AI System Evaluation

Week 7: Improving AI Fairness

Aug 23 - Week 1: 7-10	Introduction	
Aug 30 - Week 2: 7-10	AI Robustness	Exercise 1
Sep 06 - Week 3: 7-10	Improving AI Robustness	Exercise 2
Sep 13 - Week 4: 7-10	AI Backdoors	Exercise 3
Sep 20 - Week 5: 7-10	Mitigating AI Backdoors	Exercise 4; Project Proposal
Sep 27 - Week 6: 7-10	AI Fairness	Exercise 5
Oct 11 - Week 7: 7-10	Improving AI Fairness	Exercise 6
Oct 18 - Week 8: 7-10	AI Privacy	Exercise 7
Oct 25 - Week 9: 7-10	Improving AI Privacy	Exercise 8
Nov 01 - Week 10: 7-10	AI Interpretability	Project Due
Nov 08 - Week 11: 1-3	End-of-Term Exam	

Outline

Problem Definition

How do build models that are fair according to certain definition of fairness and at the same time maintain the accuracy if possible?

Approaches

Reducing data bias

Reducing algorithmic bias

- Preprocessing
- In-processing
- Post-processing
- Adaptive processing

Reducing Data Bias

Reducing Data Bias

Question

How do we avoid data bias such as selection bias and missing variable bias among others?

As we don’t know the actual data distribution, much of these are best practices only.

Answer

Use diverse and random data

- using random methods when selecting subgroups from populations;
- ensuring that the subgroups selected are equivalent to the population at large in terms of their key characteristics;
- ensuring that all relevant variables are considered;
- …

Preprocessing

Pre-Processing

Intuition

Modifying the training data or its representation before training the model so that the model is more likely fair.

Methods

Suppressing
Relabeling
Reweighting
Sampling

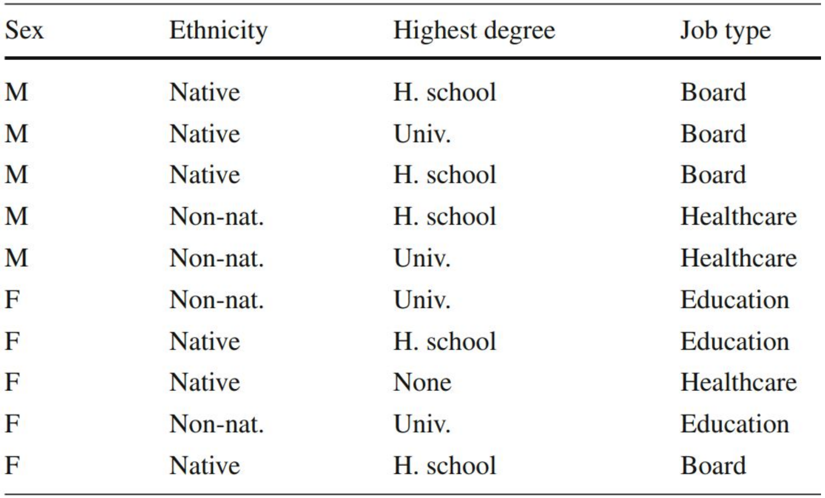
Preprocessing: Suppressing

High-level idea

We remove the sensitive feature.

We further remove features that are strongly correlated with the sensitive feature if necessary.

Example



Sex	Ethnicity	Highest degree	Job type
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

Remove feature “Sex” and then “Job type”.

Preprocessing: Suppressing

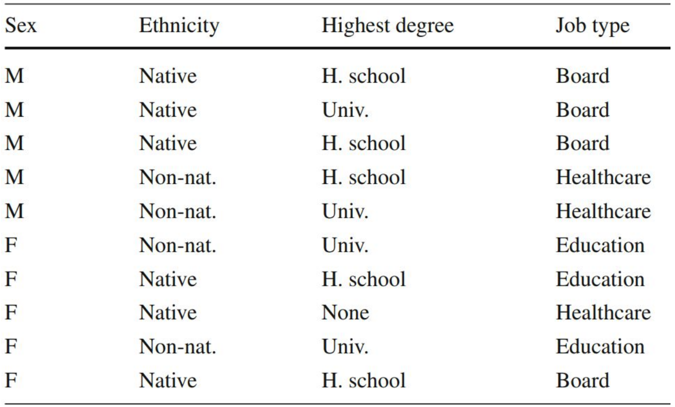
Question

How do we check whether two features are strongly correlated?

- Pearson correlation coefficients
- Spearman correlation coefficients
- p-value

Example

week7/correlation.py compute the correlation between the features in the table.



Sex	Ethnicity	Highest degree	Job type
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

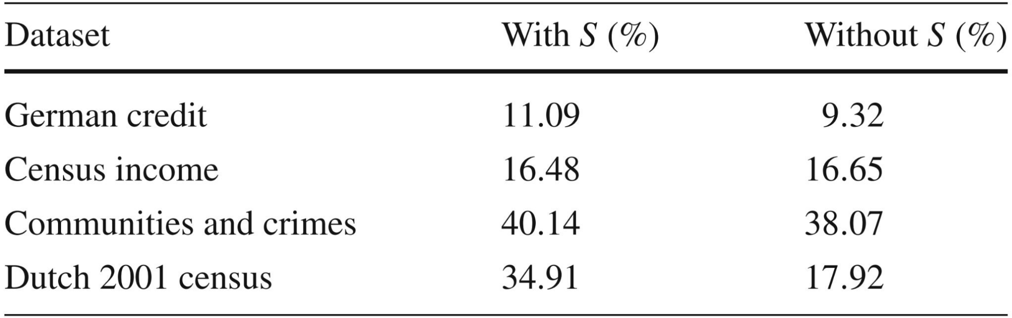
Preprocessing: Suppressing

Suppressing is often not effect

The redlining effect, i.e., removing the sensitive feature from the dataset does not always result in the removal of the discrimination, because of indirect discrimination due to other features that correlate with the sensitive feature.

Experiment

Training (decision trees) with and without the sensitive feature.



Dataset	With S (%)	Without S (%)
German credit	11.09	9.32
Census income	16.48	16.65
Communities and crimes	40.14	38.07
Dutch 2001 census	34.91	17.92

Fairness score $|Pr(y_+|A=a) - Pr(y_+|A\neq a)|$ according to Demographic Parity

Preprocessing: Relabeling

High-level idea

Changing the label of a few selected samples (which is called promotion and demotion).

It is proposed for demographic parity.

These promoted or demoted samples are those close to the decision boundary.

