An Empirical Study on Learning Fairness Metrics for COMPAS Data with Human Supervision

Hanchen Wang, Nina Grgić-Hlača, Preethi Lahoti,

Krishna P. Gummadi, Adrian Weller

UNIVERSITY OF CAMBRIDGE

The Alan Turing Institute

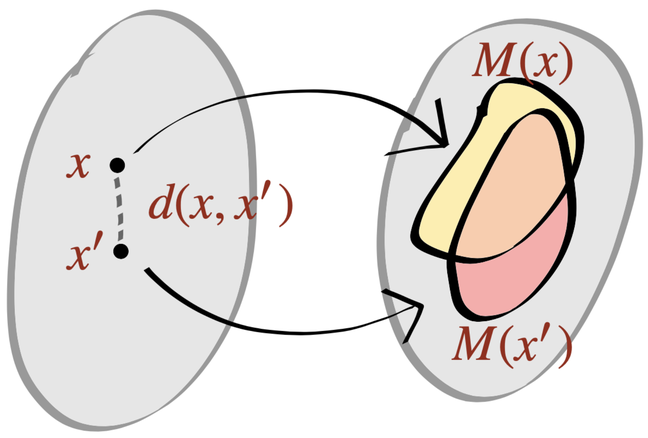
Algorithmic Fairness

Group Fairness: protected groups indicated by sensitive attributes are treated similarly to others

Individual Fairness: similar individuals should be received similar treatments ⇒ similarity metric

Previous work either incorporate collected human judgments as pairwise constraints in the learning objective, or use individual fairness as the objective as an implicit similarity metric in the optimization

Here we attempt to learn such similarity measure explicitly from human annotated data.



Fairness in Machine Learning, NeurIPS 2017 Tutorial by Moritz Hardt

Gathering Human Judgments

Survey Instrument we conducted an online survey in which participants are asked to estimate the likelihood of criminal recidivism of a fixed set of 200 defendants from the ProPublica dataset.

With showing information about the defendant's demographics and criminal history, participants need to respond to the following questions(Q1-3):

- How likely do you think it is that this person will commit another crime within 2 years? (5-point Likert scale)
- Do you think this person should be granted bail? (Yes/No)
- How confident are you in your answer about granting this person bail? (5-point Likert scale)

Procedure Participants are recruited through the online crowd-sourcing platform Prolific, with pre-screening options, we recruited 29 participants from the US who self-reported to have served on a jury.

Resources

Dataset

Paper

Statistics of Dataset

The final dataset consists of 20 participants, with an average criminal recidivism prediction accuracy of 62.4%, similar with previous reported.

Statistics of Dataset:

Recidivism Predictions(Q1) and Bail Decisions(Q2):

Bail Rate	Mean(%)	Max(%)	Min(%)
Extremely Unlikely	99.6	100	99.8
Unlikely	96.6	100	77.9
Neither	84.7	100	50.0
Likely	43.1	95.6	0
Extremely Likely	18.9	76.5	0

⇒ bail rate diverges especially when the defendants are considered to be more likely re-commit crimes

Bail Decisions(Q2) and Decision Confidence(Q3):

Acc	Overall	High Conf	Low Conf
All Judges	0.624±0.03	**0.649±0.06**	0.619±0.04
Judge 1	0.635	**0.828**	0.594
Judge 10	0.580	**0.750**	0.526
Judge 18	0.580	**0.714**	0.544
Judge 8	0.536	0.580	**0.621**
Judge 17	0.640	0.620	**0.660**

⇒ some respondents have better calibrated confidence assessments than others, more confident decisions result in higher accurate outcomes.

Metric Learning via Various Loss

Mahalanobis Distance: Distance between d-dimensional vector $\mathbf{x}$ and $\mathbf{x}'$ is defined as:

$$d_{\mathrm{M}}(\mathbf{x}, \mathbf{x}') = \sqrt{(\mathbf{x} - \mathbf{x}')^T \mathbf{M}^{-1} (\mathbf{x} - \mathbf{x}')}$$

parameterized by $\mathbf{M} \in \mathbb{S}_+^d$, where $\mathbb{S}_+^d$ is the set of real symmetric positive semi-definite $d \times d$ matrices, thus its satisfied the properties of a pseudo-distance(e.g. symmetry and triangular inequality)

The Mahalanobis matrix, $\mathbf{M}$, are derived based on the following methods respectively:

LSMM:
fully supervised, it attempts to minimize the distance between training instances and their neighbors of same class, while keeping instances of other classes out of the neighborhood. We treat the Likert scale ratings as categorical, same applied for MMC.

