OPTIMIZING FOR FAIRNESS IN MACHINE LEARNING

Are Machine Learning Algorithms Really At A Loss?

Sajad Hashemi, Zahra Nadine Kandola, Scott Oxholm, and Rachael Walker, *University of Toronto*

Abstract—Machine learning (ML) models used for sensitive decision-making can exhibit biases against marginalized populations, leading to unfair outcomes. This paper focuses on the use of fairness regularization as an approach to mitigate biases in ML models. We developed an experimentation framework to study the impact of fairness regularizers on model performance using two publicly available datasets: COMPAS Recidivism and Adult Income. Our study used logistic regression with binary cross-entropy loss and two definitions of fairness, demographic parity and equalized odds. We found that regularizing for fairness had an impact on model performance, with a fairness-accuracy trade-off observed. Extreme values of the regularization parameter resulted in models that converged to trivial classifiers, which were considered "fair" but were not useful for practical purposes. Our paper highlights the importance of considering recall and precision, in addition to accuracy, when selecting an appropriate regularization parameter, and the potential conflicts between different definitions of fairness. Further research is needed to understand how to tune the trade-off parameter when fairness definitions conflict and to generalize the findings to other models and datasets.

GitHub Link - https://github.com/zfk3/MIE424-Final-Project-.git

1 INTRODUCTION

Machine learning (ML) models are increasingly being used to support sensitive decision-making in domains like loan underwriting, employment, and criminal justice [1][2][3][4]. However, there is growing evidence that these models can exhibit biases that discriminate against marginalized populations based on attributes such as race and gender [5].

One real-world example is the commercial algorithm developed by *Northpointe, Inc.* which was trained on the COMPAS dataset to generate recidivism risk scores that support sentencing decisions in the justice system [6]. In 2016, *Propublica* found that *Northpointe*'s algorithm was more likely to overestimate recidivism risk for black defendants compared to white defendants [6]. This is because the model was trained on a dataset where the non-caucasian defendants were more likely to be re-arrested, with the model learning this relationship and using it for prediction [6]. However, deriving risk scores from re-arrest data is flawed because arrests are influenced by biased policing practices; non-caucasian individuals are more likely to be stopped and arrested by police for the same behaviour [7]. As such, ML models can *perpetuate* human biases implicit in the dataset [7]. Indeed, higher risk scores can lead to harsher sentencing, contributing to circumstances that further increase their recidivism risk for marginalized defendants [7]. This example illustrates a real-world case as to why fairness must be considered in ML in order to prevent the perpetuation of historical biases and injustices that create avoidable adversity in the lives of civilians [8].

The literature discusses three main approaches to mitigate these biases and introduce fairness [9]:

1. *Pre-processing*: modification of training data.
2. *Post-processing*: modification of model outputs after training.
3. *In-processing*: modification of model to enforce fairness when training.

We focus on in-processing since it is flexible and generalizes to a wide range of use cases [10]. Within in-processing, there are two main ways to enforce fairness: (1) including fairness constraints or (2) adding a fairness term to the objective function [10]. We explore the latter.

In this paper, we study the effects of adding fairness regularizers to the objective function when training binary classification models. First, we seek to examine the fairness-accuracy tradeoffs of regularizing for a single definition of fairness. We then extend the literature by exploring the effectiveness of regularizing for multiple definitions of fairness simultaneously.

2 RELATED WORK

The literature has attempted several approaches to implementing fairness regularization. One such methodology was outlined in "Too Relaxed to be Fair" by Lohaus et al. Specifically, they "relaxed" the true, non-convex definitions of fairness into convex metrics that could be included directly into the loss function while training [11]. However, they found that using convex proxies did not actually optimize for the desired fairness definitions, often producing unfair results [11].

Another methodology, found through the citations of "Too Relaxed to be Fair", is the process outlined by Agerwal et al. in "A Reductions Approach to Fair Classification" [5]. In contrast to the convex relaxations, this paper directly uses the true fairness definitions. Agerwal et al. focus on two popular definitions: equalized odds and demographic parity. They

develop a novel formulation that can represent both of these definitions as a set of linear constraints. These constraints can then be synthesized into a regularization term in a model's objective function.

This is exactly the approach taken by FairTorch, a repository that won first prize at the 2020 Global PyTorch Summer Hackathon in the Responsible AI category [12]. Their repository was intended to be used as a library of fairness regularizers that can be integrated into PyTorch models. However, their implementation did not include any experimentation or exploration of real-world applications. In this paper, we seek to fill this gap by developing an experimentation framework, testing with real data, and developing intuition about how these regularizers impact model performance.

3 Datasets

We conducted our analysis on two datasets: the COMPAS Recidivism dataset and the Adult Income dataset. Both are publicly available, collected through real-world processes, and contain implicit biases with respect to sensitive features. Each dataset was used by both Lohaus et al. in "Too Relaxed to be Fair" and Agarwal et al. in "A Reductions Approach to Fairness" [11][5]'. Feature selection was performed on both datasets using feature correlation and importance. Results are presented in **Appendix 8.2** and **8.3** for COMPAS and Adult Income, respectively. Preprocessing was required to one-hot encode categorical features and binary-encode binary features. Both datasets were then split using a fixed seeding into 70% train data, 20% validation data, and 10% test data. The specifics of each dataset will be discussed in more detail below.

3.1 COMPAS Recidivism Dataset

The Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) dataset was created in 2013-14 and contains 7,214 rows. Each entry contains an individual defendant's information from a screening survey including their demographic characteristics and criminal history [13]. This data is traditionally used for binary prediction tasks to determine whether a defendant will recommit a crime within two years (1) or not (0). As was mentioned in the introduction, the sensitive feature is race, which we have defined as caucasian or non-caucasian; non-caucasian defendants were 23% more likely to be re-arrested compared to caucasian defendants. The dataset is relatively balanced across the target feature (45% recommitted vs. 55% did not) but imbalanced across the race feature (34% caucasian and 66% non-caucasian).

3.2 Adult Income Dataset

The Adult Income dataset consists of a cleaned sample from the 1994 U.S. Census and contains 32,561 rows [13]. Each entry contains information about a surveyed individual including their occupation, educational level and sex. The data is commonly used to predict whether an individual's annual income exceeds $50K. The sensitive feature is sex, with females being almost 30% more likely to have a "low" income of below $50K. The dataset is heavily imbalanced across the target feature (24% have an income of above $50K and 76% do not) as well as the sex feature (33% female and 67% male).

