Learning Fair Representations [2013]

by Richard Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, Cynthia Dwork

University of Toronto

2019/11/5

Presenter: Zeou Hu (U Waterloo)

UNIVERSITY OF WATERLOO

Overview

- Previous work
- This paper: the LFR Model
- Experiments
- Follow-ups
- Some thoughts and conclusions

Previous Work: Fairness Through Awareness [2012]

Fairness Through Awareness (Dwork, Zemel et al.) proposed a framework that:

- Individual fairness
  “Similar individuals are treated similarly”
- Group fairness
  “Disparate Impact Parity”
- Optimization problem
- Probabilistic mapping

However……

Previous Work: Fairness Through Awareness [2012]

Two obstacles:

1. A distance/similarity metric is assumed to be given

This is problematic because: a good distance metric that defines similarity between individuals is important for 'Individual Fairness', but is challenging to find

2. Cannot generalize

It only works for the given data set, doesn't know what to do with future unseen data

UNIVERSITY OF WATERLOO

This paper: Learning Fair Representations (LFR model)

- Individual fairness
  "Similar individuals are treated similarly"

- Group fairness
  "Disparate Impact Parity"

- Optimization problem

- Probabilistic mapping

- Learn a (restricted form of) distance metric

- Develops a learning approach that can generalize to unseen data

The LFR model in a nutshell: One sentence

"We formulate fairness as an optimization problem of finding an intermediate representation of the data that best encodes the data (i.e., preserving as much information about the individual's attributes as possible), while simultaneously obfuscates aspects of it, removing any information about membership with respect to the protected subgroup."

The LFR model in a nutshell: Two competing goals

I. the intermediate representation should encode the data as well as possible

Preserve utility

II. the encoded representation is sanitized in the sense that it should be blind to whether or not the individual is from the protected group

Remove sensitive information

UNIVERSITY OF WATERLOO

the LFR model: some notations

"The main idea in our model is to map each individual, represented as a data point in a given input space, to a probability distribution in a new representation space."

- Original data point $\mathbf{x} \in \mathcal{X}$, for some Euclidean space $\mathcal{X}$
- the representation space $\mathcal{Z}$ is a space of **discrete distributions** over finite prototypes $\mathbf{v}_k \in \mathcal{X}$
- each individual $\mathbf{x}$ is mapped to a distribution $\mathbf{z}$, where $\mathbf{z} \in \mathcal{Z} \subset \mathcal{P}(\mathcal{X})$
- Z is a multinomial random variable, where each of the K values represents one of the 'prototypes'. Associated with each prototype is a vector $\mathbf{v}_k$ that lies in the same space as $\mathbf{x}$

the LFR model: some MORE notations (optional)

- X denotes the entire data set, X_0 denotes the training set.
- S is the sensitive attribute, i.e. a binary random variable representing the membership of sensitive groups. By convention $S = \{0, 1\}$.
- $X^+ \subset X, X_0^+ \subset X_0$ denote subset of individuals that are members of sensitive group 1 (i.e. $S = 1$). Similarly we can define X^- and X_0^-.
- Y be the target random variable that we want to predict. For example, $f : X \to Y$ is the desired classification function.
- d is a distance function on $\mathcal{X}$, a common choice is the Euclidean distance: $d(\mathbf{x}_n, \mathbf{v}_k) = \|\mathbf{x}_n - \mathbf{v}_k\|_2$

the LFR model: probabilistic mapping

Recall: "Each data point in the input space is mapped to a probability distribution in a new representation space."

How?

University of Waterloo

the LFR model: probabilistic mapping

Recall: “Each data point in the input space is mapped to a probability distribution in a new representation space.”

