Ensuring Metric-optimal Fairness for Online Decision Making

Metric-Free Individual Fairness in Online Learning, (Bechavod et al., 2020)
&
A Statistical Test for Probabilistic Fairness, (Taskesen et al., 2020)

Maxime BOUTON

Supervisor: Debabrota BASU

Université de Lille
University of the North (France)

Project Defense, April 9, 2021

Table of Contents

Table of Contents

What's fairness ?

Fairness

- One possible definition : *absence of any prejudice or favouritism toward an individual or a group based on their inherent or acquired characteristics*
  - This is not universal.
  - Hard to achieve in practice
- Different meanings whether either groups or individuals are considered

What's fairness ?

Fairness

- One possible definition : *absence of any prejudice or favouritism toward an individual or a group based on their inherent or acquired characteristics*
    - This is not universal.
    - Hard to achieve in practice
- Different meanings whether either groups or individuals are considered

In practice ?

What's fairness ?

Fairness

- One possible definition : *absence of any prejudice or favouritism toward an individual or a group based on their inherent or acquired characteristics*
  - This is not universal.
  - Hard to achieve in practice
- Different meanings whether either groups or individuals are considered

In practice ?

- Mathematical definition ?
- Numerically measurable ?
- Implementation ?

Fairness in Artificial Intelligence and Machine Learning

A need for fairness in the decision

- Avoid bias in human decision
- Popularity and importance of ML is growing
- Examples: justice decision (COMPAS), health insurance, school admissions...

Fairness in Artificial Intelligence and Machine Learning

A need for fairness in the decision

- Avoid bias in human decision
- Popularity and importance of ML is growing
- Examples: justice decision (COMPAS), health insurance, school admissions...

Bias is omnipresent in data

Different types of bias: measurement, historical, Simpson's paradox...

Fairness in Artificial Intelligence and Machine Learning

A need for fairness in the decision

- Avoid bias in human decision
- Popularity and importance of ML is growing
- Examples: justice decision (COMPAS), health insurance, school admissions...

Bias is omnipresent in data

Different types of bias: measurement, historical, Simpson's paradox...

Solution approaches

- Try to avoid interference of sensitive attributes
- Include users from sensitive groups
- Treat protected attributes as noise

Illustration

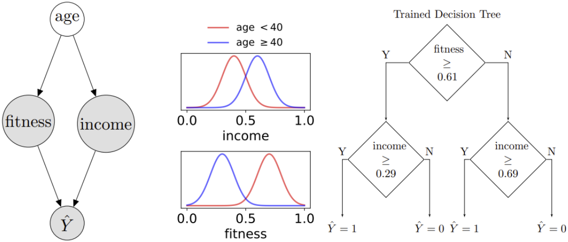


Figure: Example for health insurance eligibility

Illustration

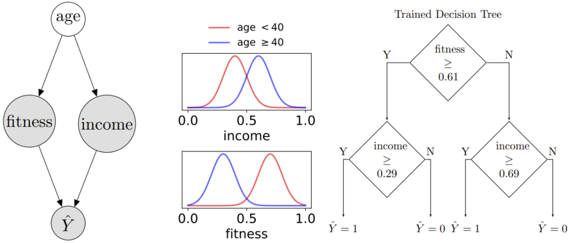


Figure: Example for health insurance eligibility

Problem

On average, a person under 40 is eligible with a probability of 18% while someone over 40 is eligible with a probability of 72%.

Table of Contents

Model presentation

Recall

Online classification: training and prediction in real time, data becomes available in sequential order

Description of our model

- Binary classification
- π : the policy deployed (here our classifier)
- $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$: an arbitrary distance function
- α - violation on the pair (x, x'): $|\pi(x) - \pi(x')| > d(x, x') + \alpha$
- $\mathcal{J}$: the auditor (spots α - violations)
- $\rho = (\rho_1, \rho_2)$: pair of indices

Main lines of implementation

1. A policy is chosen

Main lines of implementation

1. A policy is chosen
2. The related classifier receives a batch $(\bar{x}, \bar{y})$ for training (online classification)

Main lines of implementation

1. A policy is chosen
2. The related classifier receives a batch $(\bar{x}, \bar{y})$ for training (online classification)
3. This batch is audited by the auditor, which spots violations i.e. it spots $(\rho_1, \rho_2) : |\pi(\bar{x}_{\rho_1}) - \pi(\bar{x}_{\rho_2})| > d(\bar{x}_{\rho_1}, \bar{x}_{\rho_2}) + \alpha$