4 Methods

4.1 Optimizing for Fairness During Model Training

As mentioned above, we explore an in-processing approach to fairness by adding fairness regularization to the model objective function when training. Our formulation was inspired by Lohaus et al.'s approach in "Too Relaxed to be Fair". Equation (1) contains the original problem formulation they proposed

$$\min_{\mathbf{f} \in F} L(f) + \alpha R \tag{1}$$

where L is the loss with respect to a model f from the model class F, R is a fairness regularizer, and α is the tradeoff parameter which specifies the weighting of the fairness term with respect to the model loss. We extend this formulation by adding additional regularization terms to account for multiple fairness definitions that we are looking to optimize. Our formulation is presented in Equation (2).

$$\min_{f \in F} L(f) + \alpha_1 R_1 + \alpha_2 R_2 + \cdots + \alpha_n R_n \tag{2}$$

One benefit of this formulation is that it is very modular; different models, loss functions and fairness regularizer variations can be easily substituted for experimentation. This training problem can then be flexibly solved using a gradient descent optimization approach.

Fairness regularizers modify the model's learning objective (i.e., the loss function) to incentivize the model to exhibit fair behaviour, avoiding discrimination on the basis of sensitive demographic attributes. Unlike the convex proxies used by Lohaus et al., our regularizers were derived from the true non-relaxed definitions formulated in "A Reductions Approach to Fairness" [5]. This resulted in a non-convex, continuous optimization problem. Notably, the non-convexity eliminates the guarantee that gradient descent will find the global optimal solution. However, Lohaus et al.'s findings suggest that it is more desirable to find a truly fair local optimum than an unfair global optimum derived from an unrepresentative relaxation. Our non-convex regularizers penalize the absolute difference or "gap" in the predicted outcomes between groups according to a given definition of fairness. The minimization of this unfair gap may conflict with minimization of model loss, especially if the underlying data is significantly biased. Thus, modelers must decide how to balance the tradeoff between fairness and overall

model performance (i.e., ideally achieving fairness with minimal impact on performance).

For ease of exposition, we performed our experimentation using a simple logistic regression model with binary cross-entropy loss. We also limited ourselves to two definitions of fairness (i.e., combining a maximum of two distinct regularizers at a time). Specifically, we used the non-convex fairness definitions formulated by Agarwal et al. for demographic parity and equalized odds.

4.2 Regularizing for Demographic Parity

Demographic parity (DP) defines fairness as when positive classification is statistically independent from a sensitive feature [12]. Mathematically, this can be expressed as Equation (3)

$$P[f(x) = \hat{y}|A = a] = P[f(x) = \hat{y}] \quad \forall a, \hat{y} \tag{3}$$

where $f(x)$ is the prediction of the chosen classifier f, a is a value of the categorical sensitive attribute A, and $\hat{y}$ is a particular outcome value. Since $\hat{y} \in \{0, 1\}$, Agerwal et al. note that this is equivalent to Equation (4).

$$E[f(x)|A = a] = E[f(x)] \quad \forall a \tag{4}$$

DP regularization is formulated to penalize the gap (absolute difference) between these means where a "fair" model should exhibit a DP gap of zero. In addition, since our datasets have binary sensitive attributes, Equation (4) is equivalent to having equal expected classification values across groups. Thus, we can redefine the gap as the difference in expected predictions across the two different values a for the sensitive attribute A. Indeed, our regularizer for DP is formulated in Equation (5)

$$R_{DP} = |E[f(x)|A = a_0] - E[f(x)|A = a_1]| \tag{5}$$

In our implementation, we subtract the mean predictions (i.e., the expected prediction value) for each group i, written below as μ_{ai}, in each direction. We then apply the ReLU function to each subtraction, recreating the absolute value as is illustrated in Equation (6).

$$R_{DP} = ReLU[\mu_{a0} - \mu_{a1}] + ReLU[\mu_{a1} - \mu_{a0}] \tag{6}$$

Notably, DP is incompatible with perfect accuracy if the outcomes are not proportionate across sensitive attribute groups in the underlying dataset.

4.3 Regularizing for Equalized Odds

Equalized Odds (EO) defines fairness as when positive classification across groups is conditionally independent from a sensitive feature given the true label. Unlike DP, which strives for equal outcomes, EO aims to create equal error rates across different sensitive groups. Mathematically, this can be expressed as Equation (7)

$$P[f(x) = \hat{y}|A = a, Y = y] = P[f(x) = \hat{y}|Y = y] \quad \forall a, y, \hat{y} \tag{7}$$

where $f(x)$ is the prediction of the chosen classifier f, a is a value of the categorical sensitive attribute A, and $\hat{y}$ is a particular outcome value and y is the true label outcome. Since $\hat{y} \in \{0, 1\}$, Agerwal et al. note that this is equivalent to Equation (8).

$$E[f(x)|A = a, Y = y] = E[f(x)|Y = y] \quad \forall a \tag{8}$$

Like for DP, our regularization term is formulated to penalize the gap between these means; a model that is fair according to EO should have a gap of zero. Our formulation is outlined in Equation (9).

$$\begin{aligned} R_{EO} = \\ |E[f(x)|A = a_0, Y = y_1] - E[f(x)|A = a_1, Y = y_1]| \\ +|E[f(x)|A = a_0, Y = y_0] - E[f(x)|A = a_1, Y = y_0]| \end{aligned} \tag{9}$$

In a similar fashion to DP, we implemented this using the mean predictions (i.e., the expected prediction value) for each group i and label $l \in \{0, 1\}$, written below as $\mu_{ai,l}$. We then apply the ReLU function to each subtraction to recreate this absolute value, as is illustrated in Equation (10).

$$\begin{aligned} R_{EO} = \\ ReLU[\mu_{a0,y=1} - \mu_{a1,y=1}] + ReLU[\mu_{a1,y=1} - \mu_{a0,y=1}] \\ +ReLU[\mu_{a0,y=0} - \mu_{a1,y=0}] + ReLU[\mu_{a1,y=0} - \mu_{a0,y=0}] \end{aligned} \tag{10}$$

Contrary to DP, the conditional nature of EO means that this definition of fairness can be achieved with perfect overall predictive accuracy.

4.4 Our Goal: Closing the Gap

As discussed above, by including the regularizers in the objective function, we are aiming to minimize fairness gaps along with the loss. In particular, we add a squared L2 norm of each gap to heavily penalize unfair models. Integrating our regularizers for DP and EO (Equations (6) and (10) respectively) into our proposed model formulation (as presented in Equation (2)), we get the following optimization problem as presented in Equation (11).

$$\min_{f \in F} L(f) + \alpha_{DP}\|R_{DP}\|_2^2 + \alpha_{EO}\|R_{EO}\|_2^2 \quad (11)$$

The advantage of this formulation is that by adjusting the α_{DP} and α_{EO} values we can create every variation of the optimization problem we need to implement. This includes the baseline model with no regularization, models that regularize for a single definition of fairness, and the model regularizing for multiple definitions of fairness simultaneously. **Table 1** summarizes how each α needs to be adjusted to achieve each model variant.