Given the definitions of the prototypes as points in the input space, a set of prototypes induces a natural probabilistic mapping from X to Z via the softmax:

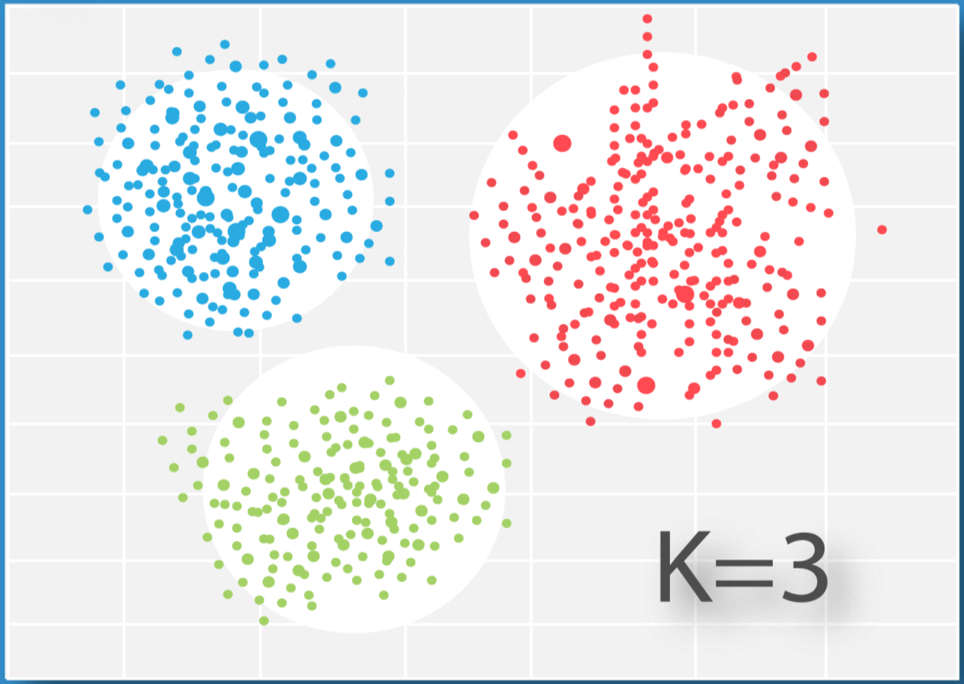
$$P(Z = k|\mathbf{x}) = \exp(-d(\mathbf{x}, \mathbf{v}_k)) / \sum_{j=1}^{K} \exp(-d(\mathbf{x}, \mathbf{v}_j)) \quad (2)$$

Actually, it’s called ‘soft-min’

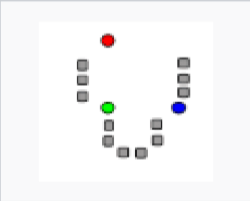
Probabilistic mapping: A clustering perspective

K-Mean clustering

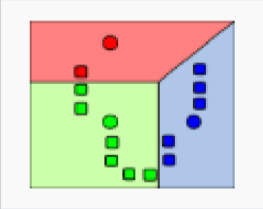
thatware.co



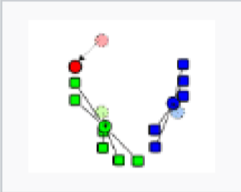
k-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.



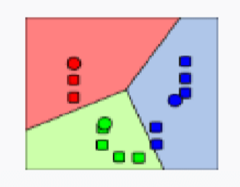
1. *k* initial "means" (in this case *k*=3) are randomly generated within the data domain (shown in color).



2. *k* clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.

