Fair machine learning

Lecture 2

Changho Suh
EE, KAIST

Aug. 25, 2021

A fair classifier
using kernel density estimation

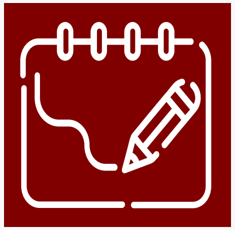
Reading: TN2

Recap: Trustworthy AI

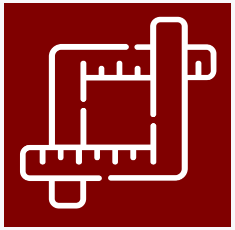
focus of this tutorial

fairness

robustness



explainability



value alignment

transparency

Recap: Fair classifiers

Focused on **group fairness**.

Studied two fairness measures:

prediction (hard decision)

1. $\text{DDP} := \sum_{z \in \mathcal{Z}} |\mathbb{P}(\tilde{Y} = 1 | Z = z) - \mathbb{P}(\tilde{Y} = 1)|$

sensitive attribute e.g., race

2. $\text{DEO} := \sum_{y \in \mathcal{Y}} \sum_{z \in \mathcal{Z}} |\mathbb{P}(\tilde{Y} = 1 | Y = y, Z = z) - \mathbb{P}(\tilde{Y} = 1 | Y = y)|$

Recap: Fairness-regularized optimization

$$\min_{w} \frac{1-\lambda}{m} \sum_{i=1}^{m} \ell_{\mathsf{CE}}(y^{(i)}, \hat{y}^{(i)}) + \lambda \cdot \mathsf{DDP}$$

Studied another approach which employs a different regularization term:

$$I(Z; \hat{Y})$$

Recap: MI-based optimization

$$\min_{w} \frac{1-\lambda}{m} \sum_{i=1}^{m} \ell_{\mathsf{CE}}(y^{(i)}, \hat{y}^{(i)}) + \lambda \cdot I(Z; \hat{Y})$$

$$I(Z;\hat{Y}) \approx \underbrace{H(Z)}_{} + \max_{D(\hat{y};z):\sum_z D(\hat{y};z)=1} \sum_{i=1}^{m} \frac{1}{m} \log D(\hat{y}^{(i)}; z^{(i)})$$

irrelevant of (θ, w)

Parameterize $D(\cdot;\cdot)$ with θ

Recap: MI-based optimization

$$\min_{w} \max_{\theta: \sum_z D_\theta(\hat{y};z)=1} \frac{1}{m} \left\{ \sum_{i=1}^{m} (1-\lambda)\ell_{\mathsf{CE}}(y^{(i)}, \hat{y}^{(i)}) + \lambda \sum_{i=1}^{m} \log D_\theta(\hat{y}^{(i)}; z^{(i)}) \right\}$$

Yields a good tradeoff performance, yet suffering from training instability (due to “min-max” structure)

Claimed: There is another fair classifier that addresses training instability while offering a better tradeoff.

Today's lecture

Will study the new fair classifier in depth.

1. Explore a way to directly compute the fairness measure DDP.
2. Introduce a trick that allows us to well approximate DDP:

   Kernel Density Estimation (KDE)

3. Formulate a KDE-based optimization for a fair classifier.
4. Study how to solve the optimization.

Revisit: the fairness measure DDP

$$\text{DDP} := \sum_{z \in \mathcal{Z}} |\mathbb{P}(\tilde{Y} = 1 | Z = z) - \mathbb{P}(\tilde{Y} = 1)|$$

Let's try to compute this directly.

First focus on:

$$\mathbb{P}(\tilde{Y} = 1) = \mathbb{P}(\hat{Y} \geq \tau) \qquad \tilde{Y} := \mathbf{1}\{\hat{Y} \geq \tau\}$$

$$= \int_{\tau}^{\infty} \underbrace{f_{\hat{Y}}(t)}_{\text{pdf}} dt$$

pdf uknown!

Instead: We are given samples $\{\hat{y}^{(1)}, \ldots, \hat{y}^{(m)}\}$

Question: A way to infer the pdf from samples?