Approach

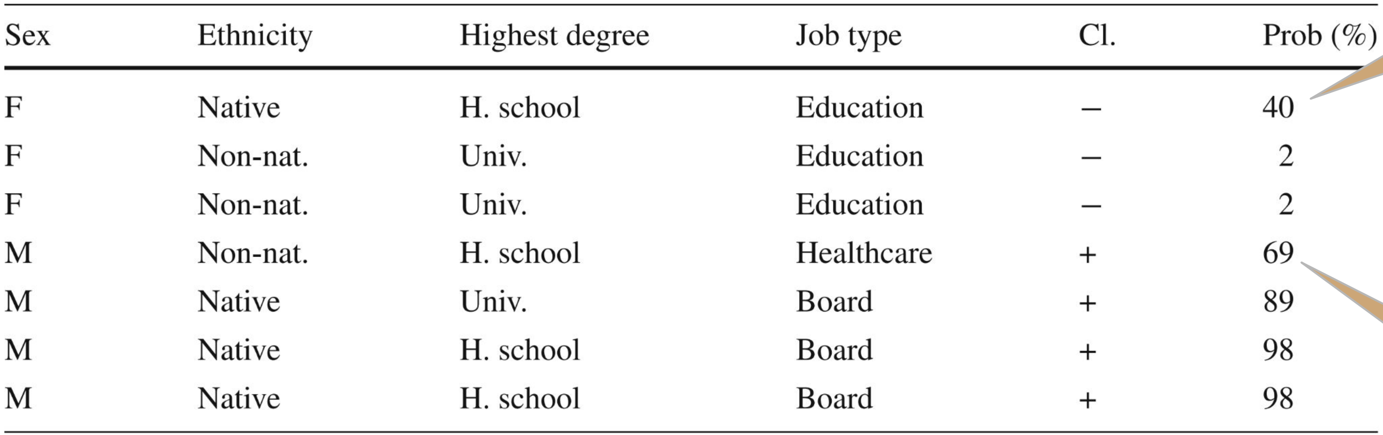
Among those samples which are predicted favorably, select the ones with least confidence for demotion.

Among those samples which are predicted not favorably, select the ones with highest confidence (for predicting the favorable label) for promotion.

Promote or demote as many samples as needed until fairness is satisfied.

Preprocessing: Relabeling

Goal: $|\Pr(y_+|M) - \Pr(y_+|F)| <= \varepsilon$



Sex	Ethnicity	Highest degree	Job type	Cl.	Prob (%)
F	Native	H. school	Education	–	40
F	Non-nat.	Univ.	Education	–	2
F	Non-nat.	Univ.	Education	–	2
M	Non-nat.	H. school	Healthcare	+	69
M	Native	Univ.	Board	+	89
M	Native	H. school	Board	+	98
M	Native	H. school	Board	+	98

Promoted

Demoted

Preprocessing: Reweighting

High-level idea

Relabeling may be considered intrusive since it alters the “truth”.

Reweighting insteads assigns different weights to the training data.

Intuitively, lower weights are assigned to samples that have been deprived or favored.

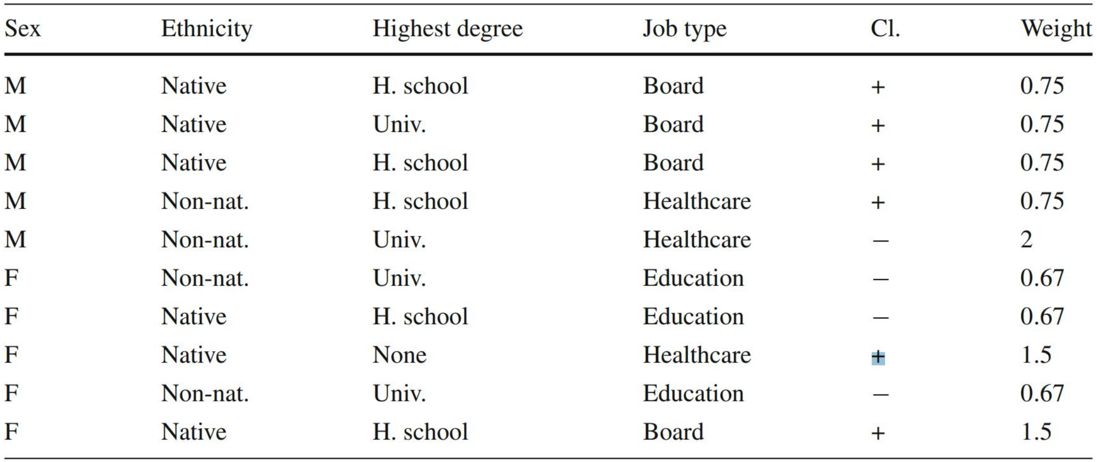
Approach

The weight of a sample x with sensitive feature value a and label y is defined as follows.

(Pr(A=a)*Pr(y))/Pr(A=a,y)

where Pr(A=a)*Pr(y) is the expected probability of any sample with A=a with label y.

Preprocessing: Reweighting



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

Example 1:

The expected Pr(F,+) = 0.5*0.6=0.3;
The observed Pr(F,+)=0.2;
The weight is 0.3/0.2=1.5

Example 2:

The expected Pr(M,-) = 0.5*0.4=0.2;
The observed Pr(M,-)=0.1;
The weight is 0.2/0.1=2

Exercise 1

Work out the weights accordingly for the remaining two cases. Make sure you can see the idea intuitively.

Preprocessing: Sampling

High-level idea

Some machine learning methods do not support weight naturally. Reweighting can be thus realized through sampling.

Approach

Compute the weights of each sample as in the case of reweighting.

Treat the weight as the number of times the sample should be sampled.

Preprocessing: Optimized Representation

High-level idea*

All the previous approaches aim to construct a slightly different dataset which is hopefully fairer. Why don’t we do it systematically and properly.

Given the raw dataset D, construct a new dataset D’ (e.g., by changing the sensitive features or labels) through optimization with three objectives.

Objectives of Preprocessing

Maximize utility: the distribution of non-sensitive features and the labels should be maintained if possible.

Minimize individual distortion: the change of each individual should not be dramatic.

Minimize discrimination: the label should be made independent of the sensitive feature.

Consider how you would define the objective function?

**Optimized pre-processing for discrimination prevention, NIPS 2017.*

Exercise 2: Discussion

It is perhaps fair to say that preprocessing works by “altering” the data from the real-world in some way. Do you approve of such methods?

Inprocessing

Inprocessing

Intuition

The overall idea is to “alter” the training process in some way so that the trained model is more likely to be fair.

Approaches

Fair feature selection

Regularization

Minmax

Neural network repair

Fair representation learning

Inprocessing: Fair Feature Selection

High-level idea*

Feature suppressing could be useful.

Human judgement on which features to use could be unreliable.

We need to systematically select the best suppressing features for accuracy and fairness.

**Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning, AAAI 2012.*

Approach

Solve the following optimization problem.

$$\text{Max}_{fs \subseteq S} \text{ accuracy}(\text{Model}_{fs}) \\ \text{subject } (\text{Model}_{fs} \text{is fair})$$

where Model_{fs} is a model trained with a dataset in which features not in fs are suppressed.

For simple models such as linear classifiers, this problem can be solved efficiently.

Does it work for neural networks?

Inprocessing: Regularization

High-level idea*

Add a regularizer to the objective function so that the trained model is likely fair.

It can be applied to all kinds of fairness definitions (as long as we can evaluate the regularizer efficiently).

Approach

Using the following objective function during the training

$$\mathrm{Min}_{\theta}\ L_{CE}(\theta, x, y)+\lambda*R(\theta, x, y)$$

where R(θ, x, y) is a regularizer which is defined according to the fairness property.

It can be defined as a statistical measure of all kinds of unfairness, e.g., the percentage of discriminatory instances in a sample set.

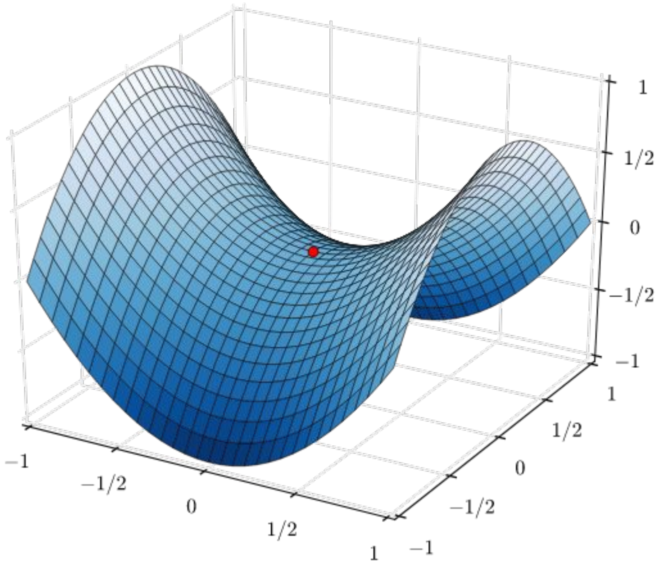
**Preventing undesirable behavior of intelligent machines, Science 2019.*

Inprocessing: Minmax

High-level idea*

Given a model N, learn a new model M such that M aims at maximizing its capability to predict the outcome while minimizing the capability to predict the sensitive feature (so that the prediction is independent of the sensitive feature).

Solved using saddle point methods.



**A Reductions Approach to Fair Classification, ICML 2018.*

Inprocessing: Neural Network Repair

High-level idea*

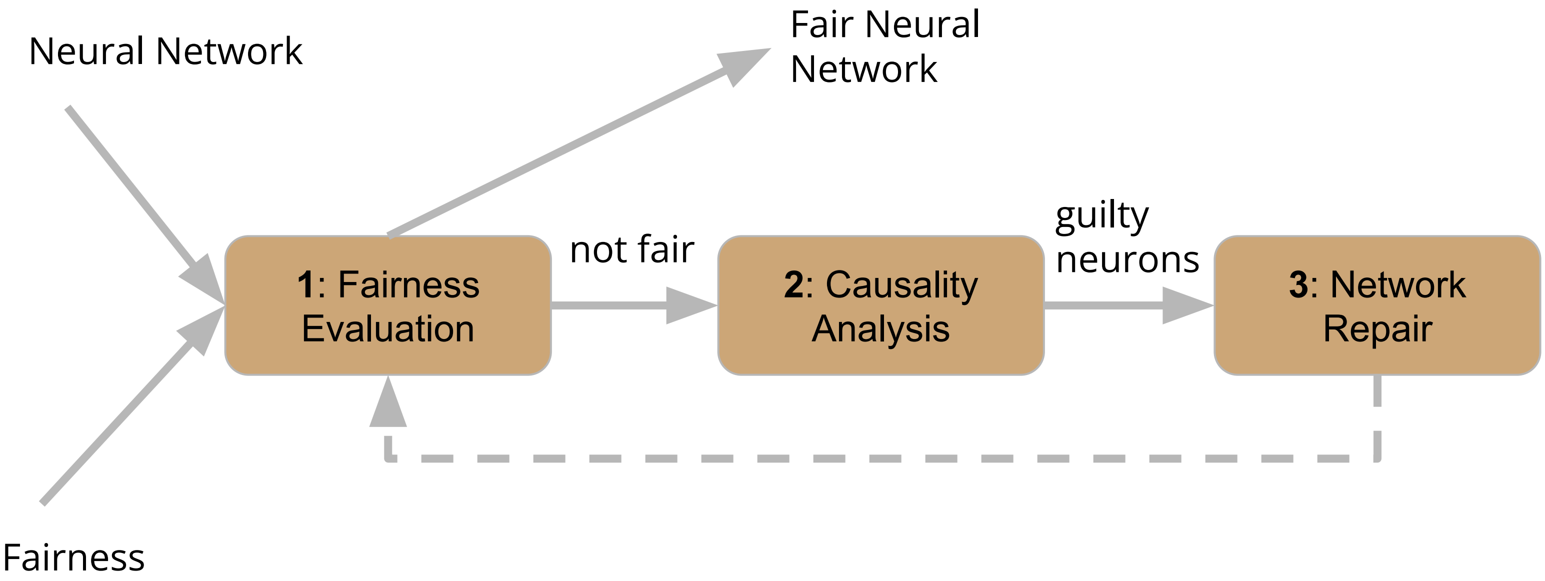
Given a neural network *N* which is shown to be unfair (e.g. with respects to demographic parity), construct a fairer network *N'* by minimally tuning N.

This is similar to program debugging.

The same method can be used to repair other aspects of neural network as well.

**“Causality-based Neural Network Repair”, ICSE 2022.*

Inprocessing: Neural Network Repair



Step 1: Fairness Evaluation

Problem

Given a neural network N and a fairness property (e.g., $|Pr(y_+|A=a) - Pr(y_+|A\neq a)| <= \varepsilon$), how do we systematically evaluate whether N is fair?

Approaches

(Week 6 slides)

Testing methods such as hypothesis testing

Verification methods

Step 1: Fairness Evaluation

Example: Census Income

Task: predict whether an individual’s income exceeds $50K per year

Model: Feed-forward 5-layer neural network

Property: fairness with a threshold of 1%, i.e. unfair if the probability difference of a favourable prediction for females and males is greater than 1%.

Verification Result

The verification algorithm is from [FM’21].

The model is unfair, i.e. *males are 3% more likely to be predicted to have an income exceeding $50K per year.*

[FM’21]: Probabilistic Verification of Neural Networks Against Group Fairness

Step 2: Causality Analysis

Debugging is all about causality.