Table 1: Creating Model Variations with Different α values.

Model	α_{DP}	α_{EO}
Baseline	0	0
DP Optimization	>0	0
EO Optimization	0	>0
DP & EO Optimization	>0	>0

The selection of the tradeoff parameter values is an important design choice that can greatly impact model performance. Thus, the main objective of our experimentation was to develop an intuition for understanding which regularization strategy is best for enforcing fairness without sacrificing performance.

5 Experiments

5.1 Experimental Setup

All experiments were run locally using CPU through an experimentation pipeline built on Google Colab in Python. The data processing and analysis portion of our pipeline was built with the standard *pandas*, *plotly*, and *numpy* packages. The ML models and testing in our pipeline were built using *torch*, *sklearn*, and *fairlearn* (a library to compute fairness metrics from ML model outputs). The experimentation pipeline consists of the following components:

1. *Data Loaders*: To load and clean each dataset.
2. *Data Splitter*: Splits the data into train/validation/test sets with fixed seeding. We used a single split for all of our testing (i.e. no cross-validation). This is because our experimentation involved training many different model variants and cross-validation for all of them would be impractical given our time constraints.
3. *Models*: To initialize the desired model architecture(s) in Pytorch.
4. *Training and Testing Loop*: A framework to train, validate and test each model variant using gradient descent.

Model performance was evaluated using loss, accuracy, precision, and recall. Fairness was evaluated according to both DP and EO gaps. As per our objective, we then analyzed the trade-offs with accuracy when introducing fairness. We used *Weights and Biases* to log all of our model training and parameters used during each run.

5.2 Hyperparameter Tuning

Hyperparameters that required tuning were the number of epochs, learning rate, and the tradeoff parameter values α. The tradeoff parameters were also the independent variables in our experiments and will be discussed in **Section 5.3**.

In each experiment, we tested 5-10 values for both α_{DP} and α_{EO}, depending on the dataset (i.e. in total we implemented 25-100 models per experiment). Each experiment took 36-240 minutes to run. Runtime was dependent on the number of combinations of α being tested, the number of epochs, and the size of the training data. While the results presented below contain the best experimental runs, over 1800 prior runs were completed to tune hyperparameters, including α_{DP} and α_{EO}.

Given the many model variations, we selected a random subset of 5 α values to tune the number of epochs and learning rate. We tested several standard epochs (50,100,150, and 200) as well as several standard learning rates (0.01, 0.001, and 0.0001). Further manual tuning was conducted by responding to validation results (e.g., oscillations, plateaus, or lack of learning in the validation loss and accuracy curves). Iterative tuning was conducted until we found the hyperparameters used for our final experiments. We also experimented with linearly and exponentially decaying learning rates (to reduce our chances of finding a solution that is a sub-optimal local minima) and early stopping (to prevent overfitting and improve generalizability). However, they were not used for COMPAS or Adult Income as they led to inconsistent performance which made our results less interpretable. For the COMPAS dataset we used 100 epochs and a learning rate of 0.0001. For the Adult Income dataset, we used 200 epochs and a learning rate of 0.0001.

5.3 Results

In this section, we will present our results for both COMPAS and Adult Income together, as our goal was to determine if we can see generalization of using fairness regularizers across several datasets. The class imbalances present in the Adult Income data unfortunately led to a trivial classifier which always predicted the same class (see **Appendix 8.4** for initial results). This is not an unexpected result given the findings presented by "Too Relaxed to be Fair", in which a trivial classifier for Adult Income was also presented [11]. As a solution, we rebalance the data by randomly undersampling the majority class (income <= $50K) and use this preprocessed dataset for our results presented in this section. This solution is not ideal since we are not using all available data. However, given pre-processing was not our focus, it was a suitable solution for our purposes. Summary statistics

pertaining to this balanced dataset (similar to **Section 3**) are presented in **Appendix 8.5**.

5.3.1 Single Fairness Regularizer

Regularizing for Demographic Parity

Figure 1 shows the trajectory of the DP gap, EO gap, accuracy, precision and recall as α_{DP} regularization increases for the COMPAS dataset. In general, as α_{DP} increases, we see that the accuracy declines and our fairness gaps decrease (i.e. fairness improves across both definitions).

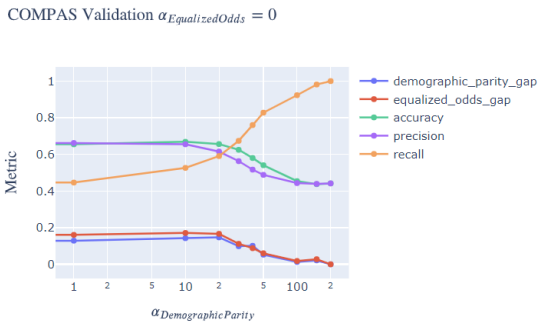

Figure 1: DP regularization results for α_{DP} values of 0, 1, 10, 20, 30, 40, 50, 100, 150, and 200 on the COMPAS data. Note that α_{DP} = 0 is the baseline.

At extreme values of α_{DP}, recall heavily increases and precision decreases because the model is converging to a trivial classifier which always predicts the positive class (i.e. guilty). This is illustrated in the confusion matrix in **Appendix 8.6.** This is an important discovery and reminder of how the model can be manipulated in undesired ways to achieve fairness. Specifically, always predicting the same thing (i.e. that a defendant will be rearrested) regardless of the sensitive attribute group is "fair" according to both DP and EO.

In **Appendix 8.7**, we see similar performance results from the balanced Adult Income dataset, but with a subtler trend. It is hypothesized that this is because the α_{DP} values were not large enough to produce a meaningful difference. This will be discussed further in **Section 5.3.2**.

Regularizing for Equalized Odds

The COMPAS results in **Figure 2** mirror the trends found when regularizing for DP; extreme values of α_{EO} also lead to a trivial classifier. The Adult Income dataset results were consistent with these findings (see **Appendix 8.7**).

Fairness-Accuracy Tradeoffs

Figure 3 shows the fairness-accuracy trade-off for EO on the COMPAS dataset across different α_{EO} values. We can see that there is a linear relationship; as the fairness gap decreases, so does the accuracy.

