Investigating Models for Mitigating Bias

Sapna Kumari, Joe O'Connor, Archer Wang
Massachusetts Institute of Technology
Cambridge, MA 02139

December 11, 2019

Abstract

In machine learning research, mitigating bias in modeling is an area of significant interest. In the real world, the data collected is often biased around certain sensitive attributes, such as race and gender, and the trained models tend to reflect these biases, resulting in unfair predictions. Several definitions have been proposed by researchers to quantify algorithmic fairness, and various algorithmic techniques have been introduced to achieve fairness with respect to proposed definitions. In this paper, we will be evaluating the effectiveness of adversarial debiasing technique introduced by Zhang et.al [1], by comparing it to some of the naive approaches such as balancing the dataset. In particular, we will be assessing the trade-offs between accuracy and fairness of various approaches and discuss how well the adversarial debiasing technique does in general.

1 Introduction

Machine learning is about finding patterns in data and using that information to create models and make accurate predictions. Especially nowadays, training effective machine learning models hinges on collecting *quality* data, and we constantly use these models in making decisions such as which applicants to hire for job, which convicted criminals to detain or release, or who to give loan to. However, oftentimes such decision making models suffers from biases which could arise from biases in the datasets or biases in the algorithm. For instance, using unbalanced data can create biases against underrepresented groups.

Training a machine learning model without accounting for biases present in the data will result in a predictions that reflect the biases with respect to the sensitive attributes. Bias persists even after removing the sensitive attributes from the set of input features due to presence of proxy variables, i.e., variables that are correlated strongly with the sensitive attributes. Removing proxy variables in addition to the sensitive attributes, may leave the model with too little information to make accurate predictions. Thus, this is a challenging problem and explains

why reducing bias in machine learning models is an area with many practical uses that draws significant attention.

The debiasing model that we will largely be focusing on in this paper is a GAN-like model developed by Zhang, Lemoine, and Mitchell [1]. Their model was quite accurate when tested on the UCI Adult dataset. We aim to reproduce these results and compare them to the results of simpler methods in order to evaluate whether adversarial debiasing is the best approach for decreasing bias while maintaining accuracy.

In Section 2, we investigate how a machine learning model is able to debias with respect to a certain attribute. We define what fairness means in the context of models, and quantitatively give several definitions. We then discuss ways to implement this into our models and focus on the adversarial model described in [1]. In Section 3, we train six different models on each of the two datasets COMPAS and UCI Adult, evaluating and reporting the accuracy and fairness for each scenario. In Section 4, we give a concluding discussion about our results.

2 Background

2.1 Definitions of Fairness

Before describing model architecture, we need to explain what it means for a model to be fair. Fairness definitions fall under two different types: individual fairness and group fairness. Individual fairness means assigning similar outcome to similar individuals regardless of what demographic group they belong to. We will be focusing on group fairness, which involves equalizing some statistic such as false positive rate across all demographic groups.

For our model, we want to predict Y from input X with protected variable Z. Given tuples (X, Y, Z), a predictor $\hat{Y} = f(X)$ can be constructed. We quantitatively measure fairness of our predictor $\hat{Y}$ via the following definitions.

Definition 2.1. $\hat{Y}$ satisfies *demographic parity* if $\hat{Y}$ and Z are independent of each other. In other words,

$$P(\hat{Y} = \hat{y}) = P(\hat{Y} = \hat{y} | Z = z).$$

for all $\hat{y}$ and z.

Definition 2.2. $\hat{Y}$ satisfies *equality of opportunity* with respect to a *certain* class y if $\hat{Y}$ and Z are conditionally independent given $Y = y$.

$$P(\hat{Y} = \hat{y} | Y = y) = P(\hat{Y} = \hat{y} | Z = z, Y = y).$$

for all $\hat{y}$ and z.

Definition 2.3. $\hat{Y}$ satisfies *equality of odds* if $\hat{Y}$ and Z are conditionally independent for all Y. Thus,

$$P(\hat{Y} = \hat{y} | Y = y) = P(\hat{Y} = \hat{y} | Y = y, Z = z).$$

for all $\hat{y}$, y, and z.

