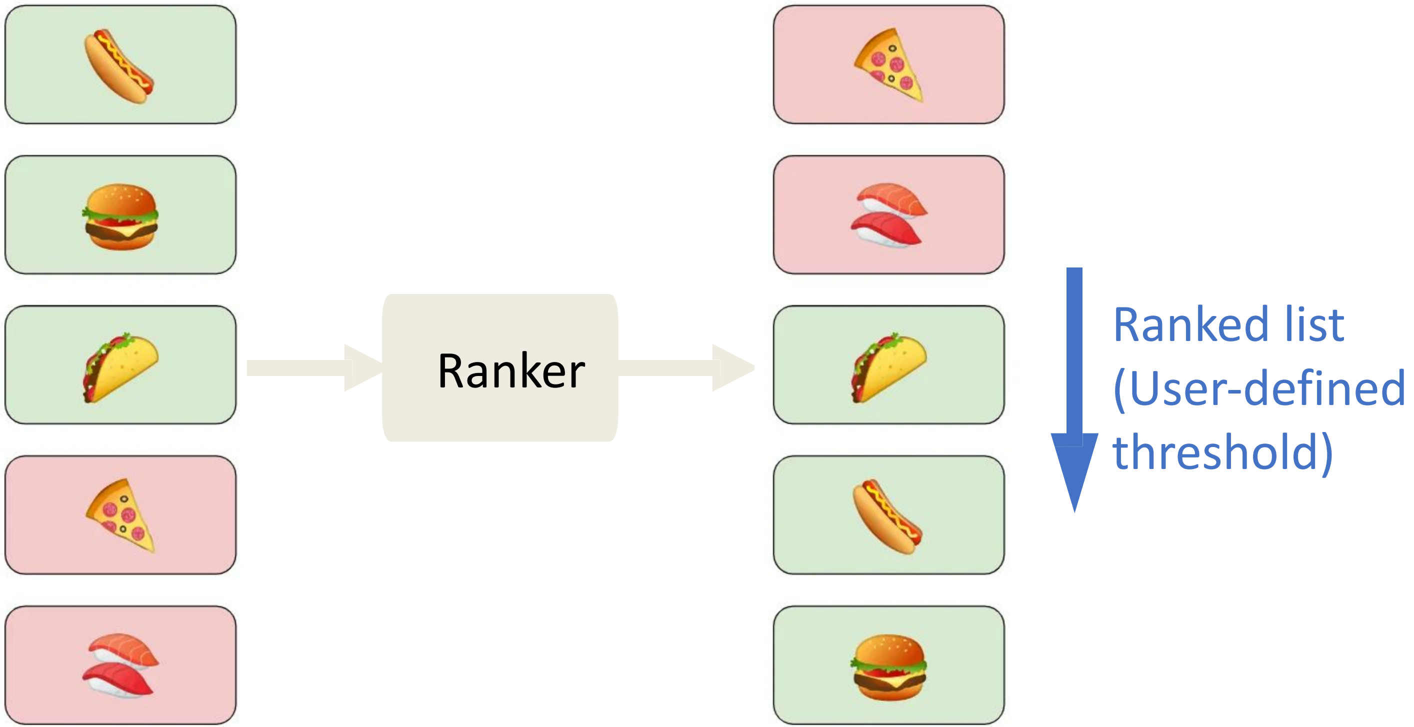
1. Background I: Socially Fair Machine Learning
2. **Background II: Pairwise Machine Learning**
3. Problem Formulation
4. Algorithmic Design
5. Convergence Analysis

Binary Classification v.s. Binary Ranking



Images from Amir Zaia

Binary Classification v.s. **Binary Ranking**



Binary Ranking Loss

Model

$$\ell(\theta; \mathbf{z}, \mathbf{z}') = \max(0, 1 - (h_\theta(\mathbf{x}) - h_\theta(\mathbf{x}'))) \mathbb{I}_{[y=1, y'=-1]}$$

Ranking scores

$$\mathbf{z} = (\mathbf{x}, y), \ \mathbf{z}' = (\mathbf{x}', y')$$

A pair of (+,-) data

Binary Ranking Loss

Positive data x has a higher ranking score than negative data x'

