Effect of Feature Hashing on Fair Classification

Ritik Dutta, Varun Gohil, Atishay Jain
IIT Gandhinagar

Background

- Algorithmic fairness - reduce bias in automated decision making
- Example - COMPAS, an algorithm used in US to decide prison sentences; was found to be racially biased
- Trade-off needs to be made between accuracy gain obtained by using sensitive information as part of statistical model and between protecting sensitive information

Definitions of Fairness

- Fairness definitions are almost always defined w.r.t. protected (sensitive) groups
- Multiple, often incompatible definitions of fairness
- Some examples -
  - Demographic Parity - the proportion of each segment of a protected class (e.g. gender) should receive the positive outcome at equal rates.
  - Equalised Odds - A predictor $\hat{Y}$ satisfies equalized odds with respect to a protected attribute A and outcome Y if $\hat{Y}$ and A are independent conditional on Y

Equal Opportunity (EOp):

$$\mathbb{P}\{f(\boldsymbol{x}, s){>}0 \mid s{=}1, y{=}\diamond 1\} = \ldots = \mathbb{P}\{f(\boldsymbol{x}, s){>}0 \mid s{=}k, y{=}\diamond 1\}$$

Difference of EOp (DEOp):

$$\mathrm{DEOp}^{\diamond} = \sum_{t \in \mathcal{S}} \left| \mathbb{P}\{\hat{y} = y | s{=}t, y{=}\diamond 1\} - \frac{1}{|\mathcal{S}|} \sum_{t' \in \mathcal{S}} \mathbb{P}\{\hat{y} = y | s{=}t', y{=}\diamond 1\} \right|$$

Feature Hashing

- Instead of 1-1 mapping of features to locations in feature vector, use hash function to determine feature's location in a vector of lower dimension
- Generally used to reduce high-dimensional, sparse data to significantly compress size

$$h \in [n] \to [m]$$

$$\sigma_1, \ldots, \sigma_n \in \{-1, 1\}$$

- Then, for every $x \in R^n$, define $f(x) \in R^m$ by

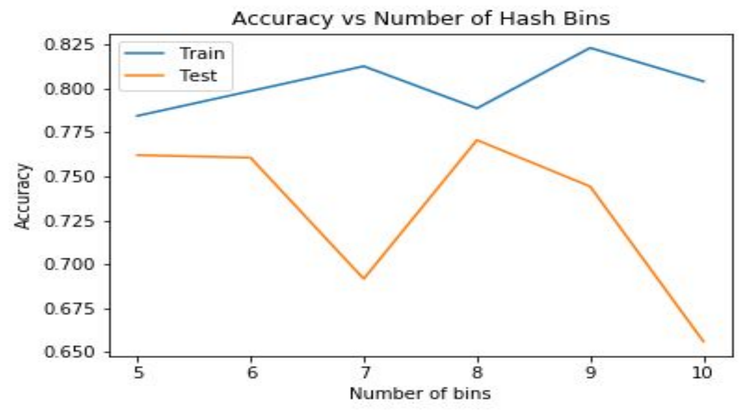
$$(f(x))_i = \sum_{j:h(j)=i} \sigma_i x_j$$

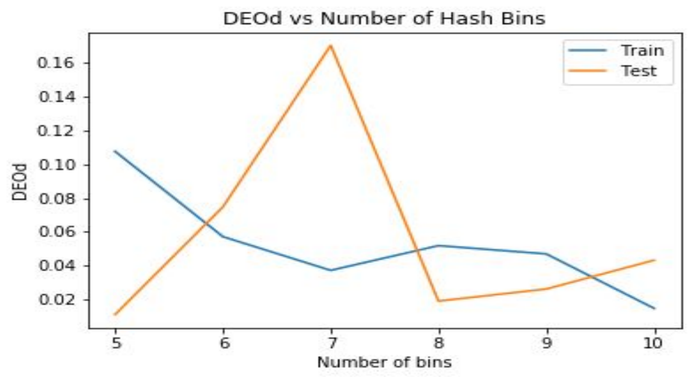
Combining Feature Hashing and Fair Classification

- Learning new representations of data to reduce correlation with sensitive attributes
- We explore the possibility of using feature hashing as a method for learning new representation of data
- 2 approaches tested:
  - First predict sensitive features using non-sensitive features, then use all features to predict the target feature
  - Directly predict the target using only non-sensitive features
- We applied feature hashing on all non-sensitive features
- Compared accuracies and fairness metrics between feature-hashed and original data
- Implemented a Multi-Task Learning Model but due to slow convergence, shifted to SVM with RBF-kernel

Metric	Without Hashing	With Hashing
Test DEOd	0.10315	0.01904
Train DEOd	0.07200	0.05187
Test Accuracy	83.152	77.065
Train Accuracy	87.346	78.869

Table 1: Results (Lower is better for DEOd)





References

1. Philip Adler, Casey Falk, Sorelle A. Friedler, Tionney Nix, Gabriel Rybeck, Carlos Scheidegger, Brandon Smith, and Suresh Venkatasubramanian. 2018. Auditing Black-box Models for Indirect Influence. Knowl. Inf. Syst. 54, 1 (Jan. 2018), 95–122. https://doi.org/10.1007/s10115- 017- 1116- 3

2. Wenlin Chen, James T. Wilson, Stephen Tyree, Kilian Q. Weinberger, and Yixin Chen. 2015. Compressing Neural Networks with the Hashing Trick. In Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37 (ICML'15). JMLR.org, 2285–229

3. Luca Oneto, Michele Doninini, Amon Elders, and Massimiliano Pontil. 2019. Taking Advantage of Multitask Learning for Fair Classification. In Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society (AIES '19). ACM, New York, NY, USA, 227–237.

4. Kilian Q. Weinberger, Anirban Dasgupta, Josh Attenberg, John Langford, and Alexander J. Smola. 2009. Feature Hashing for Large Scale Multitask Learning. CoRR abs/0902.2206 (2009). arXiv:0902.2206