To debug and repair a system is to conduct causal reasoning, i.e., to understand what is the cause of the undesirable outcome and imagine what would happen if we amend the “cause” in certain way (i.e., **counterfactual** reasoning).

Example

We debug and repair traditional programs as follows.

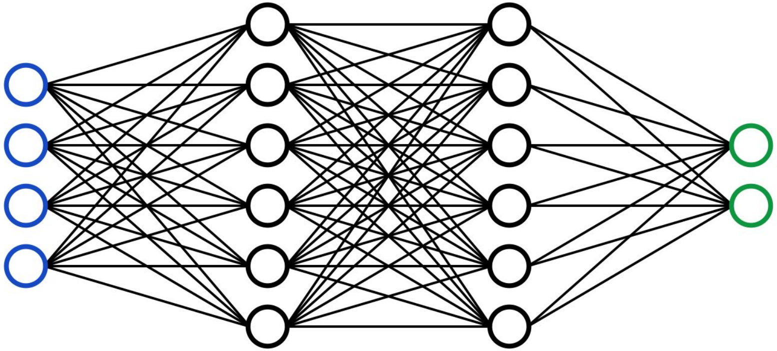
- We conduct causal reasoning through *data and control dependency analysis* based on program semantics.
- We repair programs by modifying the failure-causing statements in certain ways.

How do we find the failure-causing neurons?

Causality for Neural Networks

True: Every neuron is responsible (for certain unexpected outcome).

Also true: Not every neuron bares the same amount of responsibility.



How do we quantify the responsibility of each neuron for each mistake or for all mistakes?

Causal reasoning (J. Pearl)

Level & symbol	Typical activity	Typical questions	Examples
Association *P(y|x)*	Seeing	What is?	An exception occurred when a negative input is there.
Intervention *P(y|do(x), z)*	Intervening	What if?	What if I take the absolute value, will the exception be gone?
Counterfactuals *P(y_x |x', y')*	Imagining	What if I had acted differently?	What if I had taken the absolute value?

Algorithm

Average Causal Effect (ACE)

Measure the causal effect of a variable (x) on the outcome (y) by performing intervention on the variable.

ACE = $\mathbf{E}[y|do(x=1)] - \mathbf{E}[y|do(x=0)]$

Example

The ACE of taking a drug is the difference between the (average) effect of a randomized population taking the drug and the (average) effect of a randomized population not taking the drug.

Calculating ACE for neurons are more complicated since each neuron can take many values.

ACE for Neurons

Algorithm

Let *x* be any neuron.

Let *b* a value of *x*.

Fix *x*'s value to be *b*. Sample many inputs to approximate the impact on *P*.

Average the impact over all possible values of *x* as the *ACE(x)*.

Example: *Fairness*

Assume a neuron *x* whose value ranges from 1 to 10 (with a step size of 1).

We fix *x* to be 1. Sample 1000 inputs.

Calculate the fairness score based on the samples (e.g., $|\Pr(y_+|A=a) - \Pr(y_+|A\neq a)|$ for demographic parity).

Repeat with *x* being 2, 3, ..., 10.

Take the average percentage as *ACE(x)*.

Step 2: Causality Analysis

Some neurons are clearly guiltier than others.

Example: Census Income

Task: predict whether an individual’s income exceeds $50K per year

Model: Feed-forward 5-layer neural network

Property: fairness with a threshold of 1%, i.e. unfair if the probability difference of a favourable prediction for females and males is greater than 1%.

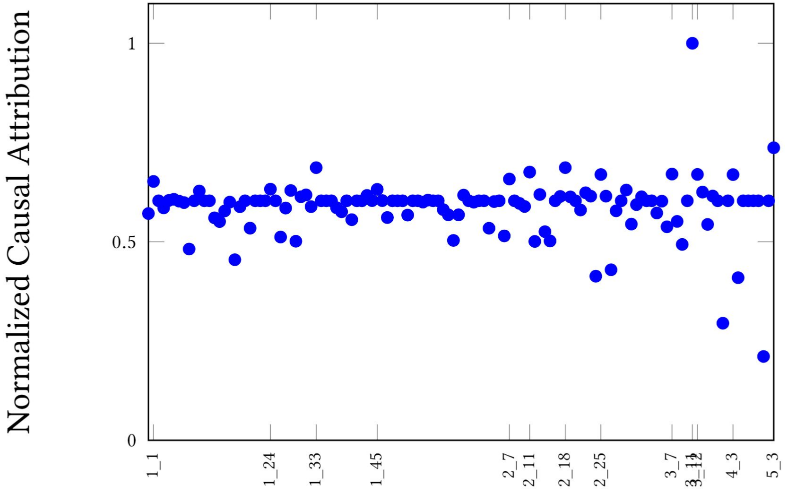


Figure 4: Causal Attribution of Neurons in the Example

Step 3: Network Repair

Optimization-based repair

Identify the most responsible neurons (e.g. the top 10%).

Apply the an optimization algorithm to optimize the weights of these neurons with the following objective function.

*MIN (1-a)*UB+a*(1-accuracy)*

where *UB* is a measure of unwanted behaviors; and *a* is a weight.

Example: *Census Income*

13 neurons are subject for optimization;