Kernel density estimation (KDE)

$$\mathbb{P}(\tilde{Y} = 1) = \int_{\tau}^{\infty} f_{\hat{Y}}(t) dt$$

Given samples $\{\hat{y}^{(1)}, \ldots, \hat{y}^{(m)}\}$, KDE is defined as:

$$\widehat{f}_{\hat{Y}}(t) := \frac{1}{mh} \sum_{i=1}^{m} f_{\text{ker}} \left(\frac{t - \hat{y}^{(i)}}{h} \right)$$

a smoothing parameter

a kernel function (e.g., Gaussian kernel)

$$f_{\text{ker}}(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}$$

Accuracy of KDE?

$$\mathbb{P}(\tilde{Y} = 1) = \int_{\tau}^{\infty} f_{\hat{Y}}(t)dt$$

Given samples $\{\hat{y}^{(1)}, \ldots, \hat{y}^{(m)}\}$, KDE is defined as:

$$\widehat{f}_{\hat{Y}}(t) := \frac{1}{mh} \sum_{i=1}^{m} f_{\text{ker}} \left(\frac{t - \hat{y}^{(i)}}{h} \right)$$

Jiang ICML17: $|\widehat{f}(t) - f(t)|_{\infty} \lesssim \frac{1}{m^{\frac{1}{d}}}$ ← dim. of an interested r.v.

→ Yields an inaccurate estimate under **high-dim.** settings

Good news: In our setting, $d = 1$

Approximation via KDE

$$\mathbb{P}(\tilde{Y}=1) = \int_{\tau}^{\infty} f_{\hat{Y}}(t)\,dt$$

$$\widehat{\mathbb{P}}(\tilde{Y}=1) = \int_{\tau}^{\infty} \widehat{f}_{\hat{Y}}(t)\,dt$$

$$= \int_{\tau}^{\infty} \frac{1}{mh} \sum_{i=1}^{m} f_{\text{ker}}\left(\underbrace{\frac{t-\hat{y}^{(i)}}{h}}_{=:\,y}\right) dt$$

$$= \frac{1}{m} \sum_{i=1}^{m} \int_{\frac{\tau-\hat{y}^{(i)}}{h}}^{\infty} f_{\text{ker}}(y)\,dy$$

$$= \frac{1}{m} \sum_{i=1}^{m} Q\left(\frac{\tau-\hat{y}^{(i)}}{h}\right) \quad \text{(Gaussian kernel)}$$

Approximation via KDE

$$\widehat{\mathbb{P}}(\tilde{Y}=1)=\frac{1}{m}\sum_{i=1}^{m}Q\left(\frac{\tau-\hat{y}^{(i)}}{h}\right)$$

Remember: $\mathrm{DDP}:=\sum_{z\in\mathcal{Z}}|\mathbb{P}(\tilde{Y}=1|Z=z)-\mathbb{P}(\tilde{Y}=1)|$

Similarly, one can obtain:

$$\widehat{\mathbb{P}}(\tilde{Y}=1|Z=z)=\frac{1}{m_z}\sum_{i\in I_z}Q\left(\frac{\tau-\hat{y}^{(i)}}{h}\right)$$

m_z ← $|I_z|$

I_z ← $\{i:z^{(i)}=z\}$

Approximated DDP

$$\text{DDP} := \sum_{z \in \mathcal{Z}} |\mathbb{P}(\tilde{Y} = 1 | Z = z) - \mathbb{P}(\tilde{Y} = 1)|$$

$$\approx \sum_{z \in \mathcal{Z}} |\widehat{\mathbb{P}}(\tilde{Y} = 1 | Z = z) - \widehat{\mathbb{P}}(\tilde{Y} = 1)|$$

$$= \sum_{z \in \mathcal{Z}} \left| \frac{1}{m_z} \sum_{i \in I_z} Q\left(\frac{\tau - \hat{y}^{(i)}}{h}\right) - \frac{1}{m} \sum_{i=1}^{m} Q\left(\frac{\tau - \hat{y}^{(i)}}{h}\right) \right|$$

$$Q(x) \approx \begin{cases} \frac{1}{2} e^{-\frac{1}{2}x^2} & x \geq 0 \\ 1 - \frac{1}{2} e^{-\frac{1}{2}x^2} & x < 0 \end{cases}$$