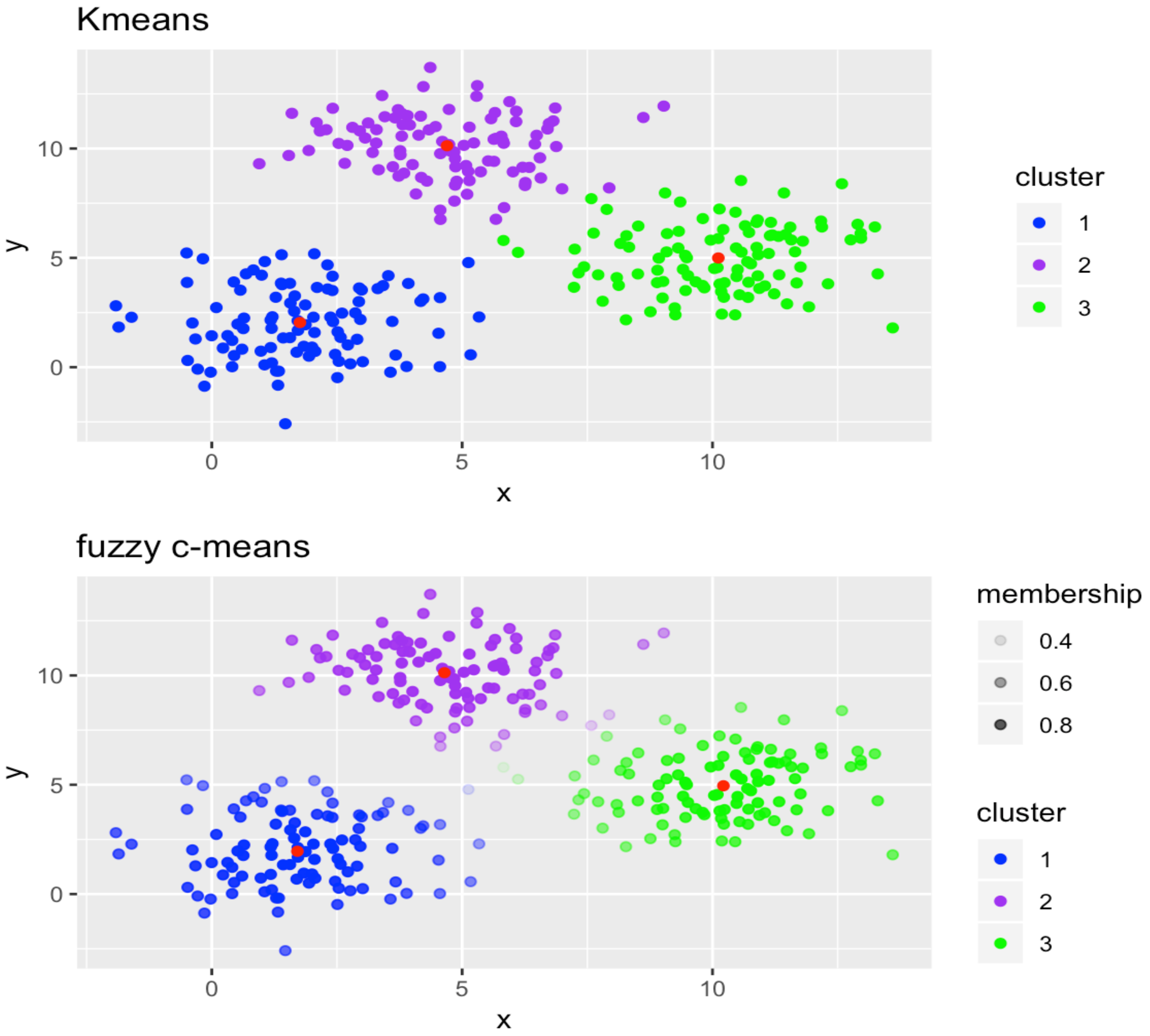


3. The centroid of each of the *k* clusters becomes the new mean.



4. Steps 2 and 3 are repeated until convergence has been reached.

Soft k-means


Kmeans
y
x
10
5
0
0
5
10
cluster
1
2
3
fuzzy c-means
y
x
10
5
0
0
5
10
membership
0.4
0.6
0.8
cluster
1
2
3

the LFR model: Objective function

The objective function consists of 3 terms:

1. Fairness term (group fairness)

2. Reconstruction term

3. Utility term

Objective function: Fairness term

In Zemel's paper, loss function for group fairness is defined as

$$L_z = \sum_{k=1}^{K} \left| M_k^+ - M_k^- \right|$$

where $M_{n,k} = P\left(Z = k | \mathbf{x}_n\right)$ and

$$M_k^+ = \mathbb{E}_{\mathbf{x} \in X^+} P(Z = k | \mathbf{x}) = \frac{1}{|X_0^+|} \sum_{n \in X_0^+} M_{n,k}$$

Each cluster should contain roughly balanced "mass" from the protected group and the unprotected group

Objective function: Reconstruction term

The second term constrains the mapping to Z to be a good description of X. We quantify the amount of information lost in the new representation using a simple squared-error measure:

$$L_x = \sum_{n=1}^{N} (\mathbf{x}_n - \hat{\mathbf{x}}_n)^2 \tag{8}$$

where $\hat{\mathbf{x}}_n$ are the reconstructions of $\mathbf{x}_n$ from Z:

$$\hat{\mathbf{x}}_n = \sum_{k=1}^{K} M_{n,k} \mathbf{v}_k \tag{9}$$

The learned representation should "resemble" the original data as good as possible

