Team Tensor: Fairness in Machine Learning

Rohan Saxena
Carnegie Mellon University
rohansaxena@cmu.edu

Eric Huang
Carnegie Mellon University
ehuang2@andrew.cmu.edu

Brandon Trabucco
Machine Learning Department
btrabucc@andrew.cmu.edu

Poojan Palwai
Carnegie Mellon University
ppalwai@andrew.cmu.edu

Ananye Agarwal
Carnegie Mellon University
ppalwai@andrew.cmu.edu

1 Data preprocessing techniques for classification without discrimination

1.1 Overview of Method

Discrimination-Aware classification attempts to learn a model that doesn't have any discrimination toward sensitive attributes. The paper Faisal Kamiran [2011] focuses on two algorithms that, given only one binary sensitive attribute, attempts to pre-process the dataset so that discrimination against sensitive attributes are removed.

$$W(s,c) = \frac{|\{X \in D|X(S) = s\}|x|\{X \in D|X(C) = c\}|}{|D|x|\{X \in D|X(C) = c \cap X(S) = s\}|} \tag{1}$$

In the equation 1, D represents the data, S is the sensitive attribute, and C is the label of the dataset.

The first method is called re-weighting and it gives weight to every sample based on a sensitive attribute and label. The weights are calculated to make the sensitive attribute and the class labels probabilities independent. There is an unique weight for every combination of the sensitive attribute and the label, with the weights calculated using the equation: 1. These weights are then used with every single sample (based on the sensitive attribute and label in the sample) to train the classifier.

The second method is called uniform sampling and it attempts to re-sample the dataset in order to remove discrimination. The weights from the previous section are re-used with each pair of sensitive features and label being re-sampled (with replacement) based on that combinations weight multiplied by the number of elements in that combination: $W(s,c) * |\{X \in D|X(C) = c \cap X(S) = s\}|$.

1.2 Implementation

For the project, I implemented the re-weight and uniform sampling methods to get rid of any bias from the poverty level feature in the paper. While the paper only uses a binary sensitive attribute, the pre-processing algorithms can be extended by having more entries for the sensitive attribute, since that doesn't change the underlying math behind the algorithms. Due to the paper's limitation, I was only able to test on one sensitive attribute and not other attributes like the teacher's gender.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

1.3 Results

As a sanity check, I compared the percentage labeled as 1 for the different poverty levels between the original dataset and re-sampled dataset. The figure fig. 1 shows that the re-sampled dataset had samples with equal probability of being labeled 1 regardless of their poverty level, which shows that the algorithm seemed to make poverty level independent of the label.

Original Dataset

Poverty Levels	Low	Moderate	High	Highest
Percent Labeled as 1	52.94%	50.02%	51.29%	57.60%

Dataset with Resample Algorithm

Poverty Levels	Low	Moderate	High	Highest
Percent Labeled as 1	54.80%	54.80%	54.80%	54.80%

Figure 1: The top table is the probability, for each of the poverty levels, of being labeled as 1 for the original model. The bottom table is the probability, for each of the poverty levels, of being labeled a 1 using the re-sampling algorithm.

I also checked whether the re-sampling really affected the base rate or baseline of the data set. From fig. 2, no real changes were between the original dataset and re-sampled dataset's base rates and baselines.

Original Dataset

Train-Validation Pair ID	Base Rate	Baseline 1: Total Asking Price
1	44.31%	55.53%
2	45.30%	59.62%
3	35.50%	48.20%

Dataset with Resample Algorithm

Train-Validation Pair ID	Base Rate	Baseline 1: Total Asking Price
1	44.58%	55.54%
2	45.36%	60.44%
3	35.21%	47.14%

Figure 2: The table are the base rate and baseline (using feature total asking price) for the original model (top table) and model with resample algorithm (bottom table).

Finally, from fig. 3, the new datasets didn't result in a significant loss or gain in performance. As a result, the algorithms seemed to have reduced discrimination towards the sensitive attribute without significantly affecting the accuracy of the model.