Approximated DDP

$$\text{DDP} := \sum_{z \in \mathcal{Z}} \left| \mathbb{P}(\tilde{Y} = 1 | Z = z) - \mathbb{P}(\tilde{Y} = 1) \right|$$

$$\approx \sum_{z \in \mathcal{Z}} \left| \widehat{\mathbb{P}}(\tilde{Y} = 1 | Z = z) - \widehat{\mathbb{P}}(\tilde{Y} = 1) \right|$$

$$= \sum_{z \in \mathcal{Z}} \left| \frac{1}{m_z} \sum_{i \in I_z} Q\left(\frac{\tau - \hat{y}^{(i)}}{h}\right) - \frac{1}{m} \sum_{i=1}^{m} Q\left(\frac{\tau - \hat{y}^{(i)}}{h}\right) \right|$$

$$\frac{\tau - \hat{y}^{(i)}}{h} \geq 0 \quad \approx \sum_{z \in \mathcal{Z}} \left| \frac{1}{m_z} \sum_{i \in I_z} \frac{1}{2} e^{-\frac{(\tau - \hat{y}^{(i)})^2}{2h^2}} - \frac{1}{m} \sum_{i=1}^{m} \frac{1}{2} e^{-\frac{(\tau - \hat{y}^{(i)})^2}{2h^2}} \right|$$

Can express DDP in terms of samples (thus *w*)

KDE-based optimization

Cho-Hwang-Suh NeurIPS20

$$\min_{w} \frac{1-\lambda}{m}\sum_{i=1}^{m}\ell_{\mathsf{CE}}(y^{(i)},\hat{y}^{(i)}) + \frac{\lambda}{m}\cdot\sum_{z\in\mathcal{Z}}\left|\frac{m}{m_z}\sum_{i\in I_z}\frac{1}{2}e^{-\frac{(\tau-\hat{y}^{(i)})^2}{2h^2}} - \sum_{i=1}^{m}\frac{1}{2}e^{-\frac{(\tau-\hat{y}^{(i)})^2}{2h^2}}\right|$$

Algorithm: Gradient descent

Issues: How to deal with the absolute function?

How to choose the smoothing parameter *h*?

How to deal with the absolution func?

$$\min_{w} \frac{1-\lambda}{m} \sum_{i=1}^{m} \ell_{\mathsf{CE}}(y^{(i)}, \hat{y}^{(i)}) + \frac{\lambda}{m} \cdot \sum_{z \in \mathcal{Z}} \left| \frac{m}{m_z} \sum_{i \in I_z} \frac{1}{2} e^{-\frac{(\tau - \hat{y}^{(i)})^2}{2h^2}} - \sum_{i=1}^{m} \frac{1}{2} e^{-\frac{(\tau - \hat{y}^{(i)})^2}{2h^2}} \right|$$

Instead, one can employ Huber loss:

$$H_\delta(x) = \begin{cases} \frac{1}{2}x^2 & \text{if } |x| \leq \delta \\ \delta\left(|x| - \frac{1}{2}\delta\right) & \text{otherwise} \end{cases}$$

This enables us to readily obtain gradient.

How to choose the smoothing parameter *h*?

$$\min_{w} \frac{1-\lambda}{m}\sum_{i=1}^{m}\ell_{\mathsf{CE}}(y^{(i)},\hat{y}^{(i)}) + \frac{\lambda}{m}\cdot\sum_{z\in\mathcal{Z}} H_\delta\left(\frac{m}{m_z}\sum_{i\in I_z}\frac{1}{2}e^{-\frac{(\tau-\hat{y}^{(i)})^2}{2h^2}} - \sum_{i=1}^{m}\frac{1}{2}e^{-\frac{(\tau-\hat{y}^{(i)})^2}{2h^2}}\right)$$

Turns out:

There is a sweet spot for h that miminizes the mean square error of KDE estimate.

Advise us to find h^* that minimizes the MSE.

See [Cho-Hwang-Suh NeurIPS20] for details.