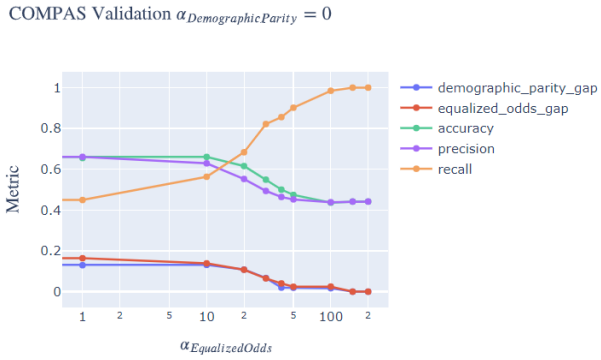

Figure 2: EO regularization results for α_{EO} values of 0, 1, 10, 20, 30, 40, 50, 100, 150, and 200 on the COMPAS data. Note that α_{EO} = 0 is the baseline.

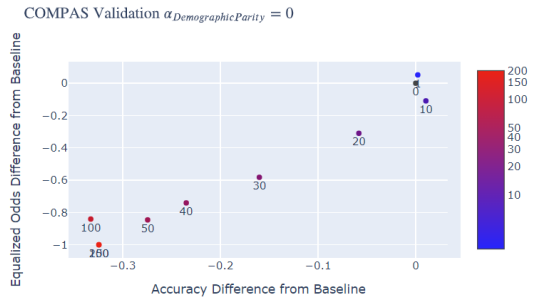

Figure 3: Accuracy vs. Fairness Tradeoffs for tested α_{EO} values.

While accuracy itself does not provide an obvious α selection, prior results illustrate that caution must be taken when selecting this tradeoff parameter; high α values not only decrease accuracy but result in trivial classifiers that do not produce valuable predictions (i.e. predicting everyone will be re-arrested within two years is "fair" but not useful for judicial decision-making). Thus, we advise that modelers use precision and recall (in addition to accuracy) to choose an α value that creates a non-trivial model.

5.3.2 Combined Fairness Regularizers

Figures 4-6 show the impact of combining regularizers on the COMPAS dataset, illustrating that there is a fairness-accuracy frontier that cannot be avoided. Specifically, two α combinations that yield the same accuracy in **Figure 4** will also yield the same DP and EO gaps in **Figures 5** and **6**, respectively. For example, the combinations $(\alpha_{DP} = 10,\ \alpha_{EO} = 10)$, $(\alpha_{DP} = 0,\ \alpha_{EO} = 30)$, and $(\alpha_{DP} = 20,\ \alpha_{EO} = 0)$ all have an accuracy of 0.62, a DP gap of around 0.1 and an EO gap of 0.11. In other words, there is no way to get around the fairness-accuracy trade-off by combining regularizers. Similar to regularizing for a single definition of fairness, high α values yield substantial declines in accuracy.

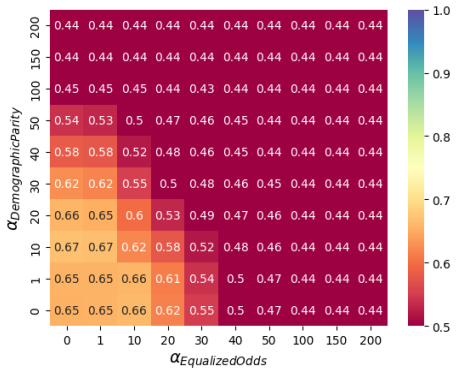

Figure 4: COMPAS Accuracy vs. α_{DP} and α_{EO}.

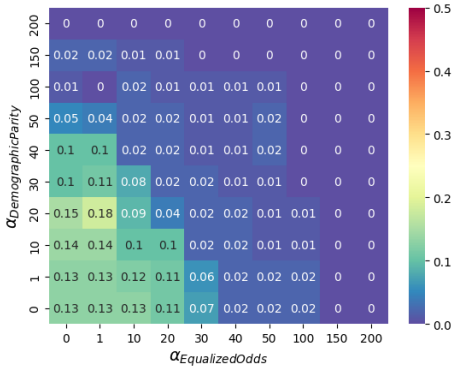

Figure 5: COMPAS DP Gap vs. α_{DP} and α_{EO}.

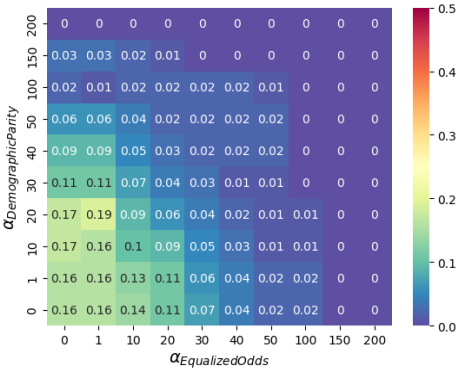

Figure 6: COMPAS EO Gap vs. α_{DP} and α_{EO}.

The two fairness regularizers appear to be complementary for COMPAS; regularizing for DP tends to improve EO and vice versa. However, this is merely because of the tendency described in **Section 5.3.1** for the classifiers to converge to trivial behaviour that satisfies both definitions of fairness (see **Appendix 8.8** for mathematical explanation). Indeed, we do not see this same pattern in the Adult Income dataset (see **Figures 7-8**). In fact, **Figure 8** shows that regularizing for EO on the Adult Income dataset *increases* the DP gap (i.e., reducing the error rates across groups leads to less equal outcomes).

Unfortunately, we cannot evaluate the tradeoff between regularizers since α_{DP} had minimal impact on any metrics for Adult Income. This is likely because the size of the DP gap for the Adult Income dataset was two orders of magnitude smaller than the EO gap and would require much larger α values to be effectively regularized. This illustrates that modelers must customize and tune α values for a given dataset.

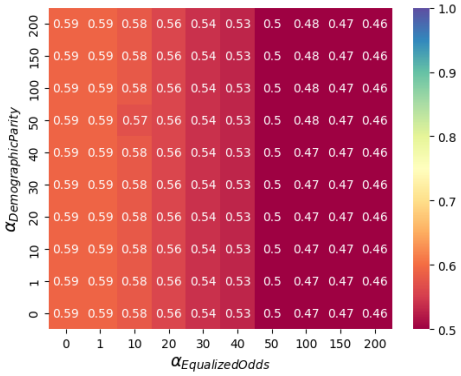

Figure 7: Adult Accuracy vs. α_{DP} and α_{EO}.

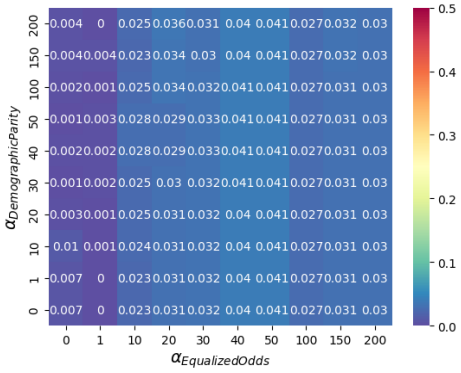

Figure 8: Adult DP Gap vs. α_{DP} and α_{EO}.