Objective function: Utility term

The final term requires that the prediction of y is as accurate as possible:

$$L_y = \sum_{n=1}^{N} -y_n \log \hat{y}_n - (1 - y_n) \log(1 - \hat{y}_n) \qquad (10)$$

Here $\hat{y}_n$ is the prediction for y_n, based on marginalizing over each prototype's prediction for Y, weighted by their respective probabilities $P(Z = k|\mathbf{x}_n)$:

$$\hat{y}_n = \sum_{k=1}^{K} M_{n,k} w_k \qquad (11)$$

We constrain the w_k values to be between 0 and 1.

The learned representation should still predict target variable quite well

UNIVERSITY OF WATERLOO

Objective function: putting all together

Given this setup, the learning system minimizes the following objective:

$$L = A_z \cdot L_z + A_x \cdot L_x + A_y \cdot L_y \qquad (4)$$

where A_x, A_y, A_z are hyper-parameters governing the trade-off between the system desiderata.

- Learnable parameters are: prototype locations $\{\mathbf{v}_k\}$ and parameters $\{w_k\}$, and α_i (will mention later)
- # of prototypes K is a hyper-parameter, in supplementary materials, they vary K ={10,20,30}, and observed that bigger K will result in better accuracy while worse fairness
- The objective function is optimized using L-BFGS

the LFR model: Learning distance metric

In order to allow different input features to have different levels of impact, we introduce individual weight parameters for each feature dimension, α_i, which act as inverse precision values in the distance function:

$$d(\mathbf{x}_n, \mathbf{v}_k, \alpha) = \sum_{i=1}^{D} \alpha_i (x_{ni} - v_{ki})^2 \qquad (12)$$

More flexible than Euclidean distance

the LFR model: what is the fairness definition?

The fairness definition used in the objective function is kind of strange,
but it is indeed a variant of Statistical Parity (aka Disparate Impact Parity)

The key property is that if the parity constraint is met, then the two groups are treated fairly with respect to the classification decisions:

$$\frac{1}{|X_0^+|}\sum_{n\in X_0^+} M_{n,k} = \frac{1}{|X_0^-|}\sum_{n\in X_0^-} M_{n,k} \Rightarrow$$

$$\frac{1}{|X_0^+|}\sum_{n\in X_0^+} M_n\mathbf{w} = \frac{1}{|X_0^-|}\sum_{n\in X_0^-} M_n\mathbf{w} \Rightarrow$$

$$\frac{1}{|X_0^+|}\sum_{n\in X_0^+} y_n^+ = \frac{1}{|X_0^-|}\sum_{n\in X_0^-} y_n^- .$$

This property follows from the linear classification approach.