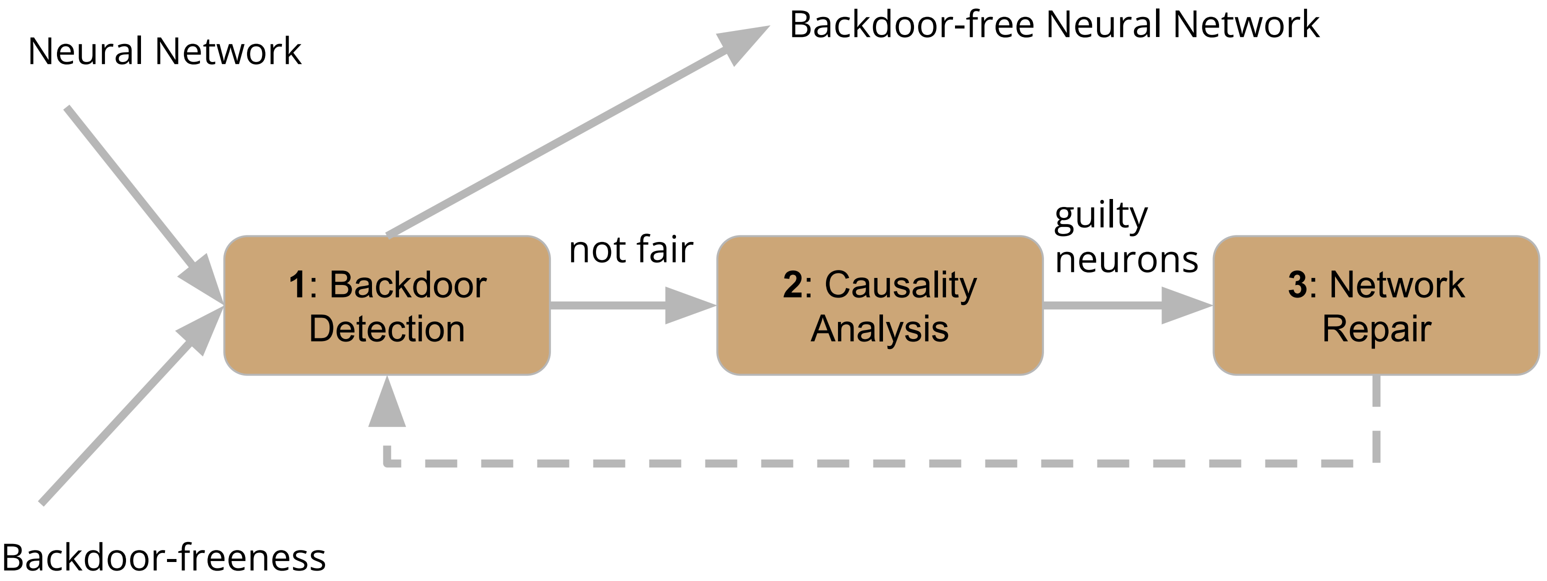
a is set to 0.8;

17 iterations of PSO;

UB is the unfairness (e.g. fairness score)

Unfairness reduces 0.7% and accuracy drops from 88% to 86%.

Causality Analysis for Backdoor Removal



Neural Network

Backdoor-freeness

Backdoor-free Neural Network

1: Backdoor Detection

not fair

2: Causality Analysis

guilty neurons

3: Network Repair

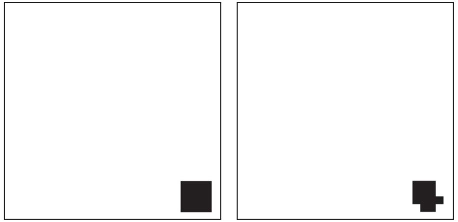
Step 1: Backdoor Detection

Problem

Given a neural network N, how do we systematically detect whether N potentially contains a backdoor and synthesize the backdoor trigger?

Approaches

Neural Cleanse (Week 5 Slide 35)



Original Trigger
(L1 norm = 16)

Reversed Trigger ($\boldsymbol{m} \cdot \boldsymbol{\Delta}$)
(L1 norm = 12.00)

Or we can use your approach.

Step 2: Causality Analysis

Algorithm

Let *x* be any neuron.

Let *b* a value of *x*.

Fix *x*’s value to be *b*. Sample many inputs to approximate the impact of x having value b.

Average the impact over all possible values of *x* as the *ACE(x)*.

Exercise 3

Take Slide 32 as an example, explain how do we approximate the causality of each neuron with respects to backdoor.

Step 3: Network Repair

Optimization-based repair

Identify the most responsible neurons (e.g. the top 10%).

Apply the an optimization algorithm to optimize the weights of these neurons with the following objective function.

*MIN (1-a)*UB+a*(1-accuracy)*

where *UB* is a measure of unwanted behaviors; and *a* is a weight.

Example: *BadNet*

UB is the approximated backdoor effectiveness (e.g. attack success rate) based on the samples.

Backdoor attack success rate drops from 99% to 0%.

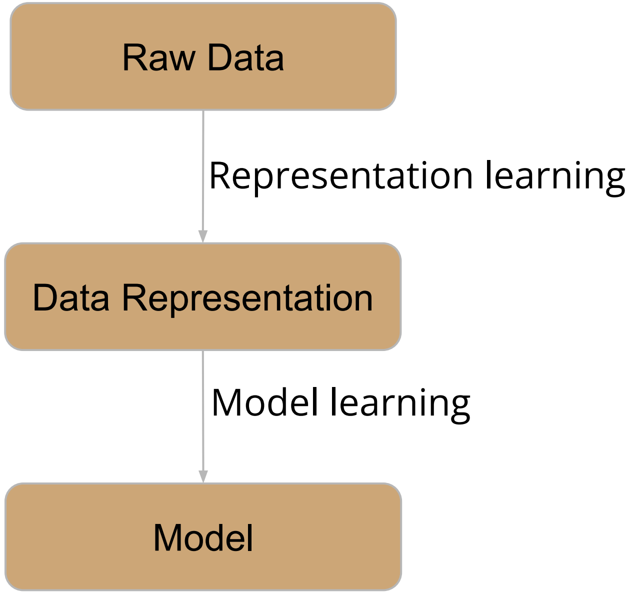
Fair Representation Learning

High-level Idea*

A part of deep learning is to learn a representation of the data.

If we learn a representation of the data which preserves the utility of the data (so that we can still predict accurately) and removes the discrimination, we solve the problem.

**The Variational Fair Autoencoder, ICLR 2016*
**Learning Certified Individually Fair Representations, NeurIPS 2020*



Raw Data

Representation learning

Data Representation

Model learning

Model

Fair Representation Learning

High-level Idea*

Can we learn a model that we can certify its individual fairness on all training samples and perhaps most testing samples?

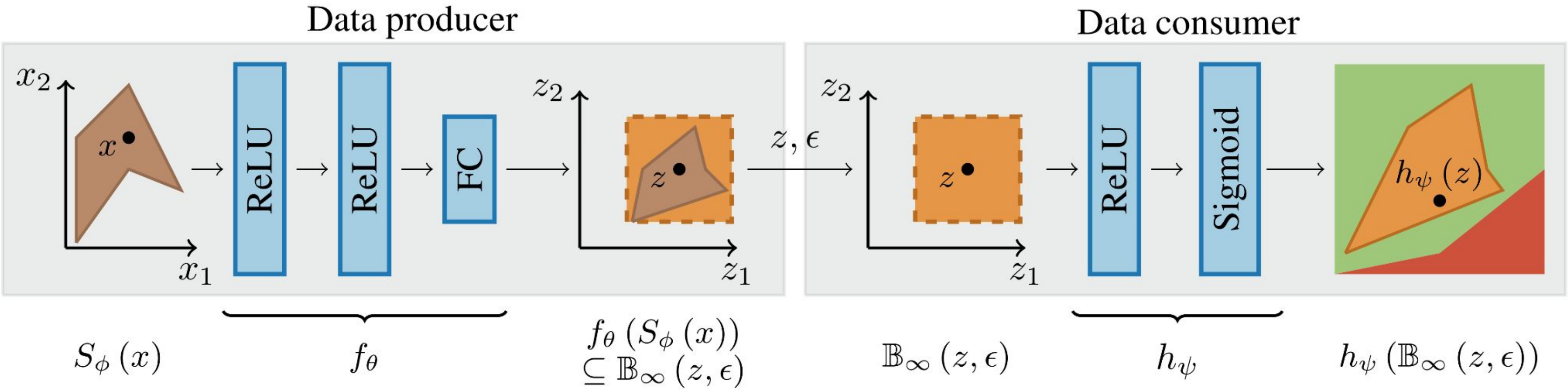
Approach

Inspired by research on certified robustness, the following approach is proposed.