6 Conclusion

This paper examined the impact of adding different combinations of non-convex regularizers based on pure definitions of fairness to the training objectives for ML models. First, we find that recall and precision are critical to measure when regularizing for fairness to ensure the model does not converge to a trivial classifier. Second, we see that combining regularizers does not allow modelers to circumvent the fairness-accuracy tradeoff. Finally, we find that fairness definitions can conflict.

Further work is needed in two main domains. First, to understand how to tune the tradeoff parameter when definitions of fairness conflict. Second, we should ideally see generalization of results presented in this paper across additional (1) models and (2) datasets. For models, since we have non-convex regularizers, it would be interesting to see how this approach performs for a non-convex model, such as neural networks. For data, verification that results generalize to other balanced datasets, like the Dutch Census dataset, would be ideal. Evidence of the same patterns across these other experimental setups will further inform the viability of our approach and findings.

7 References

[1] J. Dastin, "Amazon scraps secret AI recruiting tool that showed bias against women," Reuters, 10-Oct-2018. [Online]. Available: https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G. [Accessed: 09-Apr-2023].

[2] K. Hale, "A.I. bias caused 80% of black mortgage applicants to be denied," Forbes, 09-Nov-2022. [Online]. Available: https://www.forbes.com/sites/korihale/2021/09/02/ai-bias-caused-80-of-black-mortgage-applicants-to-be-denied/?sh=4451827736fe. [Accessed: 09-Apr-2023].

[3] K. Hao, "Ai is sending people to jail-and getting it wrong," MIT Technology Review, 02-Apr-2020. [Online]. Available: https://www.technologyreview.com/2019/01/21/137783/algorithms-criminal-justice-ai/. [Accessed: 09-Apr-2023].

[4] T. A. Press, "U.S. warns of discrimination in using artificial intelligence to screen job candidates," NPR, 12-May-2022. [Online]. Available: https://www.npr.org/2022/05/12/1098601458/artificial-intelligence-job-discrimination-disabilities. [Accessed: 09-Apr-2023].

[5] A. Agarwal, A. Beygelzimer, M. Dudik, J. Langford, and H. Wallach, "A reductions approach to fair classification," 03-Jul-2018. [Online]. Available: http://proceedings.mlr.press/v80/agarwal18a.html. [Accessed: 09-Apr-2023].

[6] J. Larson, J. Angwin, L. Kirchner, and S. Mattu, "How we analyzed the compas recidivism algorithm," *ProPublica*, 23-May-2016. [Online]. Available: https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm. [Accessed: 17-Apr-2023].

[7] L. Eckhouse, "Opinion | big data may be reinforcing racial bias in the criminal justice system," *The Washington Post*, 07-Apr-2023. [Online]. Available: https://www.washingtonpost.com/opinions/big-data-may-be-reinforcing-racial-bias-in-the-criminal-justice-system/2017/02/10/d63de518-ee3a-11e6-9973-c5efb7ccfb0d_story.html. [Accessed: 17-Apr-2023].

[8] E. T. Israni, "When an algorithm helps send you to prison," *The New York Times*, 26-Oct-2017. [Online]. Available: https://www.nytimes.com/2017/10/26/opinion/algorithm-compas-sentencing-bias.html. [Accessed: 17-Apr-2023].

[9] F. Petersen, D. Mukherjee, Y. Sun, and M. Yurochkin, "Post-processing for individual fairness," *arXiv.org*, 26-Oct-2021. [Online]. Available: https://arxiv.org/abs/2110.13796. [Accessed: 17-Apr-2023].

[10] Q. Ye and W. Xie, "Unbiased Subdata Selection for Fair Classification: A Unified Framework and Scalable Algorithms," *2012.12356.pdf*. [Online]. Available: https://arxiv.org/pdf/2012.12356.pdf. [Accessed: 17-Apr-2023].

[11] M. Lohaus, M. Perrot, and U. V. Luxburg, "Too relaxed to be fair," *PMLR*, 21-Nov-2020. [Online]. Available: https://proceedings.mlr.press/v119/lohaus20a.html. [Accessed: 17-Apr-2023].

[12] A. Fukuchi, M. Sode, and Y. Yobe. *FairTorch*, 2020. [Source Code]. https://github.com/wbawakate/fairtorch. [Accessed: 17-Apr-2023].

[13] T. L. Quy, A. Roy, V. Iosifidis, W. Zhang, and E. Ntoutsi, "A survey on datasets for fairness-aware machine learning," *arXiv.org*, 21-Jan-2022. [Online]. Available: https://arxiv.org/abs/2110.00530. [Accessed: 17-Apr-2023].

8 Appendix

8.1 Contribution Table

Legend:

- RW: Rachael Walker
- SH: Sajad Hashemi
- SO: Scott Oxholm
- ZK: Zahra Nadine Kandola

Ordering was completed alphabetically by initials.

Table A1: Contribution Table

Task	Contributor
Code	
Exploration of Relaxed Definitions of Regularization	SO
Demographic Parity Regularizer	SO
Equalized Odds Regularizer	SO, SH
Forward Pass of Gradient Descent	SO
Data Acquisition	RW, ZK
Exploratory Data Analysis	RW, ZK
Feature Selection and Data Pre-processing	RW, ZK
Logistic Regression Model	RW, ZK
Neural Network Model*	SH
Data and Experimentation Pipeline	RW, ZK
Hyperparameter Tuning	RW, ZK
Optimization Problem Implementation	RW, ZK
Model Training	RW, ZK
Results Analysis and Graphs	RW, ZK
Github Repo	RW, ZK
Paper	
Abstract	SH
Introduction	RW, SH, SO, ZK
Related Work	RW, SO, ZK
Datasets	RW, ZK
Methods	RW, SH, SO, ZK
Experiments	RW, ZK
Conclusion	RW, ZK
References	SH
Appendix	RW, ZK

* Unable to perform full experimentations due to time constraints.

8.2 Overview of COMPAS Dataset

This **Appendix** includes summary statistics and the results of feature selection for the COMPAS dataset. Below is the distribution of the dataset across the target and sensitive classes.

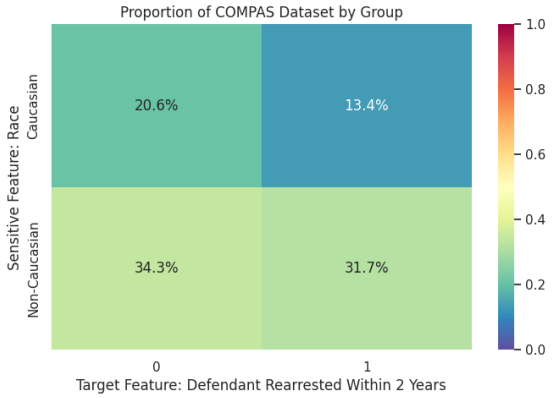

Figure A1: Proportion of COMPAS dataset across the sensitive and target classes.