Experiments

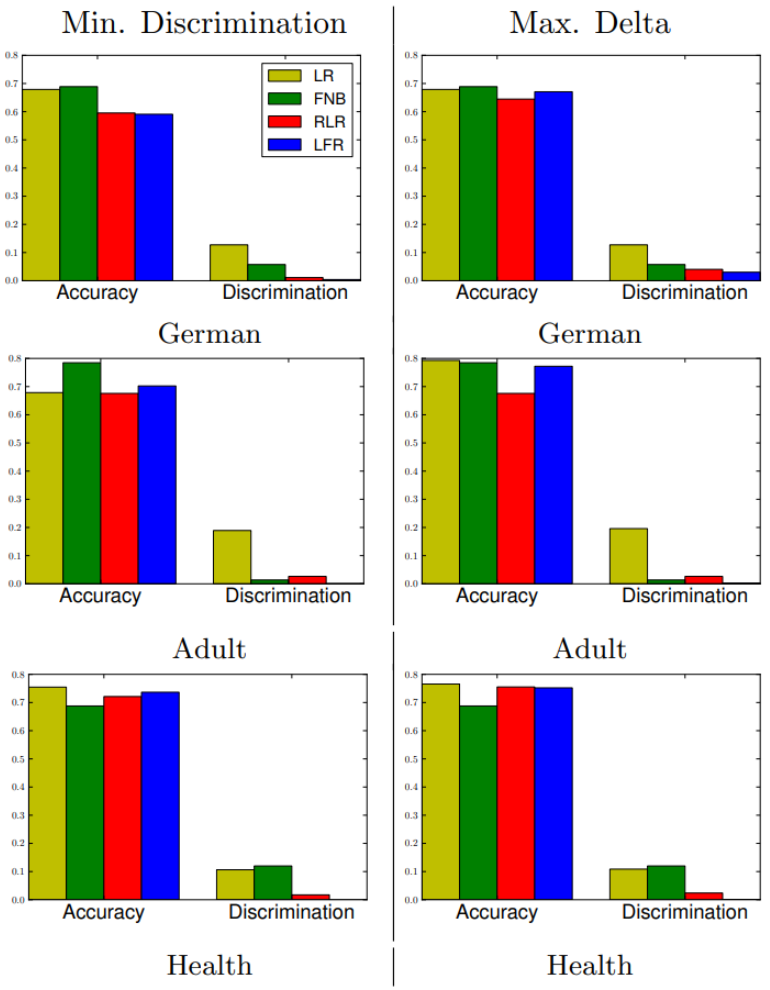


Figure 1. Results on test sets for the three datasets (German, Adult, and Health), for two different model selection criteria: minimizing discrimination and maximizing the difference between accuracy and discrimination.

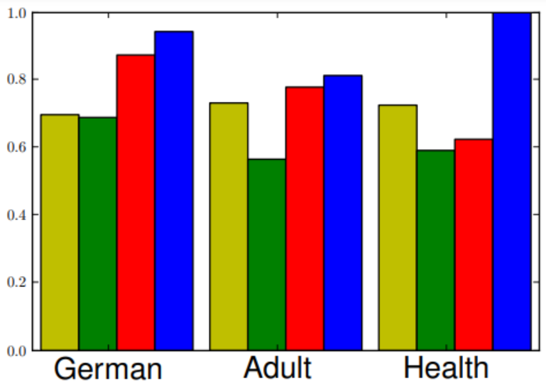


Figure 2. Individual fairness: The plot shows the consistency of each model's classification decisions, based on the yNN measure. Legend as in Figure 1.

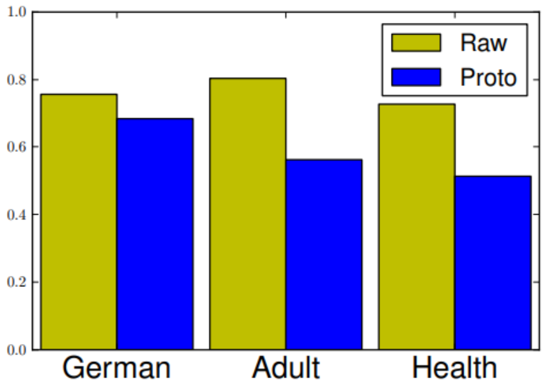


Figure 3. The plot shows the accuracy of predicting the sensitive variable ($sAcc$) for the different datasets. Raw involves predictions directly from all input dimensions except for S, while Proto involves predictions from the learned fair representations.