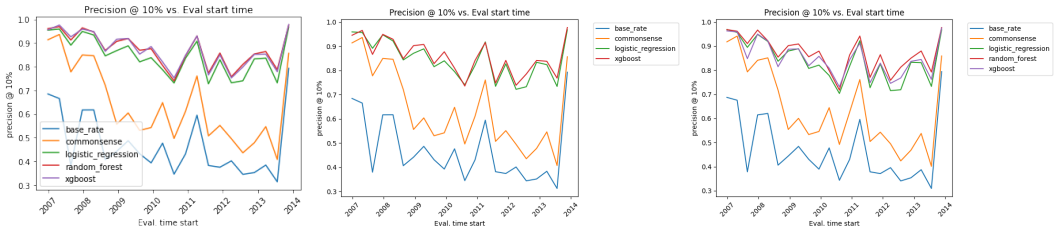


Figure 3: The above plots is the precision of the different models with original dataset (left), dataset with weights (middle) and re-sampled dataset (right). In the figures, blue represents base rate, yellow represents commonsense, green represents logistic regression, red represetns random forest, and purple represents xgboost.

1.4 Recommendations

The main issue with the pre-processing algorithms are that they can only be applied with one sensitive attribute. which limits it from working on datasets that have multiple sensitive attributes (such as gender and race). Furthermore, the algorithms fight bias by making the dataset's labels uniform between for the different elements of a sensitive attribute. This doesn't always work since a correlation between a sensitive attribute and a label can be ethical (e.g. car insurance companies should be able to base prices on the number of previous accidents even if this results in a higher number of men being denied insurance). However, the main benefits of these pre-processing algorithms is that they're efficient and make the sensitive attributes independent from the label. If a person wants a simple method to reduce discrimination, than the approaches in Faisal Kamiran [2011] work fairly well.

2 Classification with Fairness Constraints: A Meta-Algorithm with Provable Guarantees

2.1 Overview of Method

The goal of this paper Celis et al. [2019] is to develop a general method for solving fair classification tasks that applies to a variety of definitions of fairness. Previous frameworks have applied to a handful of fairness definitions, such as Hardt et al. [2016], where an algorithm is developed that can only directly handle false positive parity and true positive parity among target groups. However, when a different type of fairness is desired—this happens in practice when the goals of an organization change and a different user experience is desired—a different method is typically required.

In order to generalize to different definitions of fairness, the authors propose a framework for solving any ρ-Fair optimization problem with linear-fractional fairness constraints. Furthermore, unlike previous work, the proposed method generalizes to satisfying multiple definitions of fairness simultaneously. The proposed framework relies on linear-fractional group performance functions $q_i(f)$ defined below, where f is a classification model, G_i is a group, and $\mathcal{A}, \mathcal{B}$ are events.

$$q_i(f) = \frac{\alpha_0^{(i)} + \sum_{j=1}^{k} \alpha_j^{(i)} \cdot \Pr_\tau \left[f = 1 \mid G_i, \mathcal{A}_j^{(i)} \right]}{\beta_0^{(i)} + \sum_{j=1}^{k} \beta_j^{(i)} \cdot \Pr_\tau \left[f = 1 \mid G_i, \mathcal{B}_j^{(i)} \right]} \tag{2}$$

Given linear-fractional group performance functions, which can be defined according to most types of fairness (such as false discovery parity), the authors propose a method for solving ρ-Fair optimization problems. These are classification tasks where the goal is to find f to minimize $\Pr[f \neq Y]$, where Y is a label, augmented with fairness constraints of the form $\min_i q_i(f) / \max_i q_i(f) \geq \tau$. These constraints are non-linear, and the method achieves this by solving a relaxed optimization problem with linear constraints $l_i \leq q_i \leq u_i$, and converting the solution back to a PAC ρ-Fair solution.

2.2 Implementation

We employ this method in our project for the task of predicting the projects most at risk of being unfunded. We treat the gender of the teacher as the sensitive attribute. We infer this attribute on the basis of the prefix to the teacher's name. We assign a label of man for "Mr", and a label of woman for "Ms" and "Mrs". We consider two fairness measures – statistical rate (SR) and false discovery rate (FDR). We then implement algorithm 1 as described in the paper (using the code provided by the authors) to tune each fairness measure for our problem. Our results are summarized in fig. 5.