Main lines of implementation

1. A policy is chosen
2. The related classifier receives a batch $(\bar{x}, \bar{y})$ for training (online classification)
3. This batch is audited by the auditor, which spots violations i.e. it spots $(\rho_1, \rho_2) : |\pi(\bar{x}_{\rho_1}) - \pi(\bar{x}_{\rho_2})| > d(\bar{x}_{\rho_1}, \bar{x}_{\rho_2}) + \alpha$
4. If the auditor spots an α - violation on any pair $(\bar{x}_{\rho_1}, \bar{x}_{\rho_2})$, it adds C copies of $\bar{x}_{\rho_1}$ with label 0 and C copies of $\bar{x}_{\rho_2}$ with label 1 to the batch.

Main lines of implementation

1. A policy is chosen
2. The related classifier receives a batch $(\bar{x}, \bar{y})$ for training (online classification)
3. This batch is audited by the auditor, which spots violations i.e. it spots $(\rho_1, \rho_2) : |\pi(\bar{x}_{\rho_1}) - \pi(\bar{x}_{\rho_2})| > d(\bar{x}_{\rho_1}, \bar{x}_{\rho_2}) + \alpha$
4. If the auditor spots an α - violation on any pair $(\bar{x}_{\rho_1}, \bar{x}_{\rho_2})$, it adds C copies of $\bar{x}_{\rho_1}$ with label 0 and C copies of $\bar{x}_{\rho_2}$ with label 1 to the batch.
5. If not, it chooses any x in the batch and adds C copies of x with label 0 and C copies of x with label 1

Main lines of implementation

1. A policy is chosen
2. The related classifier receives a batch $(\bar{x}, \bar{y})$ for training (online classification)
3. This batch is audited by the auditor, which spots violations i.e. it spots $(\rho_1, \rho_2) : |\pi(\bar{x}_{\rho_1}) - \pi(\bar{x}_{\rho_2})| > d(\bar{x}_{\rho_1}, \bar{x}_{\rho_2}) + \alpha$
4. If the auditor spots an α - violation on any pair $(\bar{x}_{\rho_1}, \bar{x}_{\rho_2})$, it adds C copies of $\bar{x}_{\rho_1}$ with label 0 and C copies of $\bar{x}_{\rho_2}$ with label 1 to the batch.
5. If not, it chooses any x in the batch and adds C copies of x with label 0 and C copies of x with label 1
6. The policy incurs a "classic" online batch classification process on $(\bar{x}, \bar{y})$ to which the 2C copies were added

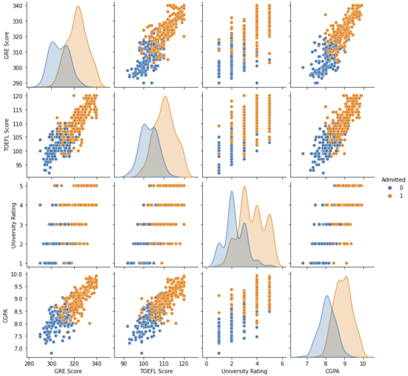
Main lines of implementation

1. A policy is chosen
2. The related classifier receives a batch $(\bar{x}, \bar{y})$ for training (online classification)
3. This batch is audited by the auditor, which spots violations i.e. it spots $(\rho_1, \rho_2) : |\pi(\bar{x}_{\rho_1}) - \pi(\bar{x}_{\rho_2})| > d(\bar{x}_{\rho_1}, \bar{x}_{\rho_2}) + \alpha$
4. If the auditor spots an α - violation on any pair $(\bar{x}_{\rho_1}, \bar{x}_{\rho_2})$, it adds C copies of $\bar{x}_{\rho_1}$ with label 0 and C copies of $\bar{x}_{\rho_2}$ with label 1 to the batch.
5. If not, it chooses any x in the batch and adds C copies of x with label 0 and C copies of x with label 1
6. The policy incurs a "classic" online batch classification process on $(\bar{x}, \bar{y})$ to which the 2C copies were added
7. Restart the process from step 2

School admission

	GRE Score	TOEFL Score	University Rating	CGPA	Admitted
4	314.0	103.0	2.0	8.21	0.0
5	330.0	115.0	5.0	9.34	1.0
6	321.0	109.0	3.0	8.20	1.0
7	308.0	101.0	2.0	7.90	0.0
8	302.0	102.0	1.0	8.00	0.0