Extension to another fairness measure DEO

$$\text{DEO} := \sum_{y \in \mathcal{Y}} \sum_{z \in \mathcal{Z}} |\mathbb{P}(\tilde{Y} = 1 | Y = y, Z = z) - \mathbb{P}(\tilde{Y} = 1 | Y = y)|$$

$$\approx \sum_{y \in \mathcal{Y}} \sum_{z \in \mathcal{Z}} |\widehat{\mathbb{P}}(\tilde{Y} = 1 | Y = y, Z = z) - \widehat{\mathbb{P}}(\tilde{Y} = 1 | Y = y)|$$

$$\approx \sum_{y \in \mathcal{Y}} \sum_{z \in \mathcal{Z}} \left| \frac{1}{m_{yz}} \sum_{i \in I_{yz}} \frac{1}{2} e^{-\frac{(\tau - \hat{y}^{(i)})^2}{2h^2}} - \frac{1}{m_y} \sum_{i \in I_y} \frac{1}{2} e^{-\frac{(\tau - \hat{y}^{(i)})^2}{2h^2}} \right|$$

m_{yz}: $|I_{yz}|$

I_{yz}: $\{i : y^{(i)} = y, z^{(i)} = z\}$

Experiments

A benmark real dataset: **COMPAS**

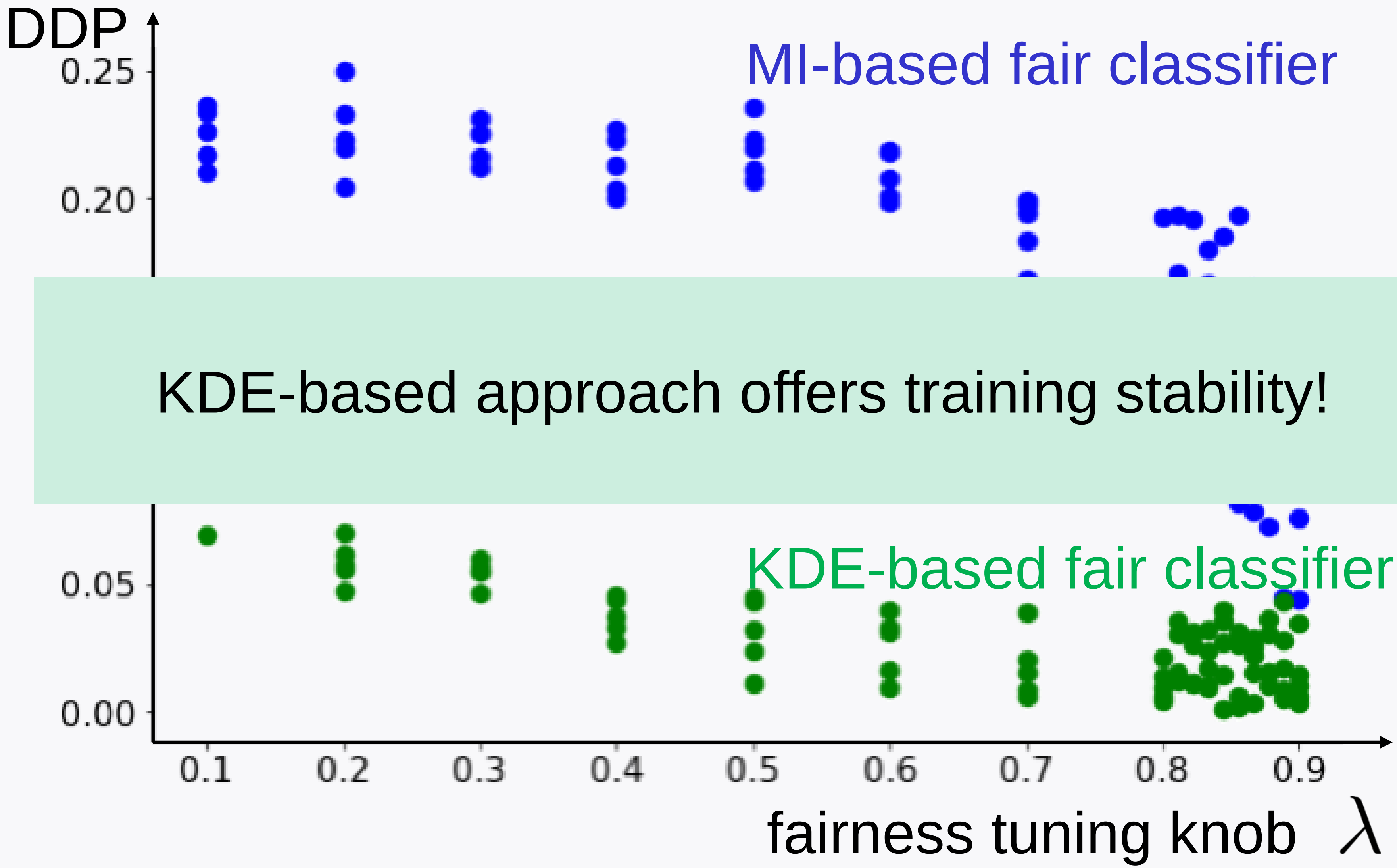
$$(x, z, y)$$

criminal records → x

black or white → z

reoffend or not in near future → y

Training stability?

