Algorithmic Fairness Final Report

MORPH 2020

Zoya Goel

Contents

1 Introduction **2**

2 Group Fairness **2**

2.1 Demographic Parity 2

2.2 Equalized Odds 3

2.3 Predictive Rate Parity 3

3 Individual Fairness **3**

4 Counterfactual Fairness **3**

5 Fairness in AI **4**

5.1 Gender Shades 4

5.2 Lipstick on a Pig 4

6 Our Project **5**

6.1 Background 5

6.1.1 AIFairness 360 5

6.1.2 FairLearn 6

6.2 Objective and Methodology 6

6.3 Findings 6

7 Takeaways **7**

References **8**

1 Introduction

Artificial Intelligence is becoming increasingly prominent in our society. It has presence in all sorts of sectors, from marketing to criminal justice. However, this technology is relatively new, so there are lots of problems with AI that have yet to be adequately addressed. One problem is the issue of algorithmic fairness. When ML algorithms learn, they train on data taken from the world around them, which is full of bias. Also, the creators of the algorithms can be biased. As a result, these machines can end up treating certain demographics more unfairly than others, exacerbating already present disparities. With AI's ever-increasing influence, as well as it's capacity to further harm marginalized groups, researchers must take action in regards to this problem. [1]

2 Group Fairness

Group fairness refers to whether a group in its entirety is treated fairly. Three types of group fairness are demographic parity, accuracy parity, and predictive rate parity. A common way to demonstrate these different types of group fairness is through an example, in which an algorithm decides whether to hire an employee. [9]

- $X \in R^d$: X refers to the qualifications the job applicant has.
- $A \in 0, 1$: A is a binary sensitive attribute (is the applicant in a minority subgroup or not?)
- $C = c(X, A) \in 0, 1$ C is a binary classifier which determines whether the applicant gets the job.
  - This is based on a score $R = r(x, a) \in [0, 1]$
- $Y \in 0, 1$: Y represents the true ability of the applicant.
- (X, A, Y) D: D is the distribution that generates X, A, and Y.
- $P_0[c] := P[c|A = 0]$: P_0 represents the outcome when the applicant is a minority.
- When no fairness constraints are applied, the most accurate result is achieved when $C(X, S) = Y \forall (X, S, Y)$ D.

2.1 Demographic Parity

In Demographic Parity, the acceptance rates for each subgroup must be equal. In this example, it is as:

$$\text{C is independent of A: } P_0[C = 1]/P_1[C = 1] \geq 1 - \epsilon \tag{1}$$

Demographic parity is useful, because it helps employers fulfill basic selection rules that prevent adverse impact towards minorities, such as the four-fifths rule. There are also experts who argue that demographic parity can be vital for minorities gaining a foothold in the workforce. However, demographic parity can also hurt minorities, because if an employer dislikes minorities, they will purposefully select people who will under-perform. As a result, it will appear that the minority group is not as productive, which will hurt minorities that want to apply in the future. Demographic parity also does not address correlations between an applicant's abilities and their subgroup. [9]

2.2 Equalized Odds

With Equalized odds, the rate of positives for each demographic is equal

$$\text{C is independent of A, conditional on Y: } P_0[C = r|Y = y] = P_1[C = r|Y = y] \forall r, y \tag{2}$$

There is also the idea of accuracy parity, where the rate of false outcomes is equal across demographics. However, operating on accuracy parity alone is problematic, because accuracy parity might be fulfilled, but the types of false outcomes are different across demographics (e.g: The minority group experiences more false negatives, and the majority experiences more false positives). A more useful idea is Equality of Opportunity, where the rates at which the most qualified people are hired from each group is the same. Equalized odds is useful, because it rewards hard work. However, it might not be enough to help fix larger societal disparities. [9]

2.3 Predictive Rate Parity

In Predictive Rate Parity, both Positive Rate Parity and Negative Rate Parity are satisfied - Positive and Negative rates are equal across all demographics.

$$\text{Y is independent of A, conditional on C: } P_0[Y = y|C = c] = P_1[Y = y|C = c] \forall y, c \in 0, 1 \tag{3}$$

Predictive Rate Parity helps satisfy Equalized Odds, and acceptance will accurately reflect an employee's ability. However, it might not be enough to fix disparities between the majority and minority group at large. [9]

3 Individual Fairness

Individual fairness, unlike group fairness, is centered around the treatment of individuals.[9] Ultimately, similar individuals should be treated similarly, and this is exemplified by the Lipschitz condition.[6]