1. Learn a data producer f_θ so that similar individuals (i.e. those should be treated similarly according to individual fairness) are encoded similarly (e.g., within certain L_p-norm).
2. Apply certified training (refer to Week 3) to train a data consumer which is robust.

**Learning Certified Individually Fair Representations, NeurIPS 2020*

Fair Representation Learning



Ideally, if x' differs from x only by the protected feature, $||f_\theta(x)-f_\theta(x')||_\infty < \epsilon$.
Since the data consumer is trained to L_∞ robust, N(x') = N(x) always.

Fair Representation Learning

Step 1: Learning data producer

The data producer should satisfy two constraints. Let simi(x,x’) be true if and only if x and x’ differ only by a protected feature.

- For all x, simi(x,x’) => $||f_\theta(x)-f_\theta(x')||_\infty < \epsilon$
- we are able to accurately predict the label after the data producer processes the data.

Learning Objective

Jointly train the data producer f_θ and a classifier M by minimizing

$$\mathbf{L}(\text{simi}(x,x') \Rightarrow ||f_\theta(x)-f_\theta(x')||_\infty < \epsilon)$$

and a cross-entropy loss of M. Note that **L**(φ) is a measure of how bad φ is violated (i.e., it is positive if φ is violated and 0 if φ is satisfied).

Fair Representation Learning

Step 1: Cont’d

The data producer is not guaranteed to satisfy

$$\text{simi}(x,x') \Rightarrow ||f_\theta(x)-f_\theta(x')||_\infty < \epsilon$$

Address the problem by solving the following optimization problem to relax ε.

$$\max ||f_\theta(x)-f_\theta(x')\ ||_\infty \text{subject simi}(x,x')$$

Step 2: Learning data consumer

The data consumer should be robust within the L_∞ norm.

Certified training (refer to Week 3)

$$\text{Min}_\theta\ \text{Max}_{D(x,x')<\varepsilon}\ L(\theta, x', y)$$

What do you think?

Postprocessing

Postprocessing

High-level idea

We keep the data as it is (since they do represent the real-world) and we do not interfere the training process (since fairness would come at the cost of accuracy).

We instead “alter” the predictions (in order to abide the laws).

Approaches

Individual+group debiasing

Cost-of-Fairness

Equal opportunity predictor

Individual+Group Debiasing

High-level Idea*

To improve group fairness, instead of postprocessing every sample, target those individual samples which are more likely to suffer from individual discrimination.

Approach

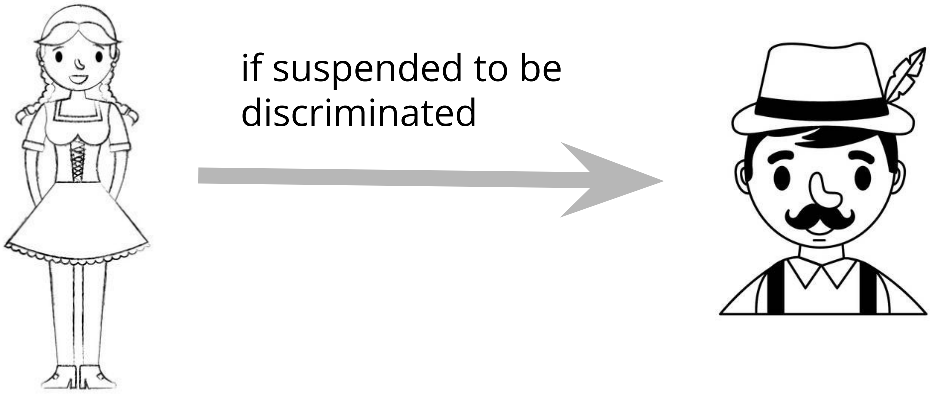
Train a classifier to predict whether a sample is likely to suffer from individual discrimination.

If an unprivileged individual (e.g., a female) is predicted to suffer from individual discrimination, set the prediction to be the one that would be the case if the individual is privileged (e.g., change the prediction to be the one if she were a male).

**Bias Mitigation Post-processing for Individual and Group Fairness. ICASSP 2019*

Individual+Group Debiasing

Example: Census Income dataset



if suspended to be discriminated

[4, 0, 6, 6, 0, 1, 2, 1, **0**, 0, 0, 40, 100]

[4, 0, 6, 6, 0, 1, 2, 1, **1**, 0, 0, 40, 100]

Which fairness this method improves?

Compare this to relabelling.

Cost-Of-Fairness

High-level idea*

Demographic parity comes at a cost of accuracy.

$$|Pr(y_+|A=a) - Pr(y_+|A≠a)| <= ε$$

The goal is to minimize the cost whilst being fair.

**Algorithmic decision making and the cost of fairness, KDD 2017*.

Approach

Use a set of decision rules which takes the predictions of the model N and process the predictions.

Given a sample x where A=a with prediction y_+ according to N, predict y_+ only if Pr(N, x, y_+) > Th_a where Pr(N, x, y_+) is the probability of predicting y_+(a.k.a. score) and Th_a is a threshold specific to sensitive feature value a.

Th_a is identified through optimization.

Equal Opportunity Predictor

E.g., higher SAT threshold for Asian kids.

High-level idea*

Demographic parity is fundamentally problematic and we should aim for equal opportunity.

The difference between true-positive rates (TPRs) of the two groups should be bounded.

$$|Pr(y_+|A=a, y_+) - Pr(y_+|A \neq a, y_+)| <= \varepsilon$$

Approach

Build an equal opportunity predictor which takes the predictions of the model N and process the predictions.

Given a sample x where A=a with prediction y_+ according to N, predict y_+ only if $Pr(N, x, y_+) > Th_a$ where $Pr(N, x, y_+)$ is the probability of predicting y_+(a.k.a. score) and Th_a is a threshold specific to sensitive feature value a.

Th_a is identified through optimization.

**Equality of Opportunity in Supervised Learning, NIPS 2016.*

COF vs. EOP

Example: Cost-of-Fairness

On average, a student has a 10% chance of entering an elite university.

Certain racial group of student C has a 20% chance of entering an elite university.