Remark. Note that this is a stronger condition than equality of opportunity.

To understand why the above notions of fairness are useful, we can think of demographic parity as mitigating bias in the data, and think of equality of odds as mitigating bias in the prediction algorithm.

2.2 Algorithms for Achieving Fairness

Several methods have been tested that produce predictors satisfying one or more of the fairness conditions. For example, in 2019, Wang et. al. [2] posited causal models and used counterfactual decisions for their model. Out of the many possible ways of constructing a model to achieve fairness, we will be focusing on the method of adversarial debiasing. We will turn to supervised deep learning to address this issue. We aim to assess the adversarial debiasing model described in [1].

2.3 The Adversarial Debiasing Model

We explain the architecture of adversarial debiasing model described in [1]. We will implement one model that satisfies demographic parity and the other that satisfies equality of odds. As stated before, we have tuples (X, Y, Z) consisting of input variable X, output variable Y, and protected variable Z. The adversarial model consists of two parts: the *predictor* and the *adversary*.

For our models on both datasets, the variables Y and Z will be both binary-valued. The first part of the model is the *adversary*. The *predictor* is a logistic regression model having weights W, taking the input x and outputting $\hat{y}$, given by

$$\hat{y} = \sigma(Wx + b_1).$$

Then, the loss function $L_P(\hat{y}, y)$ is computed, where y is the ground truth outcome. The loss function that we will use for both parts of the model is binary cross-entropy loss, which is

$$L(\hat{y}, y) = -y \log(\hat{y}) - (1 - y) \log(1 - \hat{y}).$$

The second part of the model is the *adversary*, much like a discriminator in a typical GAN. If we will make our predictor satisfy the *equality of odds* fairness definition, then the output $\hat{y}$ from the predictor is used along with y to form a vector $v = [s, sy, s(1 - y)]$, where

$$s = \sigma((1 + |c|)\sigma^{-1}(\hat{y})),$$

from which c is an adjustable parameter. If we will make our predictor satisfy the *demographic parity* definition, then we simply set v to be $[s]$. The vector v is then used as an input into the discriminator, and the discriminator tries to predict the protected variable value $\hat{z}$, which is given by

$$\hat{z} = \sigma(Uv + b_2)$$

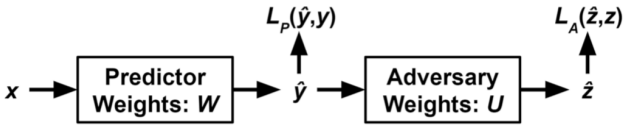
for weights U. Then, the loss function $L_A(\hat{z}, z)$ is computed. The entire model architecture is summarized in Figure 1(a).

Remark. The magnitude of c more or less determines how "discrete" $\hat{y}$ should be. When c goes to infinity, s simply becomes 0 or 1 depending on whether $\hat{y}$ is greater than $\frac{1}{2}$ or not.

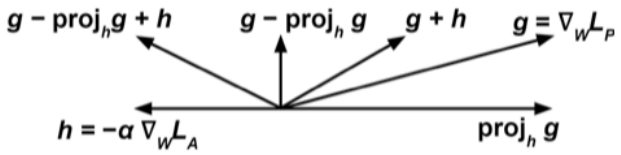
Next, we delve into training the model. When training, the updates consists of two parts, one to the weights of the adversary and one to the weights of the predictor. We update the weights U of the adversary via traditional gradient descent. We find $\nabla_U L_A$ and update the weights accordingly. We then update the weights W of the predictor according to the following rule:

$$\nabla_W L_P - \text{proj}_{\nabla_W L_A} \nabla_W L_P - \alpha \nabla_W L_A.$$

This is different from the update rule to the adversary party. As seen in Figure 1(b), we minus the projection term in the middle to prevent the predictor weights from moving in the direction that helps the adversary decrease its loss.



(a) Architecture of Adversarial Model



(b) Projection Diagram for Updating Predictor Weights

Figure 1

Overall, the objective of training is to maximize the ability of predictor to predict the outcome while minimizing the ability of discriminator to infer the sensitive attribute from the output of the first model.