Definition 3.1 (Lipschitz Condition)**.** The distance between two individuals x and y, $d(x, y)$, must be greater than or equal to the distance between x and y when they are mapped to a distribution, $d(M(x), M(y))$.

$$D(M(x), M(y)) \leq d(x, y) \tag{4}$$

Individual fairness is helpful because people tend to care about individuals more than groups at large, and because it is a very precise way of looking at fairness. However, figuring out how to define "distance" in terms of the Lipschitz condition in real life is a challenge.[9]

4 Counterfactual Fairness

Counterfactual fairness deals with the causes of disparities. When put in practice, a sensitive trait would be replaced, so everything that happened downstream as a cause of that sensitive trait would

end up changing. If counterfactual fairness was employed in the hiring example, one would change a sensitive attribute, such as race. As a result, downstream events, such as education, might change. A classifer could then decide whether to hire the applicant or not based on the counterfactual [9]. Here is how it is represented, using the previous hiring example:

$$P[C_{A\leftarrow 0}] = P[C_{A\leftarrow 1}] = c|X, A = a] \tag{5}$$

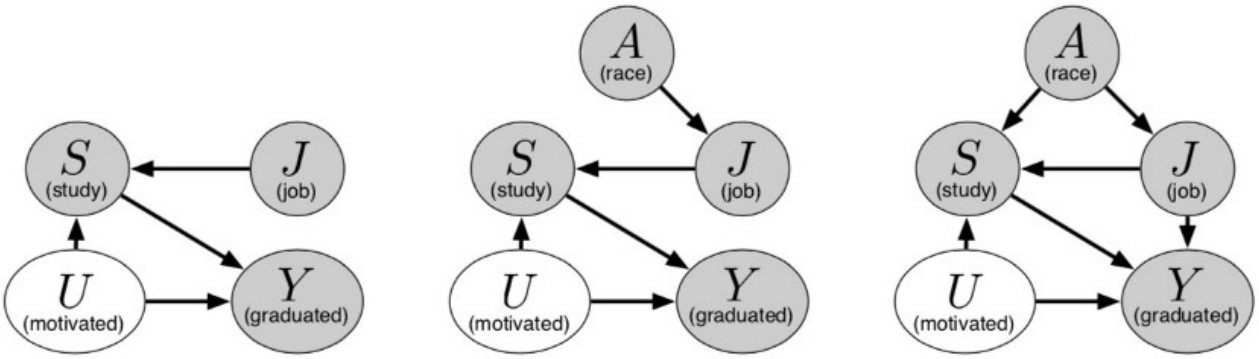

Figure 1: This is an example of a causal graph in a hiring scenario [9]

5 Fairness in AI

5.1 Gender Shades

One research discovery that stood out to me was Joy Buolamwini and Timnit Gebru's work with facial recognition algorithms. She tested several leading facial recognition algorithms that classify gender, and evaluated their performance on light-skinned males, light-skinned females, dark-skinned males, and dark-skinned females. They found that these gender classifiers tended to perform better on light skinned people and men, and worse on dark skinned people and women. What I found unique about their work was how they tested performance on intersectional groups of people. Buolamwini and Gebru's work should convince other researchers that analyzing algorithmic fairness in the context of intersectionality could be extremely useful. What I most value in their work, however, is how they show that today's facial recognition algorithms have enormous room for improvement. Facial recognition algorithms are being used for crucial purposes, such as identifying suspects in a criminal case. If these algorithms are extremely inaccurate for certain demographic, that is cause for massive concern. They also show how the problem is not just limited to the algorithms they tested, as they found that widely used facial recognition datasets are overwhelmingly white and male.[4]

5.2 Lipstick on a Pig

Another research discovery that stood out to me was Hila Gonen and Yoav Goldberg's work with word embeddings. Their paper, Lipstick on a Pig, shows how current debiasing methods for word

embeddings do not truly remove gender bias. They test two existing debiasing methods, and they found that male and female associated words still cluster together in the embedding, despite them being equidistant from explicitly gendered words. This paper was impactful to me because it changed my perception on debiasing word embeddings. I had read Bolukbasi et al. [3], and I felt like their definition of gender bias in a word made sense. However, Gonen and Yoav's work show how that definition is inadequate. [7]

6 Our Project

6.1 Background

As a part of our project, we sought out reputable fairness toolkits that can analyze the bias present in an algorithm, as well as mitigate it. We found three toolkits: IBM's AIFairness 360, Microsoft's Fairlearn, and Google's What-If Tool. However, we only ended up using AIFairness360 and Fairlearn. The What-If tool focused on counterfactual fairness[8], while the other tools focused on group fairness[2][5]. Also, there was not an obvious way to analyze bias in the dataset through the What-If tool. For those reasons, we felt like it would have been difficult to compare it to the other toolkits. We exclusively used the demos Microsoft and IBM provided, due to their ease of use, and since they all analyzed the same dataset. The dataset we used was UCI's Income Census dataset, which is used to train classifiers to predict whether a person has an income over 50,000 dollars [2][5].