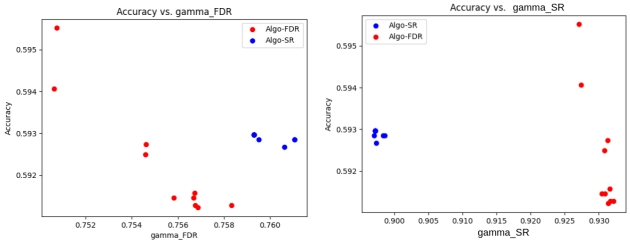
We plot some particularly interesting trends in fig. 4. In fig. 4a, we see that Algo-FDR is able to increase the fairness metric γ_{FDR}, albeit at the cost of accuracy. What is more interesting is that while Algo-SR optimizes γ_{SR}, it does have confounding effects on the value of γ_{FDR} too. We also see that the method suffers from a loss in accuracy even when we do not optimize for fairness at all, as reported by the original paper (proved to be due to the error in estimating the data distribution).

Meanwhile, in fig. 4b, we see that the method converges to a high value of $\gamma_{\mathrm{SR}} \approx 0.9$ even without optimizing it, suggesting that the dataset may already be fair with respect to the SR metric. Indeed, trying to optimize this metric further yields diminishing returns. We also observe a similar trend as before, where trying to optimize γ_{FDR} has side effects on the value of γ_{SR}.

	Acc	ΔAcc	ΔTPR	γ_{FDR}	γ_{SR}	γ_{FPR}	γ_{FNR}	γ_{TPR}	γ_{TNR}	γ_{AR}	γ_{FOR}	γ_{PPR}	γ_{NPR}
Unconstrained	0.42	0.11	0.04	0.83	0.97	0.98	0.49	0.96	0.35	0.75	0.95	0.74	0.89
Algo-FDR	0.60	0.03	0.05	0.75	0.93	0.99	0.87	0.92	0.99	0.95	0.78	0.73	0.89
Algo-SR	0.59	0.03	0.06	0.76	0.90	0.97	0.84	0.90	0.97	0.95	0.78	0.74	0.90

Figure 5: Values of accuracy, loss in accuracy, loss in true positive rate, and various fairness measures obtained when using Algorithm 1 from the paper to optimize either FDR or SR.

2.3 Recommendations



(a) (b)

Figure 4: Algo-X denotes the version of Algorithm 1 from the paper used to optimize fairness metric X, where X = {FDR, SR}.

There are various scenarios in which this method may be useful. Firstly, it provides a single algorithm to optimize any linear-fractional fairness measure. Incidentally, most (if not all) fairness measures used in practice happen to be linear-fractional. Secondly, it can also optimize multiple fairness measures simultaneously, as the fairness metric q can be vector valued. Thirdly, it provides theoretical guarantees about being probably approximately correct, so it is more suitable for applications where safety is a concern. Finally, it also happens to be the first method to provide theoretical guarantee for predictive parity, and is a clear winner when this fairness measure needs to be optimized.

One crucial limitation of this method is that it requires computing an estimate of the data distribution. In addition to the modeling complexity introduced by this requirement, it also leads to a loss in accuracy on the task even when we use this method without optimizing any fairness measure. We also observed that tuning the parameter τ (lower bound on fairness measure) is sensitive in practice, with small changes leading to potentially large drops in accuracy.

3 Promoting Fairness through Hyperparameter Optimization

3.1 Overview of Method

The method of fairness through hyperparameter optimization is to create a fairer model in the postprocessing steps of the machine learning pipeline. Current bias mitigation approaches focus on certain phases of the ML process, such as data sampling and model training, and frequently only address one fairness criterion or group of ML models. As a result, these techniques are limited in the real word. The paper Andre F. Cruz [2021] uses an adaptive hyperband algorithm, weighting the loss with another fairness metric. The adaptive weights balance the hyperparameter search to still find a relatively optimal solution, balancing the real loss with the fairness to create a more fair prediction in the long run. The fairness metric balances disparate error rates across subgroups of the population. In this case, we use the recall of each poverty group to measure how well the model predicts for each subgroup, and take the minimum of these measure fairness. As a result, we try to boost the lowest recall among the subgroups.