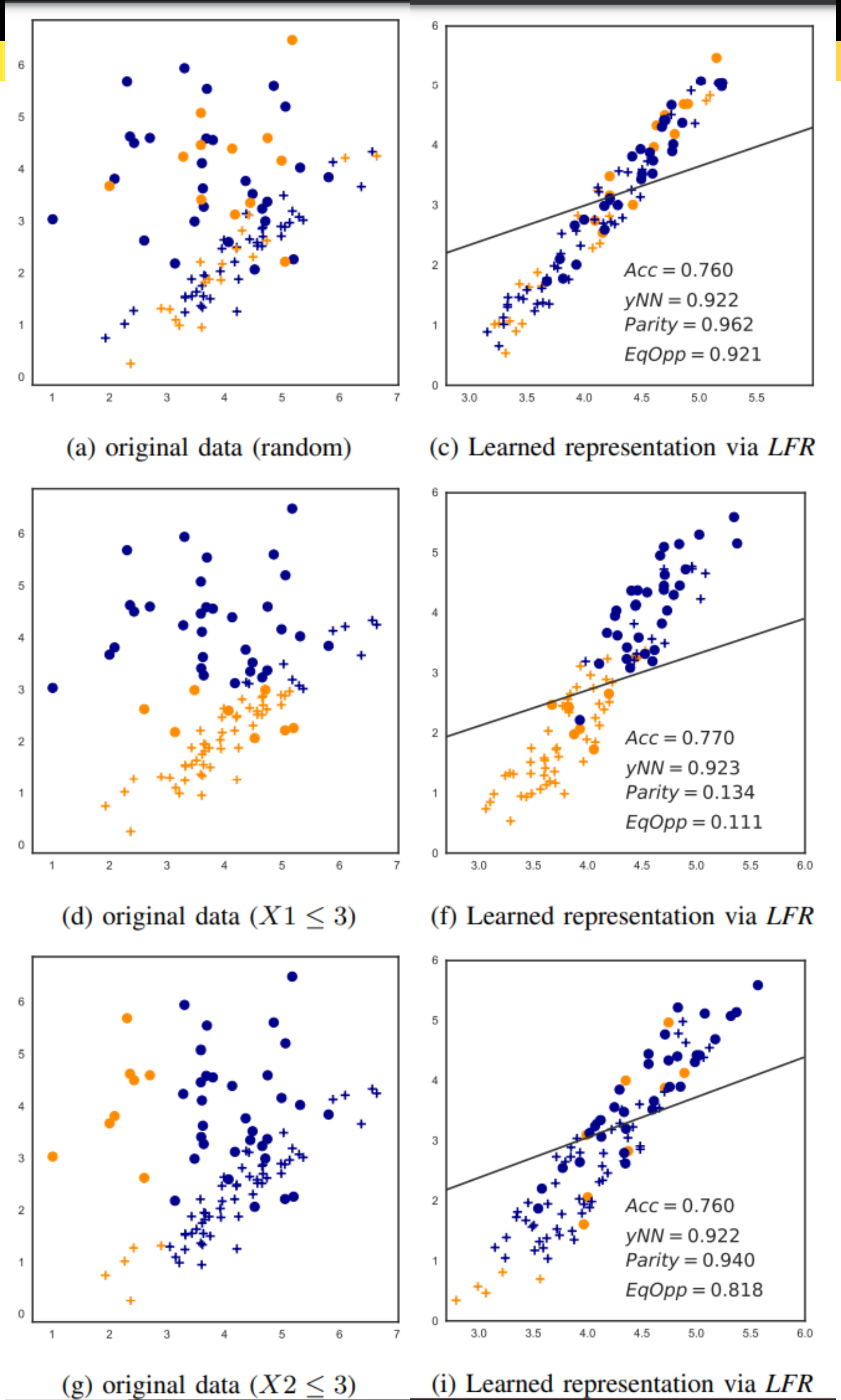
It works!

UNIVERSITY OF WATERLOO

Experiments

Figure from:

[iFair: Learning Individually Fair Data Representations for Algorithmic Decision Making]



UNIVERSITY OF WATERLOO

Follow-ups

There are a bunch of follow-up work on learning fair representation:

- Explicitly deals with Individual Fairness [P Lahoti et al. 2018]
- Use neural networks (MLP,VAE etc.) to learn fair representation (the most common approach right now) [E Creager et al. 2019] etc.
- Adversarially fair representation [D Madras et al. 2018] etc.
- Inherent trade-offs in learning fair representation [H Zhao et al. 2019]
- And more……

UNIVERSITY OF WATERLOO

Some thoughts and conclusions

- The paper formulates the fairness problem in a novel way that deserves a lot of further study
- Some choices of loss functions and mappings are crude, worth discussing if there are better alternatives, e.g. why using 'L1 norm' to compare two probability histogram? Cross-entropy seems to be a more suitable choice
- This 'prototype learning' approach is quite unusual, nowadays most papers on learning fair representation use neural networks. Neural network approach is more flexible and compatible with the problem. The choice in this paper seems to have a historical reason.
- Fair representation learning seems to be restricted to Statistical Parity only, can other definitions of fairness apply? (may not)
- How to *deconstruct* a given classifier to determine to what extent it is fair? (Interpretability)


UNIVERSITY OF
WATERLOO
THANK YOU!