(a) Slice of the dataset



(b) Pairplot visualization

Results

Offline score: 0.85
Online score: 0.80

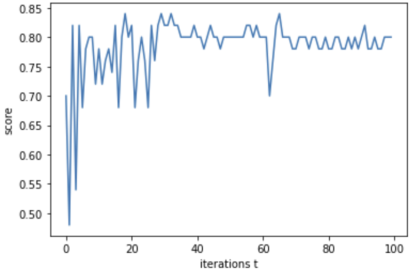
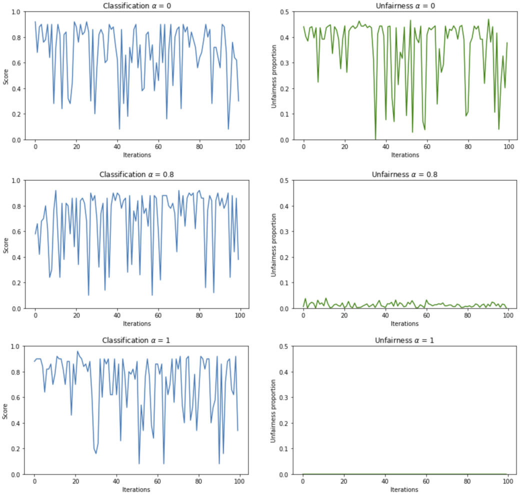


Figure: Online

Results

Online score for fair settings: ...



Conclusion

Potential reasons

- Solution theoretically working but not convenient in practice
- Error coming from a statistical selection (and copies)

Possible corrections ?

- Requires a better understanding, and control on copies (boosting ?)
- Studying group fairness can lead to potential solutions

Table of Contents

Presentation of the problem

Dealing with groups

- Individuals are grouped by sensitive (or protected) attributes α
- Sensitive attributes do not appear in the training process

Main lines

- Build a fair online classifier

Presentation of the problem

Dealing with groups

- Individuals are grouped by sensitive (or protected) attributes α
- Sensitive attributes do not appear in the training process

Main lines

- Build a fair online classifier
- Relying on a statistical test for fairness ($\mathcal{H}_0$: the classifier is fair)

Presentation of the problem

Dealing with groups

- Individuals are grouped by sensitive (or protected) attributes α
- Sensitive attributes do not appear in the training process

Main lines

- Build a fair online classifier
- Relying on a statistical test for fairness ($\mathcal{H}_0$: the classifier is fair)
- Based on optimal transport (OT)

Introduction to Optimal Transport

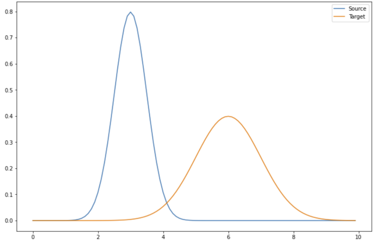


Figure: Pairplot visualization

Introduction to Optimal Transport

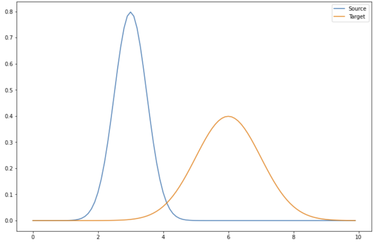


Figure: Pairplot visualization

How can we measure it ?

Wasserstein distance: $\mathbb{W}(\mathbb{Q}, \mathbb{Q}') = \min_{\pi \in \Pi(\mathbb{Q}, \mathbb{Q}')} \sqrt{\mathbb{E}[c(\epsilon, \epsilon')^2]}$
In our case, it will be used to compute, the **most favorable distribution**

$$\mathbb{Q}^* = \arg\min_{\mathbb{Q} \in \mathcal{F}} \mathbb{W}(\mathbb{P}^N, \mathbb{Q})$$