Model

$$\ell(\theta; \mathbf{z}, \mathbf{z}') = \max\left(0, 1 - \left(\underbrace{h_\theta(\mathbf{x}) - h_\theta(\mathbf{x}')}_{\text{Ranking scores}}\right)\right)\mathbb{I}_{[y=1, y'=-1]}$$

$$\mathbf{z} = (\mathbf{x}, y), \ \mathbf{z}' = (\mathbf{x}', y')$$

A pair of (+,-) data

Binary Ranking Loss

Positive data x has a higher ranking score than negative data x'

$$\ell(\theta; \mathbf{z}, \mathbf{z}') = \max\Big(0, 1 - \big(h_\theta(\mathbf{x}) - h_\theta(\mathbf{x}')\big)\Big)\mathbb{I}_{[y=1, y'=-1]}$$

$$\min_{\theta \in \Theta} \mathbb{E}_{\mathbf{z}, \mathbf{z}' \sim \mathbb{D}} \left[\ell(\theta; \mathbf{z}, \mathbf{z}')\right]$$

Outline

1. Background I: Socially Fair Machine Learning
2. Background II: Pairwise Machine Learning
3. **Problem Formulation**
4. Algorithmic Design
5. Convergence Analysis

Socially Fair Pairwise Learning

$$\ell(\theta; \mathbf{z}, \mathbf{z}') = \max\Big(0, 1 - \big(h_\theta(\mathbf{x}) - h_\theta(\mathbf{x}')\big)\Big)\mathbb{I}_{[y=1, y'=-1]}$$

$$\min_{\theta \in \Theta} \max_{i \in [g]} f(\theta; \boxed{\mathbb{D}_i}, \mathbb{D}_i), \qquad f(\theta; \mathbb{D}_i, \mathbb{D}_i) := \mathbb{E}_{\mathbf{z}, \mathbf{z}' \sim \mathbb{D}_i}[\ell(\theta; \mathbf{z}, \mathbf{z}')]$$

Data distribution of the i-th group

Setting

1. Online training data: one data point at a time

Setting

1. Online training data: one data point at a time
2. A relatively small offline validation set
    - Help our algorithm decide which group is the worst

Setting

1. Online training data: one data point at a time
2. A relatively small offline validation set
   - Help our algorithm decide which group is the worst
3. Goal: design an algorithm to make the following quantity as small as possible (“convergence”)

$$\mathbb{E}\left[\max_{i\in[g]} f(\bar{\theta}_T; \mathbb{D}_i, \mathbb{D}_i)\right] - \min_{\theta\in\Theta} \max_{i\in[g]} f(\theta; \mathbb{D}_i, \mathbb{D}_i)$$

Outline

1. Background I: Socially Fair Machine Learning
2. Background II: Pairwise Machine Learning
3. Problem Formulation
4. **Algorithmic Design**
5. Convergence Analysis

The Proposed Algorithm

1: **Input:** initial weight θ_0, validation sets $\{\hat{\mathcal{S}}_i\}_{i=1}^{g}$, $\{\hat{\mathcal{S}}_i'\}_{i=1}^{g}$, initial buffer $\mathcal{B}_0$

2: $\boxed{\mathcal{B} \leftarrow \mathcal{B}_0}$

A buffer to store b training data points for each group

The Proposed Algorithm

Input: initial weight θ_0, validation sets $\{\hat{\mathcal{S}}_i\}_{i=1}^{g}$, $\{\hat{\mathcal{S}}'_i\}_{i=1}^{g}$, initial buffer $\mathcal{B}_0$

$\mathcal{B} \leftarrow \mathcal{B}_0$

for $\boxed{t = 1, \ldots, T}$ **do**

Training iterations

  Compute $\boxed{i_t = \arg\max_{i \in [g]} \hat{f}(\theta_{t-1}; \hat{\mathcal{S}}_i, \hat{\mathcal{S}}'_i)}$

Use the validation sets to decide which group is the worst for the current model

The Proposed Algorithm

Input: initial weight θ_0, validation sets $\{\hat{\mathcal{S}}_i\}_{i=1}^{g}$, $\{\hat{\mathcal{S}}'_i\}_{i=1}^{g}$, initial buffer $\mathcal{B}_0$

$\mathcal{B} \leftarrow \mathcal{B}_0$

for $t = 1, \ldots, T$ **do**

  Compute $i_t = \arg\max_{i \in [g]} \hat{f}(\theta_{t-1}; \hat{\mathcal{S}}_i, \hat{\mathcal{S}}'_i)$

  $\boxed{\text{Sample } \mathbf{z}_t \sim \mathbb{D}_{i_t}}$ and retrieve $Z_t \leftarrow \mathcal{B}_{i_t}$.