MMC:
weakly supervised, input is pairwise relative comparisons, it aims at maximizing the sum of pairwise distances between dissimilar pairs while keeping that of similar pairs relatively small. The learned metric can be constrained either in a diagonal form (weighted Euclidean) or as a full matrix

LSML: semi-supervised, it learns the metric from a set of triplet relative comparisons of the form "a and b are more similar than a and c". The triplet constraint set $\mathcal{C}$ is constructed as $\mathcal{C} = \{a, b, c | S_a \leq S_b + \sigma < S_c\}$, where $\sigma > 0$ and $S_{a,b,c}$ are the recidivism scores from our dataset. We adapted it by adding a trade-off coefficient $\alpha = 0.01$ on the logdet regularization term.

Effectiveness of Learned Metric

The learned metrics are evaluated in these terms:

Triplet Relative Comparison Loss:

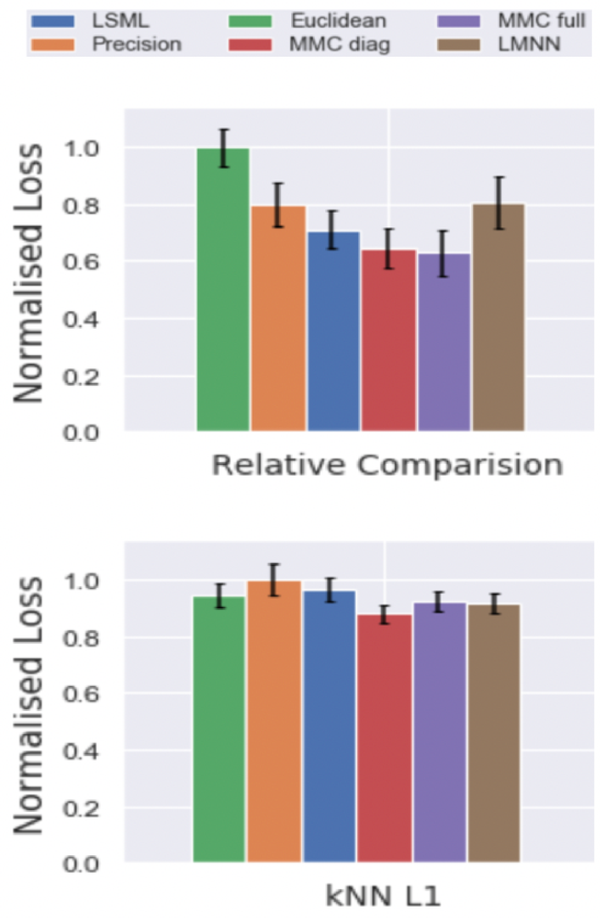
$$L\left(\mathbf{M}|\mathcal{C}_t\right) = \frac{H\left(d_{\mathrm{M}}\left(\mathbf{x}_a, \mathbf{x}_c\right) - d_{\mathrm{M}}\left(\mathbf{x}_a, \mathbf{x}_b\right)\right)}{|\mathcal{C}_t|}$$

kNN Lp Loss:

$$L_p\left(\mathbf{M}|\mathcal{C}_t\right) = \sum_{\mathbf{x}_i \in \mathcal{C}_t} |\hat{y}_i - y_i| = \sum_{\mathbf{x}_i \in \mathcal{C}_t} |\sum_j w_{ij} y_j - y_j|^p$$

where $w_{ij} \propto 1/d_{\mathrm{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right)$, $p = 1$ or 2

Each metric (LMNN, MMC, LSML) is trained on 140 inputs and evaluated on 60 inputs. These 200 inputs were randomly selected from generated entries. In the evaluation, we repeat this process 10 times and report the average results.



The learned metrics slightly outperform the Euclidean and Precision in terms of kNN L1, for triplet relative comparison loss, which incorporates the relative order between nearby ratings rather than treating ratings as categorical, the learned metrics have significantly outperformed the baseline.

we evaluate the sensitivity of our adapted LSML method to σ and its own training setting σ_t, which controls the minimum required distance between inputs b and c from the triple entries $\{a, b, c\}$.

σ_t	Euclidean	Ours($\sigma = 0$)	Ours($\sigma = 2$)
0	0.40±0.037	0.39± 0.041	**0.38± 0.035**
2	0.40±0.027	0.39± 0.032	**0.37± 0.037**
4	0.35±0.031	0.31± 0.045	**0.30± 0.034**
6	0.31±0.033	**0.29± 0.060**	**0.29± 0.060**

Summary & future work

- Deep Metric Learning, e.g. Siamese Network