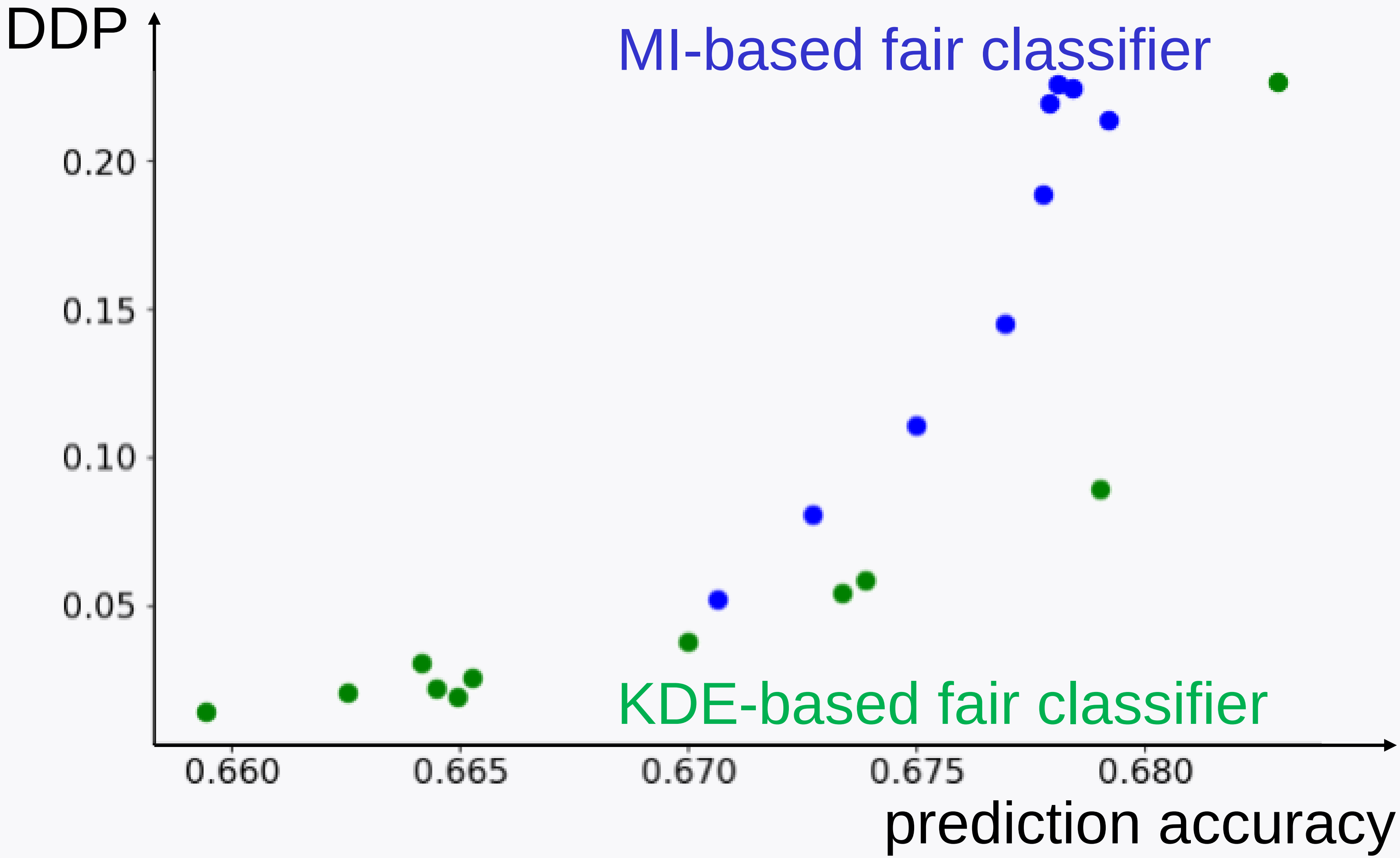


KDE-based approach offers training stability!

Accuracy vs DDP tradeoff

	Accuracy	DDP
Non-fair classifier	68.29 ± 0.44	0.2263 ± 0.0087
MI-based fair classifier	67.07 ± 0.85	0.0522 ± 0.0373
KDE-based fair classifier	67.00 ± 0.45	0.0374 ± 0.0079

Accuracy vs DDP tradeoff



Summary of Lectures 1 and 2

1. Explored fairness measures in fair classifiers.

2. Studied an MI-based fair classifier which yields a good tradeoff while suffering from training instability.

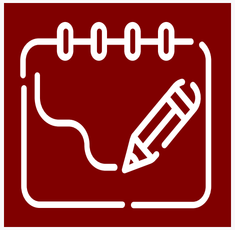
3. Investigated another fair classifer based on KDE, which addresses training instability.