6.1.1 AIFairness 360

An IBM-produced toolkit, AI Fairness 360 [2] provides numerous methods both for bias-detection and bias-mitigation. For assessing existing bias in the dataset, there are several options:

1. Statistical Parity Difference, which is computed as the difference of the rate of favorable outcomes received by the unprivileged group to the privileged group.
2. Equal Opportunity Difference, which is computed as the difference of true positive rates between the unprivileged and the privileged groups. The true positive rate is the ratio of true positives to the total number of actual positives for a given group.
3. Average Odds Difference, computed as average difference of false positive rate (false positives / negatives) and true positive rate (true positives / positives) between unprivileged and privileged groups.
4. Disparate Impact, computed as the ratio of rate of favorable outcome for the unprivileged group to that of the privileged group.
5. Theil Index, computed as the generalized entropy of benefit for all individuals in the dataset, with alpha = 1. It measures the inequality in benefit allocation for individuals.

There are methods to mitigate bias during diiferent stages of implementing the model:

1. Pre-processing: distributes and adjusts the weights after training, but before classification, in order to ensure fairness and give more weight to specific attributes.

2. In-processing: learns a classifier that maximizes prediction accuracy and simultaneously reduces an adversary's ability to determine the protected attribute from the predictions.
3. Post-processing - uses a linear program to decide whether to change the classification labels for an instance, provides favorable outcomes to unprivileged groups and unfavorable outcomes to privileged groups to equalize them.

6.1.2 FairLearn

Fairlearn is a fairness toolkit developed by Microsoft which focuses on quality-of-service and allocative harms. A major advantage of Fairlearn is it's interactive visualization dashboard, which makes understanding the algorithm's performance easier. It's features include:

- Showing how a group can be negatively effected by an algorithm
- Comparing the fairness of multiple models
- Evaluating either classification or regression models
- Using fairness metrics such as demographic parity, equalized odds, and worst-case accuracy rate

It also has algorithms that can mitigate unfairness in a model. Fairlearn has one post-processing model, and two reduction models. Their post-processing algorithms will alter the output in regards to certain fairness metrics, and their reduction algorithm retrains the model with the constraints implied by the fairness requirements [5].

6.2 Objective and Methodology

Our objective is to compare bias detection tools in the Fairlearn toolkit, as well as the AIFairness 360 toolkit. We used their toolkit demos on UCI's Income Census dataset. The toolkit assesed models trained on the dataset, which aimed to predict whether a person made over 50,000 dollars a year. After the toolkits had assessed the models for bias, we viewed their dashboards to come to our findings. We evaluated fairness using the Statistical Parity difference metric, and the sensitive attribute that was taken into consideration was gender.

6.3 Findings

In AIFairness360, the parity difference spectrum goes from -1 to 1 (rather than 0 to 1), so the magnitude of difference of 0.33 should be divided by the total range of 2 to get the percentage, which is 16.5%. As for Fairlearn, we calculate the difference as 25.7% - 7.43%, which is 18.27%. Interestingly, 16.5% and 18.27% are rather far apart, although the bias-assessment method was the same, as well as the data. There are two possible explanations for this disparity. One explanation is that the respective logistic regression models that the toolkits analyzed were slightly different. Even if two models are trained on the exact same data in the exact same way, they can end up with different amounts of loss. However, this disparity can also be explained by inner differences in the bias-detection algorithms. This seems like a more likely reason, because even if the models have slight differences, the differences are most likely marginal, since they trained on the same data using the same methods. Although this disparity is not huge, in some cases it might affect

the outcome. For example, if a certain company decides to asses bias present in the dataset using the AIFairness360 toolkit, and chooses to set a threshold of 15% (if the bias rate is higher than 15%, then work on mitigating it, otherwise - leave it as it is), AIFairness360 might output 14%, but Fairlearn - 15.77%, then this slight difference could potentially influence real people.