These features were not used for COMPAS because they contain latent information about the target variables.

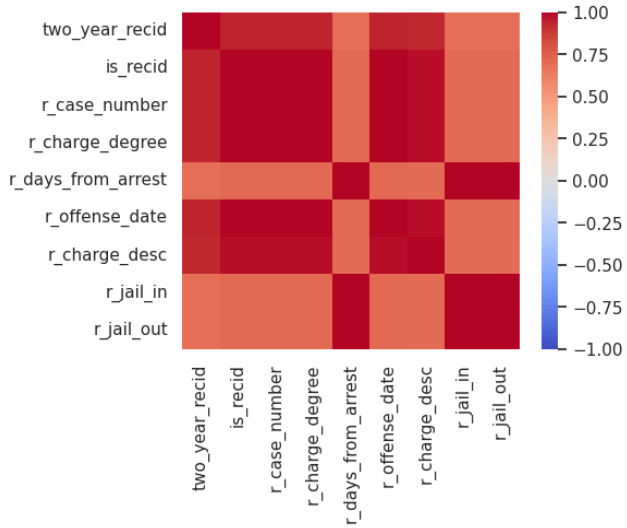

Figure A2: Correlation Matrix for COMPAS dataset for features with latent recidivism information (i.e. using null vs. non-null values for comparison).

Next, we show a correlation matrix containing the final feature set we used for COMPAS models.

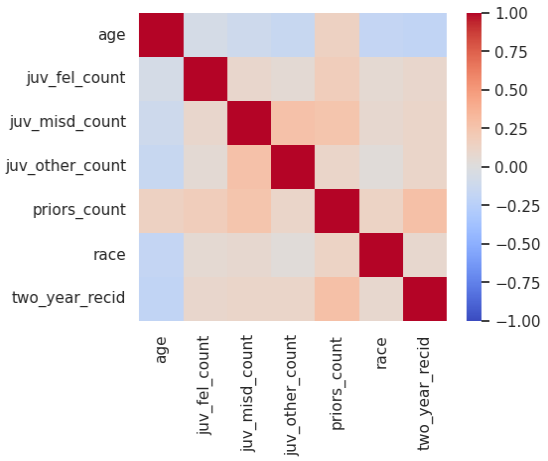

Figure A3: Correlation Matrix for COMPAS dataset for chosen feature set.

8.3 Overview of Adult Income Dataset

This **Appendix** includes summary statistics and the results of feature selection for the Adult Income dataset. Below is the distribution of the dataset across the target and sensitive classes.

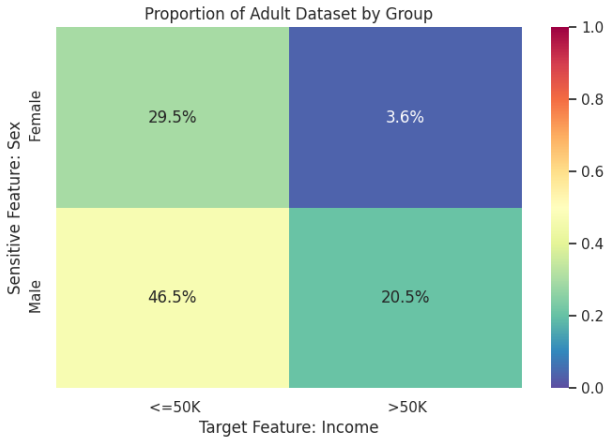

Figure A4: Proportion of Adult dataset across the sensitive and target classes.

Below is a correlation matrix for one-hot-encoded marital status features for the adult dataset which was used to pick a subset of features which had signal with respect to the target feature. The strongest features, outlined in purple, were used in our final prediction pipeline.

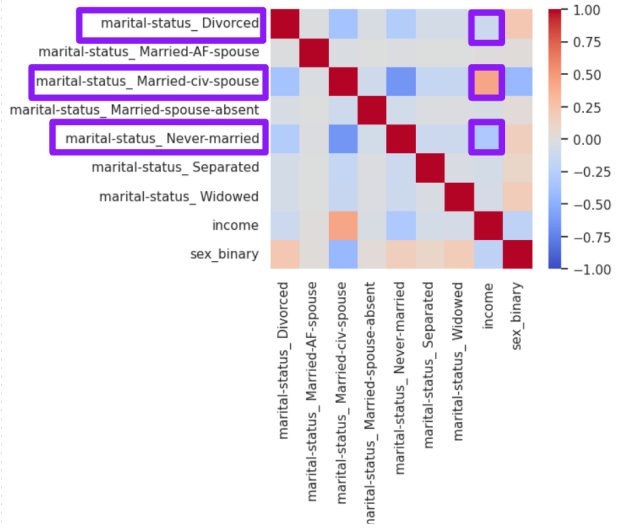

Figure A5: Correlation Matrix for one-hot encoded marital status features. Chosen features are in purple as they had the strongest correlation with the target feature (income).

Below is a correlation matrix for one-hot-encoded occupation features for the adult dataset which was used to pick a subset of features which had signal with respect to the target feature. The strongest features, outlined in purple, were used in our final prediction pipeline.

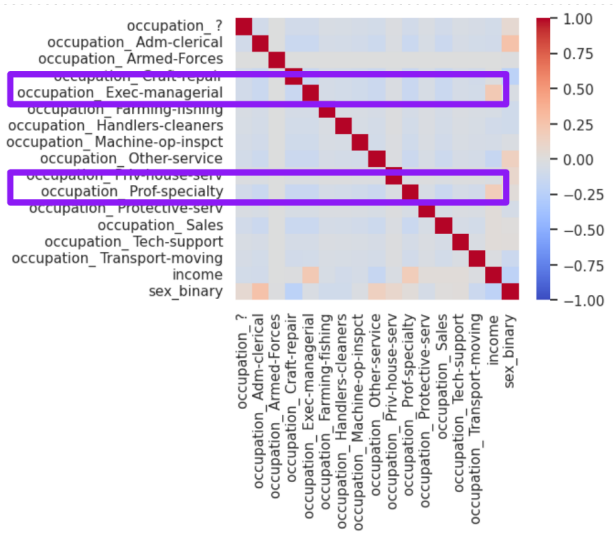

Figure A6: Correlation Matrix for one-hot encoded occupation features. Chosen features are in purple as they had the strongest correlation with the target feature (income).

Next, we show a correlation matrix containing the final feature set we used for Adult models.