Revisit: Five aspects for trustworthy AI

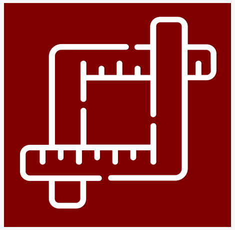
A recent progress: Roh-Lee-Whang-Suh, ICML20

fairness

robustness



explainability



value alignment

transparency

Look ahead

Will explore the recent work on fairness & robustness, and discuss some relative issues.

Reference

[1] J. Cho, G. Hwang and C. Suh. A fair classifier using mutual information. *IEEE International Syposium on Inofrmation Theory* (*ISIT),* 2020.

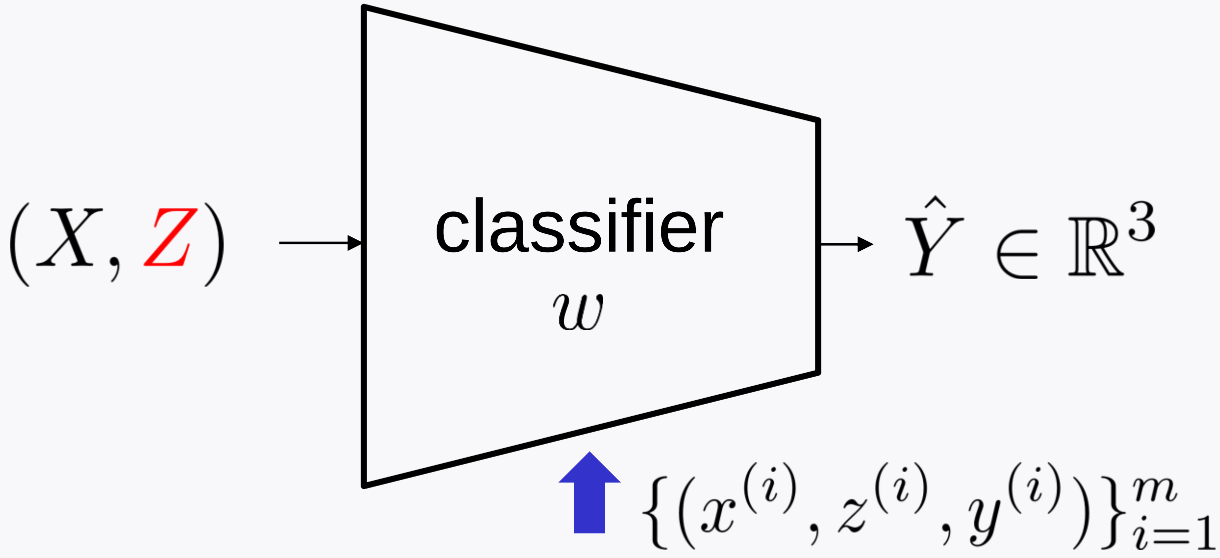
[2] J. Cho, G. Hwang and C. Suh. A fair classifier using kernel density estimation. *In Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.