The adopted approach

Initial idea

Use the Wasserstein distance as a regularizer of our online classifier

Actual implementation

The adopted approach

Initial idea

Use the Wasserstein distance as a regularizer of our online classifier

Actual implementation

1. A training batch is sampled

The adopted approach

Initial idea

Use the Wasserstein distance as a regularizer of our online classifier

Actual implementation

1. A training batch is sampled
2. The training classifier parameters are updated w.r.t. the batch

The adopted approach

Initial idea

Use the Wasserstein distance as a regularizer of our online classifier

Actual implementation

1. A training batch is sampled
2. The training classifier parameters are updated w.r.t. the batch
3. The most favorable distribution can be computed from

$$\begin{aligned}\mathcal{R}(\mathbb{P}^N, \hat{p}^N) &= \mathbb{W}(\mathbb{P}^N, \mathbb{Q}^*)^2 \\ &= \sup_{\gamma \in \mathbb{R}} \frac{1}{N} \sum_{i \in \mathcal{I}_1} \min_{k_i \in [0, 1/8]} \gamma^2 \lambda_i^2 \|\beta\|_2^2 k_i^2 + \frac{\gamma \lambda_i}{1 + \exp\left(\gamma \lambda_i \|\beta\|_2^2 k_i - \beta^\top \hat{x}_i\right)}\end{aligned}$$

The adopted approach

Initial idea

Use the Wasserstein distance as a regularizer of our online classifier

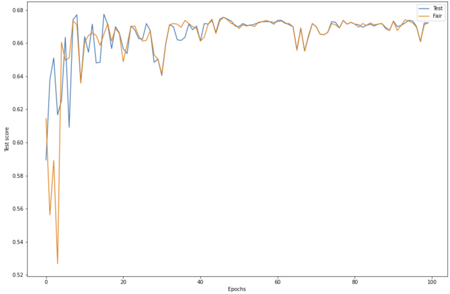
Actual implementation

1. A training batch is sampled
2. The training classifier parameters are updated w.r.t. the batch
3. The most favorable distribution can be computed from

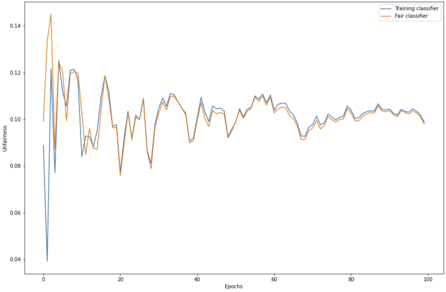
$$
\begin{aligned}
\mathcal{R}(\mathbb{P}^N, \hat{p}^N) &= \mathbb{W}(\mathbb{P}^N, \mathbb{Q}^*)^2 \\
&= \sup_{\gamma \in \mathbb{R}} \frac{1}{N} \sum_{i \in \mathcal{I}_1} \min_{k_i \in [0, 1/8]} \gamma^2 \lambda_i^2 \|\beta\|_2^2 k_i^2 + \frac{\gamma \lambda_i}{1 + \exp\left(\gamma \lambda_i \|\beta\|_2^2 k_i - \beta^\top \hat{x}_i\right)}
\end{aligned}
$$

4. Train another classifier on this most favorable distribution

Results

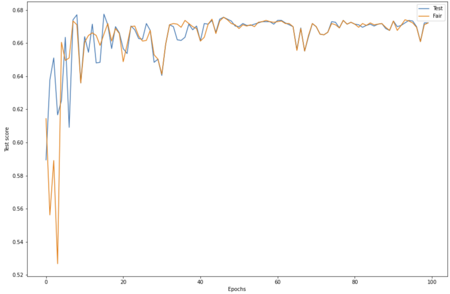


(a) Scores

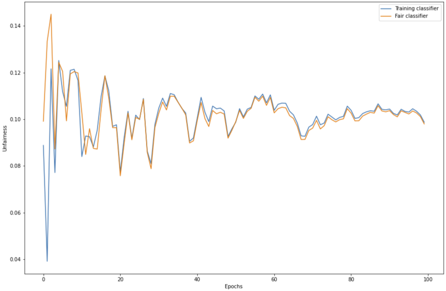


(b) Unfairness

Results



(a) Scores



(b) Unfairness

Conclusion

- Slight improvement, but almost negligible
- Should be much better with Wasserstein projection as regularizer

Thank you for your attention !

References

1. *Metric-Free Individual Fairness in Online Learning*, Bechavod et al., 2020
2. *A Statistical Test for Probabilistic Fairness*, Taskesen et al., 2020
3. *Justicia: A Stochastic SAT Approach to Formally Verify Fairness*, B.GHOSH, D. BASU, K.S. MEEL, 2020
4. *A Survey on Bias and Fairness in Machine Learning*, N. MEHRABI, 2019
5. *Pain-Free Random Differential Privacy with Sensitivity Sampling*, B.I.P RUBINSTEIN, F. ALDA, 2017