Use a threshold specific for students of C so that only 10% of them are admitted.

Which is better?

Example: Equal Opportunity Predictor

Assume that fairness requirement is every qualified students has a 50% chance of entering an elite university.

Certain racial group of students W has a 60% chance of entering an elite university if they are qualified.

Use a threshold specific for W so that only 50% of the qualified students are admitted.

Adaptive Processing

Adaptive Processing

High-level idea*

It is difficult to decide when to apply preprocessing, inprocessing or postprocessing.

Applying different fairness improving methods may incur different costs in terms of accuracy and may even reduce fairness.

"Adaptive Fairness Improvement based Causality Analysis", ESEC/FSE 2022

Approach

Apply causality analysis to determine which is the most responsible for the unfairness, i.e., is it more due to the inputs or due to certain hidden neurons? If it is the latter, how are the responsibility distributed.

Based on the causality analysis results, choose preprocessing, inprocessing or postprocessing methods accordingly.

Adaptive Processing: Empirical Study

Experimental Setup

An empirical study is conducted to compare the effect of different fairness improving methods on different.

Dataset

Adult Income (gender and race)
German Credit (gender and age)
Bank Marketing (age)
COMPAS (gender and race)

Fairness Improving Methods

Preprocessing

- Reweighting (RW)
- Disparate Impact Remover (DIR)

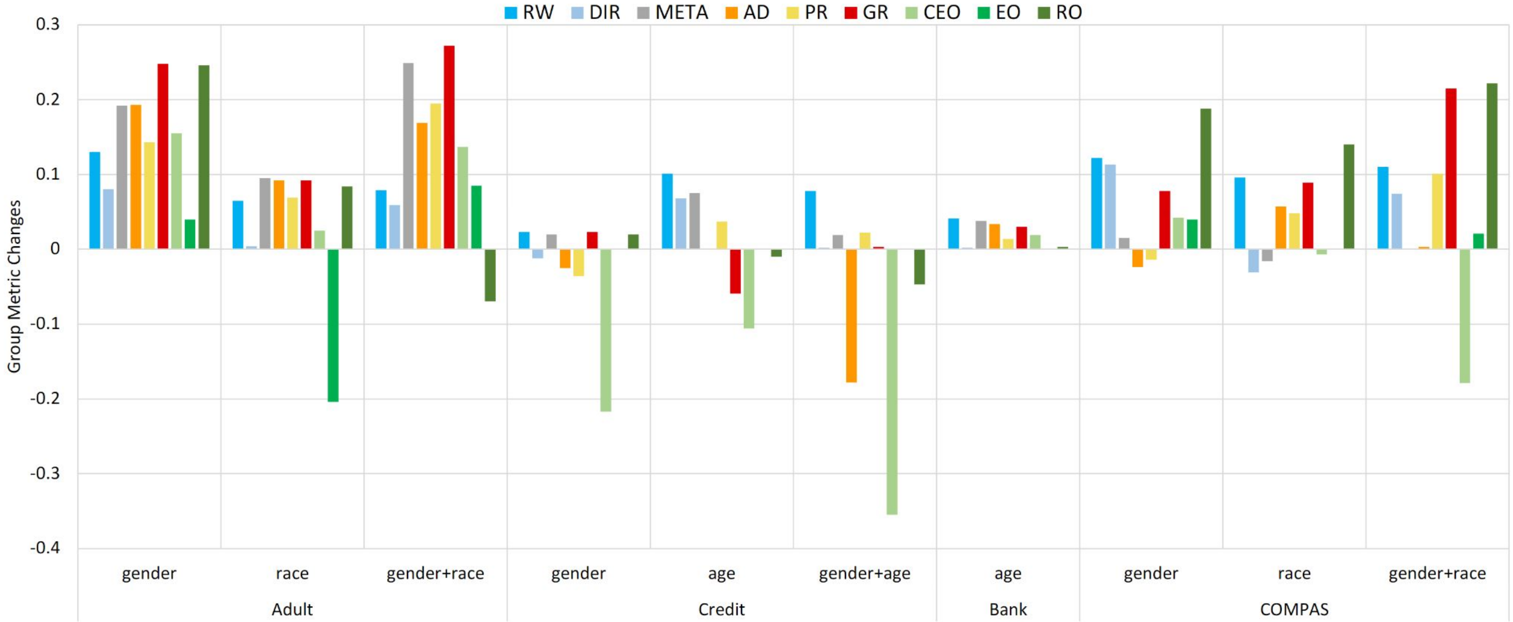
Inprocessing:

- Classification with fairness constraints (META)
- Adversarial debiasing (AD)
- Prejudice remover regularizer (PR)
- Exponential gradient reduction (GR)

Postprocessing

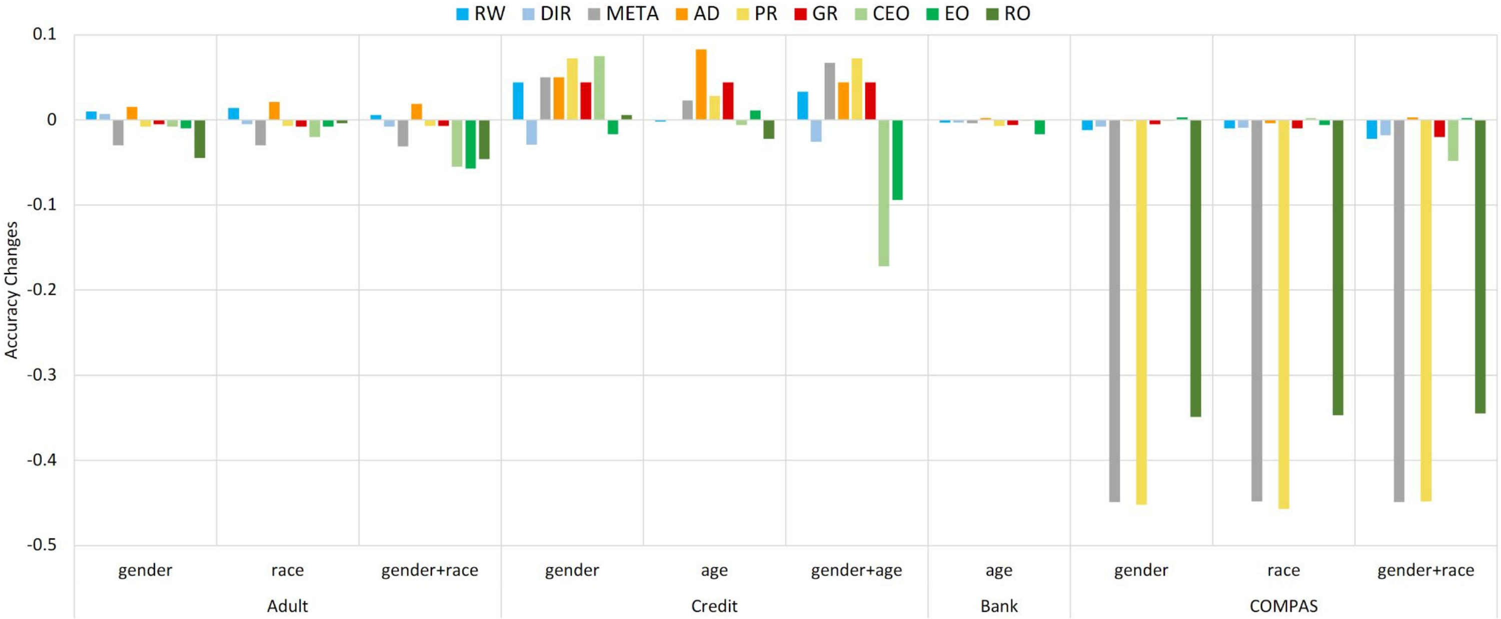
- Equalized Odds (EO)
- Calibrated Equalized Odds (CEO)
- Reject Option Classification (RO)