[3] H. Jiang. Uniform convergence rates for kernel density estimation. *International Conference on Machine Learning (ICML)*, 2017.

[4] J. Angwin, J. Larson, S. Mattu, and L. Kirchner. Machine bias: There's software used across the country to 272 predict future criminals. And it's biased against blacks. *https://www.propublica.org/article/machine-bias-risk-assessments-incriminal-sentencing*, 2015.

backup

Extension to a non-binary classifier

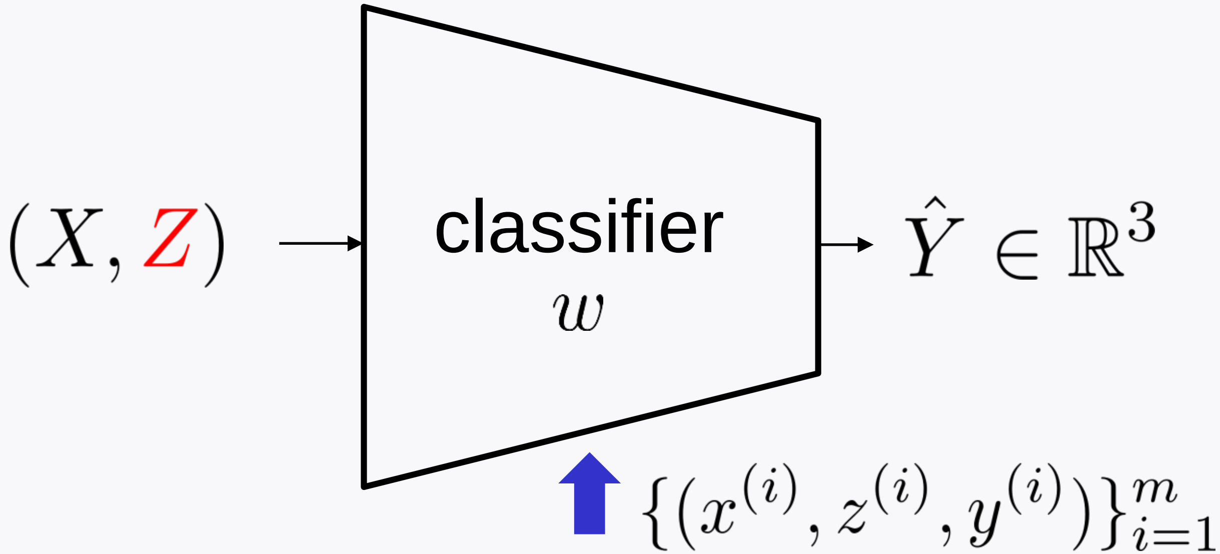


$$\text{DDP} := \sum_{y \in \mathcal{Y}} \sum_{z \in \mathcal{Z}} |\mathbb{P}(\tilde{Y} = y | Z = z) - \mathbb{P}(\tilde{Y} = y)|$$

$\mathbb{P}(\tilde{Y} = 1) = \mathbb{P}(\hat{Y}_1 > \hat{Y}_2, \hat{Y}_1 > \hat{Y}_3)$ (original hard decision)

Turns out: DDP is not differentiable under the original hard decision.

A proposed approach



$$\text{DDP} := \sum_{y \in \mathcal{Y}} \sum_{z \in \mathcal{Z}} |\mathbb{P}(\tilde{Y} = y | Z = z) - \mathbb{P}(\tilde{Y} = y)|$$

$$\mathbb{P}(\tilde{Y}_{\text{proposed}} = 1) = \mathbb{P}(\hat{Y}_1 > 0.5) \quad \mathbb{P}(\tilde{Y} = 1) = \mathbb{P}(\hat{Y}_1 > \hat{Y}_2, \hat{Y}_1 > \hat{Y}_3)$$

Turns out: DDP is differentiable in this case.