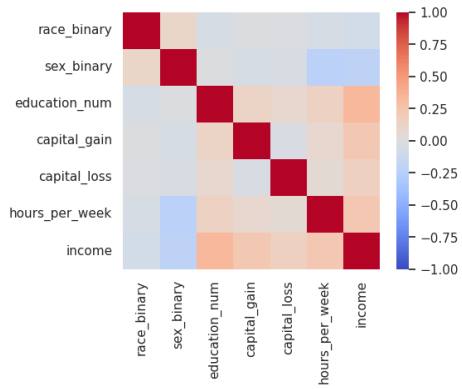

Figure A7: Correlation Matrix for Adult dataset for chosen non-one-hot-encoded feature set.

8.4 Unbalanced Adult Income Dataset Preliminary Results

This **Appendix** includes preliminary results found for the original, unbalanced, Adult Income dataset. **Figures A8** and **A9** show that all metrics were flat across all regularization values.

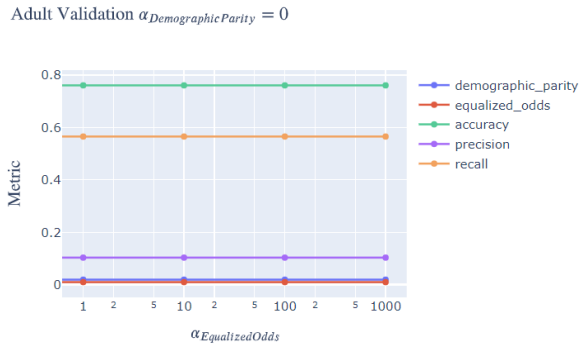

Figure A8: DP regularization results for α_{DP} values of 0, 1, 10, 100, and 1000 on the Adult Income data. Note that α_{DP} = 0 is the baseline.

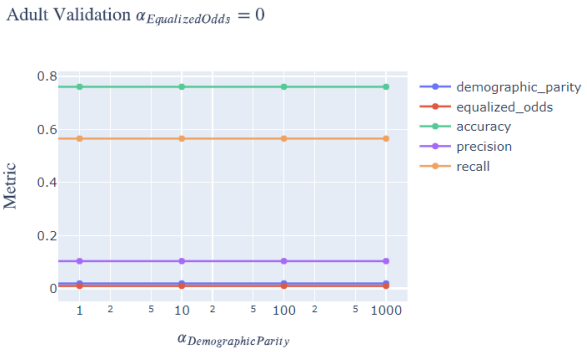

Figure A9: EO regularization results for α_{EO} values of 0, 1, 10, 100, and 1000 on the Adult Income. Note that α_{EO} = 0 is the baseline.

Accuracy was flat at 76%, the same as the proportion of data in the majority class (income <=$50K). This is symptomatic of the fact that all models converged to trivial classification of the majority class.

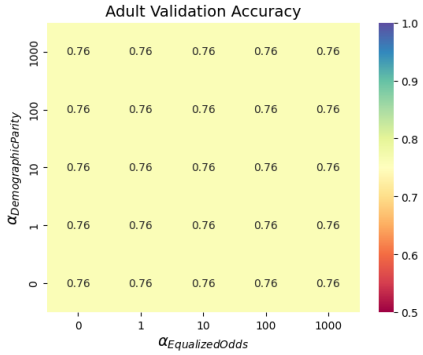

Figure A10: Adult Income validation Accuracy vs. α_{DP} and α_{EO}.

DP and EO are very close to zero because, as is discussed in **Appendix 8.8**, this behaviour is technically "fair" according to both definitions.

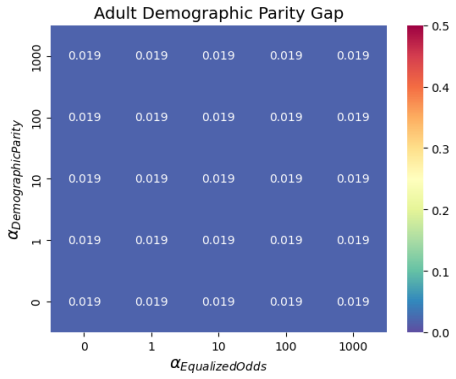

Figure A11: Adult Income DP Gap vs. α_{DP} and α_{EO}.

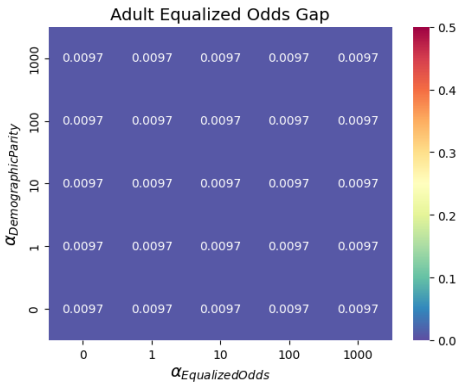

Figure A12: Adult Income EO Gap vs. α_{DP} and α_{EO}.

8.5 Overview of Balanced Adult Income Dataset

The balanced Adult Income dataset consists of a cleaned sample of the original Adult Income dataset with 15,682 rows. We used the same features as for the original Adult Income dataset. The dataset is now balanced across the target feature (50% have an income of above $50K and 50% do not) but very imbalanced across the sex feature (27% female and 73% male).

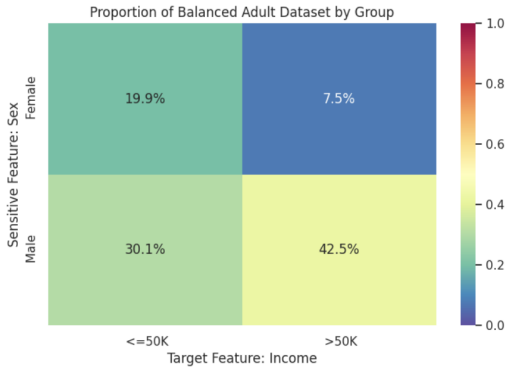

Figure A13: Proportion of Balanced Adult dataset across the sensitive and target classes.

8.6 COMPAS Dataset Single Regularizer Results

In **Figure A14** is a confusion matrix from one of our extreme regularization cases which shows that the model has converged to a trivial classifier which always predicts that a defendant will be re-arrested in two years.

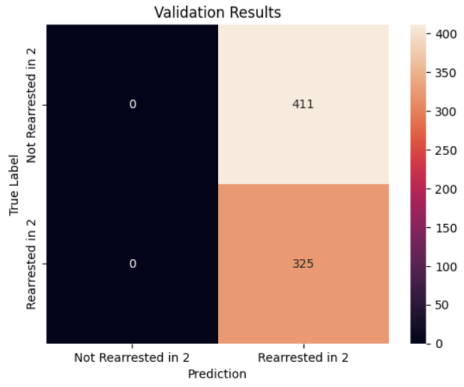

Figure A14: Confusion matrix for validation results of when regularizing for DP using an α_{DP} = 200. This matrix indicates that the model has converged to a trivial classifier.