Adaptive Processing: Empirical Study



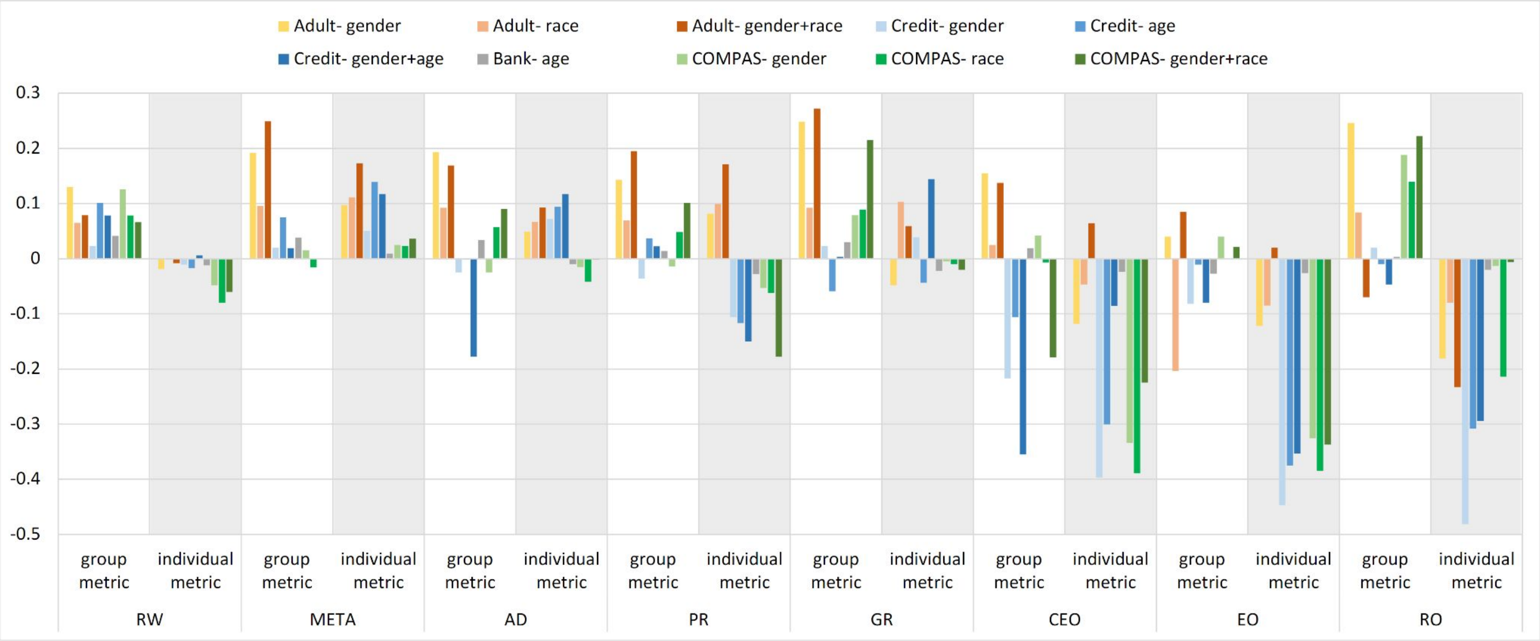
Fairness improvement in terms of demographic parity.

Adaptive Processing: Empirical Study



Accuracy cost

Adaptive Processing: Empirical Study



Demographic parity vs. individual fairness

Adaptive Processing

Approach

Conduct causality analysis as shown on Slide 29-33.

Based on the causality analysis result, choose either preprocessing, inprocessing or postprocessing.

Algorithm

If few input neurons or hidden neurons (e.g., less than 10%) are responsible for unfairness (more so than on average), apply postprocessing.

Otherwise, if input neurons are more responsible than hidden neurons, apply preprocessing.

Otherwise, apply inprocessing.

Does it make sense?

Conclusion

Fairness can improved through preprocessing (i.e., process the data), or inprocessing (i.e., process the model) or postprocessing (i.e., process the predictions) or adaptively.

Fairness improvement often costs accuracy.

Exercise 4

The program week7/exercise4/train_model_orig.py trains a neural network to predict whether an individual makes more than 50K annually.

- Apply Suppressing to train two new models, one suppressing the gender attribute and the other suppressing an additional correlated attribute (i.e., the one which is most correlated to the gender attribute according to Spearman coefficient).
- Compare the accuracy of the three models.
- Compare the fairness score |Pr(>50K|Male) - Pr(>50K|Female)| of the three models.

Assignment Exercise 6

Submit a zip file containing a report (word, or pdf) and programs showing your working of Exercise 1-4 to elearn (under Assignments and Exercise 6) by Oct 17, 2022 11:59 PM.

Aug 23 - Week 1: 7-10	Introduction	
Aug 30 - Week 2: 7-10	AI Robustness	Exercise 1
Sep 06 - Week 3: 7-10	Improving AI Robustness	Exercise 2
Sep 13 - Week 4: 7-10	AI Backdoors	Exercise 3
Sep 20 - Week 5: 7-10	Mitigating AI Backdoors	Exercise 4; Project Proposal
Sep 27 - Week 6: 7-10	AI Fairness	Exercise 5
Oct 11 - Week 7: 7-10	Improving AI Fairness	Exercise 6
Oct 18 - Week 8: 7-10	AI Privacy	Exercise 7
Oct 25 - Week 9: 7-10	Improving AI Privacy	Exercise 8
Nov 01 - Week 10: 7-10	AI Interpretability	Project Due
Nov 08 - Week 11: 1-3	End-of-Term Exam	