3 Experiments

We study the fairness-accuracy trade-off of algorithmic decisions on two real-world datasets: UCI Adult and COMPAS Recidivism. In all cases, we remove the sensitive attribute (gender in UCI adult and race in COMPAS) from our data, and we also balance the dataset with respect to our predicted attribute (income in UCI adult and reoffend in COMPAS). Beyond that, we try combinations of the following modifications:

- Removing proxy variables: We use normalized mutual information (MI) to identify the attributes that give significant information about sensitive attribute (e.g., marital status, captial gain, relationship status give informtion about gender in UCI adult), and we remove them from the data during training.

- Balancing the dataset: In addition to balancing with respect to the predicted attribute, we use down sampling to make sure that the distribution over the outcome values is the same for both men and women.

As for reporting results, we compute the following fairness metrics:

- Demographic Parity (DP) Difference: Computed as the difference of the rate of positive outcomes received by the unprivileged group to the privileged group. In UCI adult, unprivileged group is females and privileged group is males, whereas in COMPAS, unprivileged group is African-Americans and privileged group is Caucasians.

- False Positive Rate (FPR) Difference: Computed as the difference of the false positive rate of unprivileged group to the privileged group. False positive rate is defined as fraction of people with true negative outcome who are assigned a positive outcome by the predictor.

- False Negative Rate (FNR) Difference: Computed as the difference of the false negative rate of unprivileged group to the privileged group. False negative rate is defined as fraction of people with true positive outcome who are assigned a negative outcome by the predictor.

- Average Odds (AO) Difference: Computed as average difference of false positive rate and true positive rate (1 - false negative rate) between unprivileged and privileged groups. AO difference of 0 implies equality of odds since then both false positive rate and false negative rate will be equal for each of two groups.

3.1 UCI Adult Dataset

The dataset consists of continuous features (ex. age, hours of work per week, years of education) and categorical features (ex. marital status, native country). A more comprehensive description is shown below. The purpose of the dataset is to predict whether a person's income is over \$50k a year.

Feature	Type	Description
age	Cont	Age of the individual
capital_gain	Cont	Capital gains recorded
capital_loss	Cont	Capital losses recorded
education_num	Cont	Highest education level (numerical form)
fnlwgt	Cont	# of people census takers believe that observation represents
hours_per_week	Cont	Hours worked per week
education	Cat	Highest level of education achieved
income	Cat	Whether individual makes > $50K annually
marital_status	Cat	Marital status
native_country	Cat	Country of origin
occupation	Cat	Occupation
race	Cat	White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black
relationship	Cat	Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried
sex	Cat	Female, Male
workclass	Cat	Employer type

(a) UCI Adult Dataset Description of Features

3.1.1 Preprocessing and Data Splitting

Categorical features such as education, marital status, and country of origin are represented as one-hot encoded vectors. Continuous features such as hours of work per week, highest education level are scaled between 0 and 1 using min-max scaling. We randomly split the data to have 75% of it to be trained and the remaining 25% to be tested by our models.

3.1.2 Results

We train six different variations of logistic regression model, with some combination of balancing the datasets, removing the proxy variables and adversarial debiasing model, and report accuracy and various fairness metrics on test dataset for each model.

As shown in table below, we find that we achieve the highest accuracy of 79.5% on the baseline model which is reasonable since addition of fairness constraints could only lower accuracy. Removing proxy attributes leads to lower average odds difference but also leads to lower accuracy as proxy variables may be useful indicators of income. Balancing the dataset and removing proxy variables leads to fairest outcomes but also the lowest accuracy. Thus, we see a trade-off between accuracy and fairness. Adversarial models, implemented with both demographic parity constraint and equality of odds constraint's, led to poor fairness results due to issues with hyperparameter tuning. We experimented with various values of α parameter that controls the amount with which predictor tries to hurt adversary's loss, but with no improvement in fairness metrics. Increasing the α parameter to a very large value only led to poorer prediction accuracy without any improvement in the fairness which was bit strange.