In **Figure A15**, we see that changes in accuracy and DP have a linear relationship; as DP decreases so does accuracy. When regularization gets really high and the DP gap closes to zero accuracy stays static because the model has converged to the trivial classifier case illustrated in the confusion matrix above.

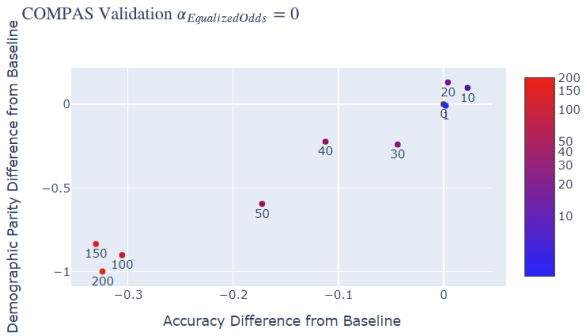

Figure A15: Accuracy vs. Fairness Tradeoffs for tested α_{DP} values.

8.7 Adult Income Dataset Single Regularizer Results

In **Figure A16**, we see the same trend as with COMPAS but with a more subtle trend. See **Section 5.5.1** for more information.

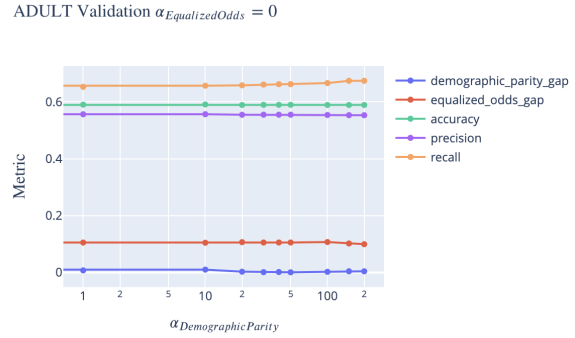

Figure A16: DP regularization results for α_{DP} values of 0, 1, 10, 20, 30, 40, 50, 100, 150, and 200 on the Balanced Adult Income data. Note that $\alpha_{DP} = 0$ is the baseline.

In **Figure A17**, we see the same trend as with COMPAS although it has not yet fully converged to a trivial classifier. See **Section 5.5.1** for more information.

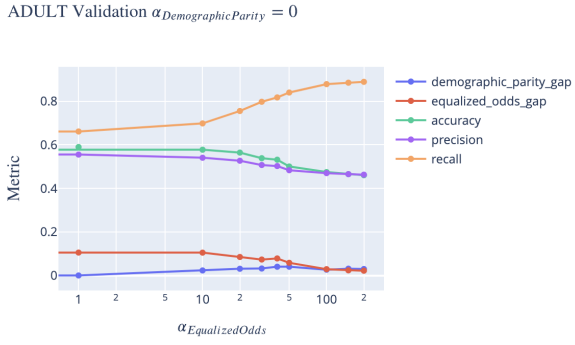

Figure A17: EO regularization results for α_{EO} values of 0, 1, 10, 20, 30, 40, 50, 100, 150, and 200 on the Balanced Adult Income data. Note that $\alpha_{EO} = 0$ is the baseline.

In **Figure A18**, we see the same trend as with COMPAS. See **Section 5.5.1** for more information.

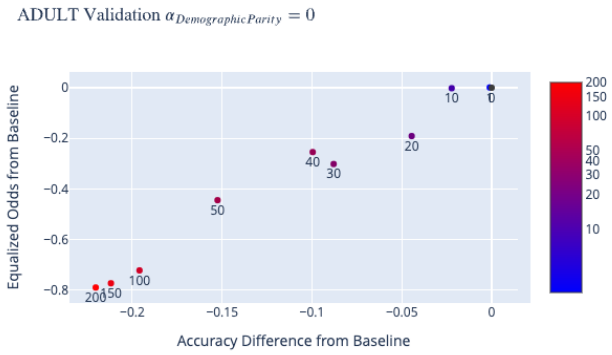

Figure A18: Accuracy vs. Fairness Tradeoffs for tested α_{EO} values.

In **Figure A19**, we see that there is no trend for demographic parity because the tradeoff parameter was not high enough.

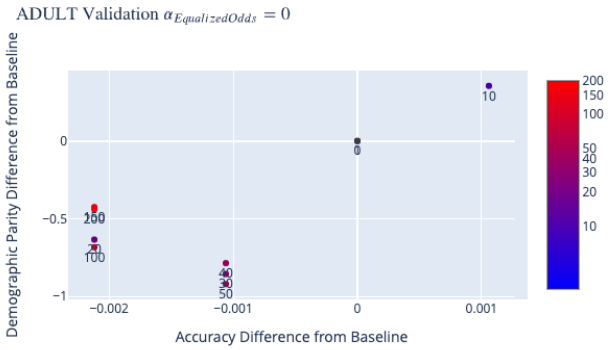

Figure A19: Accuracy vs. Fairness Tradeoffs for tested α_{DP} values.

8.8 Combining Regularizers Results

Emergence of Trivial Classifiers Discussion

When a model only predicts in one class, then $\forall a \in A$:

$$P(h(x)|A=a)=P(h(x))=0$$

DP will always be achieved when uniformly predicting a single class.

Also, $\forall a \in A, \forall y \in Y$:

$$P(h(x)|A=a,Y=y)=P(h(x))=0$$

EO will always be achieved when uniformly predicting a single class. Thus, regularizing may only further incentivize the model to take this approach and predict uniformly.

Additional Results for the Adult Income Dataset

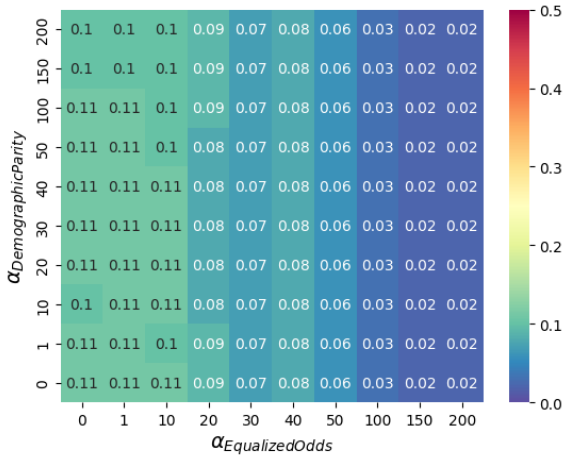

Figure A20: Adult Income EO Gap vs. α_{DP} ***and*** α_{EO}***.***