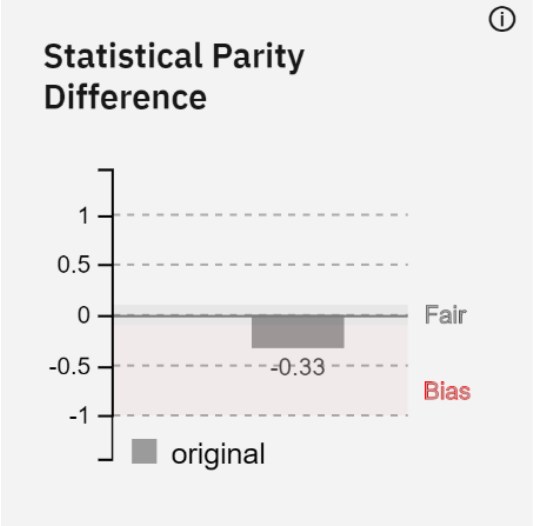

Figure 2: Dashboard from AIFairness360 demo[2]

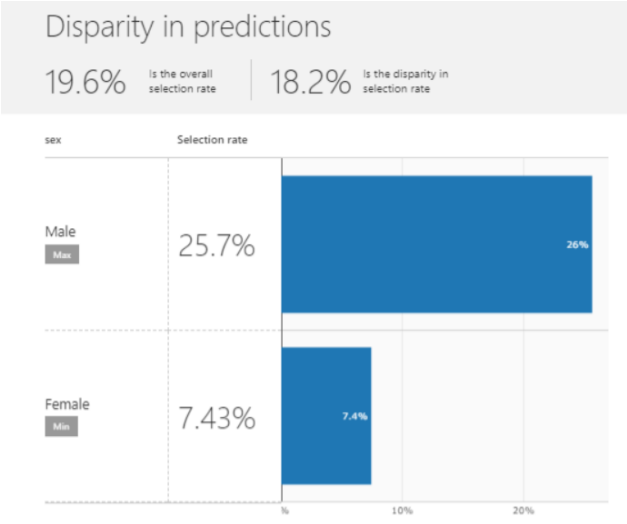

Figure 3: Dashboard from Fairlearn demo[5]

7 Takeaways

Looking at these findings, I find the difference between the parity differences between the toolkits considerable, and worth further investigation. First of all, we must be sure whether the difference is a result of the models slightly differing, or the performance of the toolkit. If the toolkits are truly the cause, we should investigate why this disparity exists between the toolkits. This difference

might seem nominal - about less than 2%. However, that 2% can become a large problem in other contexts. If the model was mitigated assuming that the lower parity difference was correct, and the model was widely implemented by banks to predict whether to give someone a loan, for example, it could create further disparities between men and women. Toolkits should be accurate and consistent across the board, or a user with a potential conflict of interest (such as valuing accuracy over the mitigation of societal disparities) could potentially select a toolkit that would align with that conflict of interest.

References

[1] Solan Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning: Limitations and Opportunities*. URL: `https://fairmlbook.org/`. (accessed:08.10.2020).

[2] Rachel K. E. Bellamy et al. *AI Fairness 360 Open Source Toolkit*. URL: `https://aif360.mybluemix.net/`. (accessed: 08.12.2020).

[3] Tolga Bolukbasi et al. "Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings". In: *CoRR* abs/1607.06520 (2016). arXiv: `1607.06520`. URL: `http://arxiv.org/abs/1607.06520`.

[4] Joy Buolamwini and Timnit Gebru. "Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification". In: *Proceedings of Machine Learning Research* 81 (2018), pp. 77–91.

[5] Miro Dudík et al. *Fairlearn*. URL: `https://fairlearn.github.io/`. (accessed: 08.10.2020).

[6] Cynthia Dwork et al. "Fairness Through Awareness". In: *CoRR* abs/1104.3913 (2011). arXiv: `1104.3913`. URL: `http://arxiv.org/abs/1104.3913`.

[7] Hila Gonen and Yoav Goldberg. "Lipstick on a Pig: Debiasing Methods Cover up Systematic Gender Biases in Word Embeddings But do not Remove Them". In: *CoRR* abs/1903.03862 (2019). arXiv: `1903.03862`. URL: `http://arxiv.org/abs/1903.03862`.

[8] James Wexler et al. *What-If Tool*. URL: `https://pair-code.github.io/what-if-tool/`. (accessed: 08.10.2020).

[9] Ziyuan Zhong. *A Tutorial on Fairness in Machine Learning*. URL: `https://towardsdatascience.com/a-tutorial-on-fairness-in-machine-learning-3ff8ba1040cb`. (accessed: 08.10.2020).