UCI Adult: Models vs Accuracy-Fairness

	Accuracy (%)	DP Difference	FPR Difference	FNR Difference	AO Difference
Baseline	79.5	-0.378	-0.245	0.199	-0.222
No Proxies	75.3	-0.230	-0.120	0.040	-0.082
Balanced	77.8	-0.165	-0.248	0.089	-0.168
Balanced & No Proxies	73.9	-0.011	-0.091	-0.064	-0.013
Adversarial ($\hat{y}$)	76.2	-0.598	-0.407	0.617	-0.512
Adversarial ($\hat{y}$, y)	76.2	-0.584	-0.407	0.574	-0.490

3.2 COMPAS Recidivism Risk Score Dataset

COMPAS is a propriety software used by U.S. courts nationwide to assess whether a criminal defendant would be a recidivist (i.e., will repeat offend) to decide whether to release or detain the defendant. The dataset consists of criminal records of several convicted criminal with features such as prior crime count, charge description, charge degree, length of stay in jail, race, gender etc. and binary label of whether the convict reactivated (label 1) or not (label 0).

3.2.1 Preprocessing and Data Splitting

Our input to the model will be a vector that is the result of concatenating several vectors corresponding to features from the COMPAS dataset. We treated age, charge degree, descriptions as categorical variables, and used one-hot encoding to represent them. For all continuous variables such as time spent in jail, number of priors, we applied min-max scaling to restrict their range between 0 and 1. We randomly split the data to have 75% of it to be trained and the remaining 25% to be tested by our adversarial model.

3.2.2 Results

With COMPAS, we trained the same six variants as with UCI Adult. Strangely, we actually achieved best accuracy with the adversarial models (63%) compared to 60.7% on the baseline. The best results on fairness come from the model with no proxy variables, with similar results coming from balancing the data. The reason that the accuracies are lower on this dataset than on the UCI Adult dataset, is that here we are trying to solve a more challenging problem; whether a criminal reoffends depends on a wide variety of factors that simply cannot be captured in data, so making predictions in this setting is very difficult.

COMPAS Recidivism: Models vs Fairness-Accuracy					
	Accuracy (%)	DP Difference	FPR Difference	FNR Difference	AO Difference
Baseline	60.7	0.166	0.148	-0.136	0.142
No Proxies	54.8	0.043	0.047	-0.020	0.034
Balanced	56.2	0.063	0.089	-0.036	0.062
Balanced & No Proxies	59.2	0.128	0.133	-0.121	0.127
Adversarial ($\hat{y}$)	63.0	0.147	0.120	-0.115	0.117
Adversarial ($\hat{y}$, y)	61.0	0.180	0.145	-0.168	0.157

4 Discussion

In this paper, we assessed the performance of an adversarial debiasing technique on the COMPAS Recidivism dataset and the UCI Adult Dataset. After preprocessing and splitting our data, we trained the model described in [1]. Based on the above results, we conclude that in fact adversarial debiasing is not superior to simpler conventional approaches. Not only can we achieve good performance with the simpler methods, but achieving good performance with the adversarial model is unreasonably challenging, which makes it impractical as a standard method. That being said, debiasing machine learning algorithms in general is not yet a solved problem, and there is plenty more research to be done.

5 Acknowledgement

We would like to thank our TA Hongge Chen, PhD student of EECS at MIT, for giving tremendous insight for our project and helping us understand research papers related to our project. We would also like to thank David Sontag for his input on our project and the entire 6.867 staff for teaching machine learning, providing handouts, and holding office hours this semester.

6 Work Contribution

Sapna Kumari worked on implementing the logistic regression model and the adversarial debiasing model. Joe O'Connor worked on implementing the adversarial debiasing model and the proxy variables / balancing as well as preprocessing the data. Archer Wang worked with creating the tables and graphs from training the models on the data.

References

[1] Brian H Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating Unwanted Biases with Adversarial Learning. (ii), jan 2018. URL http://arxiv.org/abs/1703.06114.

[2] Yixin Wang, Dhanya Sridhar, David M. Blei. Equal Opportunity and Affirmative Action via Counterfactual Predictions *arXiv preprint arXiv:1905.10870*, 2019.