Sample one data point
from the worst group

The Proposed Algorithm

Input: initial weight θ_0, validation sets $\{\hat{\mathcal{S}}_i\}_{i=1}^{g}$, $\{\hat{\mathcal{S}}_i'\}_{i=1}^{g}$, initial buffer $\mathcal{B}_0$

$\mathcal{B} \leftarrow \mathcal{B}_0$

for $t = 1, \ldots, T$ **do**

  Compute $i_t = \arg\max_{i \in [g]} \hat{f}(\theta_{t-1}; \hat{\mathcal{S}}_i, \hat{\mathcal{S}}_i')$

  Sample $\mathbf{z}_t \sim \mathbb{D}_{i_t}$ and $\boxed{\text{retrieve } Z_t \leftarrow \mathcal{B}_{i_t}}$

Retrieve the saved data for the worst group => compute the pairwise loss

The Proposed Algorithm

Input: initial weight θ_0, validation sets $\{\hat{\mathcal{S}}_i\}_{i=1}^{g}$, $\{\hat{\mathcal{S}}'_i\}_{i=1}^{g}$, initial buffer $\mathcal{B}_0$
$\mathcal{B} \leftarrow \mathcal{B}_0$
for $t = 1, \ldots, T$ **do**
    Compute $i_t = \arg\max_{i \in [g]} \hat{f}(\theta_{t-1}; \hat{\mathcal{S}}_i, \hat{\mathcal{S}}'_i)$
    Sample $\mathbf{z}_t \sim \mathbb{D}_{i_t}$ and retrieve $Z_t \leftarrow \mathcal{B}_{i_t}$.
    Compute $\boxed{\nabla_t = \frac{1}{b} \sum_{\mathbf{z}' \in Z_t} \nabla \ell(\theta_{t-1}; \mathbf{z}_t, \mathbf{z}')}$
    Update $\boxed{\theta_t = \Pi_\Theta[\theta_{t-1} - \eta \nabla_t]}$

Compute the gradient
and do (projected) SGD step

The Proposed Algorithm

Input: initial weight θ_0, validation sets $\{\hat{\mathcal{S}}_i\}_{i=1}^{g}$, $\{\hat{\mathcal{S}}'_i\}_{i=1}^{g}$, initial buffer $\mathcal{B}_0$

$\mathcal{B} \leftarrow \mathcal{B}_0$

for $t = 1, \ldots, T$ **do**

    Compute $i_t = \arg\max_{i \in [g]} \hat{f}(\theta_{t-1}; \hat{\mathcal{S}}_i, \hat{\mathcal{S}}'_i)$

    Sample $\mathbf{z}_t \sim \mathbb{D}_{i_t}$ and retrieve $Z_t \leftarrow \mathcal{B}_{i_t}$.

    Compute $\nabla_t = \frac{1}{b} \sum_{\mathbf{z}' \in Z_t} \nabla \ell(\theta_{t-1}; \mathbf{z}_t, \mathbf{z}')$

    Update $\theta_t = \Pi_{\Theta}[\theta_{t-1} - \eta \nabla_t]$

    Update $\boxed{\mathcal{B} \leftarrow \mathrm{FIFO}(\mathcal{B}, i_t, \mathbf{z}_t)}$

Update the buffer with
the fresh training data

Outline

1. Background I: Socially Fair Machine Learning
2. Background II: Pairwise Machine Learning
3. Problem Formulation
4. Algorithmic Design
5. **Convergence Analysis**

Convergence Results

$$\min_{\theta\in\Theta}\max_{i\in[g]} f(\theta;\mathbb{D}_i,\mathbb{D}_i)$$

$$f(\theta;\mathbb{D}_i,\mathbb{D}_i) := \mathbb{E}_{\mathbf{z},\mathbf{z}'\sim\mathbb{D}_i}[\ell(\theta;\mathbf{z},\mathbf{z}')]$$

(Theorem)

Suppose that the loss function ℓ is convex w.r.t. θ, and the domain Θ is convex and compact. After T iterations, the proposed algorithm with buffer size b leads to

$$\mathbb{E}\left[\max_{i\in[g]} f(\bar{\theta}_T;\mathbb{D}_i,\mathbb{D}_i)\right] \le \min_{\theta\in\Theta}\max_{i\in[g]} f(\theta;\mathbb{D}_i,\mathbb{D}_i) + O\left(\frac{1}{\sqrt{T-1}} + \sqrt{\frac{2\log(1/\delta)}{T-1}} + \frac{1}{\sqrt{b}}\right) \qquad \text{w.p. } 1-4\delta.$$

$$+\frac{2c}{T-1}\sum_{t=2}^{T}\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon',\mathcal{G},L_1(\mathbb{D}))/\delta)}{2m_{i_t}}}\right] + 2c\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon',\mathcal{G},L_1(\mathbb{D}))/\delta)}{2\min_i m_i}}\right]$$

Convergence Results

w.p. $1 - 4\delta$.