3.2 Implementation

The method employed in this project is based off of hyperband, with a custom loss function and an adaptive α parameter. The standard implementation of Hyperband is used, where the number of resources are limited, and we use the idea of successive halving to zone in on the better sections we are exploring. This problem is a joint maximization problem, where it is defined as

$$\underset{\lambda \in \Lambda) G(\lambda)}{\arg\max} = (\rho(\lambda), \phi(\lambda))$$

where λ represents the hyperparameter configuration drawn from the whole hyperparameter space Λ. Both ρ and ϕ are functions that map Λ to $[0, 1]$ where ρ is the performance metric and ϕ is the fairness metric. The number of incomparable solutions can quickly dominate the size of the population, especially for higher dimensional problems. The paper notes that a l_p-norm is effective in scalarizing to reduce all objectives into a scalar. The weighted l_p-norm is defined as

$$\arg\max_{\lambda \in \Lambda} \|H(\lambda)\|_p = \left(\sum_{i=1}^{k} w_i h_i(\lambda)^p \right)^{\frac{1}{p}}$$

where the weights vector w induces some preference over the objectives $H(\lambda) = (h_1(\lambda), \ldots, h_k(\lambda))$. The paper notes that the simplification of this problem is generally convex and we should just consider the l_1 norm. Thus, we only optimize

$$g(\lambda) = \alpha \cdot \rho(\lambda) + (1 - \alpha) \cdot \phi(\lambda)$$

where α is between 0 and 1. This is now simply a linear interpolation between the fairness metric and the performance metric. Finally, there is a dynamic α weighting parameter, that is updated by the following equation:

$$\alpha = 0.5 \cdot (\bar{\phi} - \bar{\rho}) + 0.5$$

$\bar{\phi}$ and $\bar{\rho}$ are the mean performance and fairness metrics over the different subgroups. If the model prefers performance over fairness, the α will increase, thus favoring the fairness the next time around.

3.3 Implementing on Our Project

As for our project, we chose precision as the performance metric, while the fairness metric is the minimum recall across the poverty levels defined by the dataset.

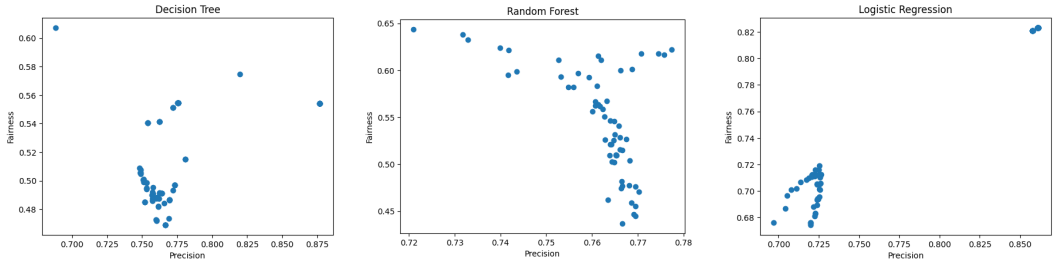


Figure 6: Results of Fair Hyperband Implementation

From fig. 6, we see that the fairness hyperband algorithm does a decent job in terms of producing fairer results while minimizing the tradeoff with precision. The models with more hyperparameters (decision tree, random forest) have a better Pareto Frontier, where the logistic regression model has a less obvious frontier. We see that we can get pretty comparable results in precision while maintaining equal levels of fairness.

3.4 Recommendations

Hyperparameter Optimization is a very flexible method which is designed to be used no matter the model or data. This method allows any model to be tuned to be more fair while trading off minimal amounts of performance. Any classical models with hyperparameters will be easy to tune under this framework, as it is just an adaption of hyperband. There are concerns of computation, as running large number of models repeatedly with different hyperparameter configurations is expensive. This is expected for most postprocessing solution, as we are fixing the model that already might not be fair. Overall, this method increases fairness by a significant amount while maintaining performance and is easy to implement.

References

Catarina Belem Carlos Soares Pedro Bizarro Andre F. Cruz, Pedro Saleiro. Promoting fairness through hyperparameter optimization. volume 29, 2021. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9679036&tag=1.

L. Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K. Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 319–328, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361255. doi: 10.1145/3287560.3287586. URL https://doi.org/10.1145/3287560.3287586.

Toon Calders Faisal Kamiran. Data preprocessing techniques for classification without discrimination. volume 33, page 33, 2011. URL https://link.springer.com/article/10.1007/s10115-011-0463-8.

Moritz Hardt, Eric Price, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/9d2682367c3935defcb1f9e247a97c0d-Paper.pdf.