T: number of iterations

$$O\left(\boxed{\frac{1}{\sqrt{T-1}} + \sqrt{\frac{2\log(1/\delta)}{T-1}}} + \frac{1}{\sqrt{b}}\right)$$

$$+ \frac{2c}{T-1}\sum_{t=2}^{T}\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon', \mathcal{G}, L_1(\mathbb{D}))/\delta)}{2m_{i_t}}}\right] + 2c\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon', \mathcal{G}, L_1(\mathbb{D}))/\delta)}{2\min_i m_i}}\right]$$

Convergence Results

w.p. $1 - 4\delta$.

b: buffer size

$$O\left(\frac{1}{\sqrt{T-1}} + \sqrt{\frac{2\log(1/\delta)}{T-1}} + \boxed{\frac{1}{\sqrt{b}}}\right)$$

$$+ \frac{2c}{T-1}\sum_{t=2}^{T}\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon', \mathcal{G}, L_1(\mathbb{D}))/\delta)}{2m_{i_t}}}\right] + 2c\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon', \mathcal{G}, L_1(\mathbb{D}))/\delta)}{2\min_i m_i}}\right]$$

Convergence Results

w.p. $1 - 4\delta$.

$$O\left(\frac{1}{\sqrt{T-1}} + \sqrt{\frac{2\log(1/\delta)}{T-1}} + \frac{1}{\sqrt{b}}\right)$$

$$+\frac{2c}{T-1}\sum_{t=2}^{T}\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon', \mathcal{G}, L_1(\mathbb{D}))/\delta)}{2\boxed{m_{i_t}}}}\right] + 2c\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon', \mathcal{G}, L_1(\mathbb{D}))/\delta)}{2\min_i m_i}}\right]$$

m_i: size of validation set of the i-th group

Convergence Results

w.p. $1 - 4\delta.$

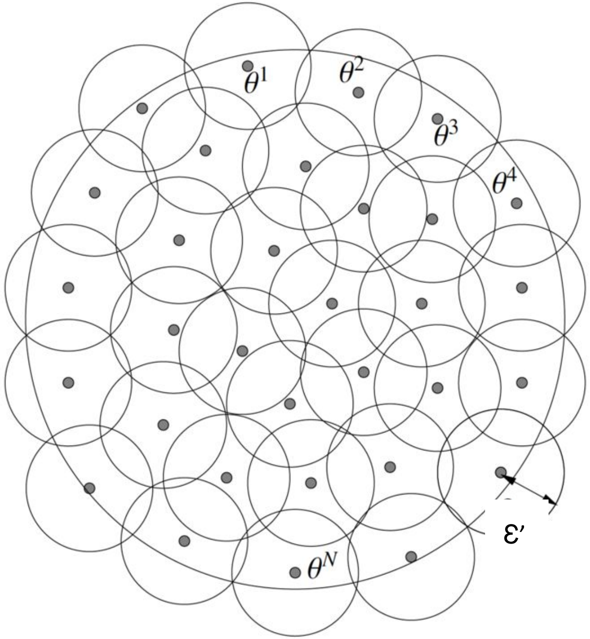
$$O\left(\frac{1}{\sqrt{T-1}} + \sqrt{\frac{2\log(1/\delta)}{T-1}} + \frac{1}{\sqrt{b}}\right)$$

$$+\frac{2c}{T-1}\sum_{t=2}^{T}\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon',\mathcal{G},L_1(\mathbb{D}))/\delta)}{2\boxed{m_{i_t}}}}\right] + 2c\inf_{\epsilon'>0}\left[\epsilon' + \sqrt{\frac{\ln(N_{LB}(\epsilon',\mathcal{G},L_1(\mathbb{D}))/\delta)}{2\min_i m_i}}\right]$$

m_i: size of validation set of the i-th group

Convergence Results

$$+\frac{2c}{T-1}\sum_{t=2}^{T}\inf_{\epsilon'>0}\left[\epsilon'+\sqrt{\frac{\ln(\boxed{N_{LB}(\epsilon',\mathcal{G},L_1(\mathbb{D}))}/\delta)}{2m_{i_t}}}\right]+2c\inf_{\epsilon'>0}\left[\epsilon'+\sqrt{\frac{\ln(N_{LB}(\epsilon',\mathcal{G},L_1(\mathbb{D}))/\delta)}{2\min_i m_i}}\right]$$



Covering number