7th ICML Workshop on Automated Machine Learning (2020)

Bayesian Optimization with Fairness Constraints

Valerio Perrone VPERRONE@AMAZON.COM
Michele Donini DONINI@AMAZON.COM
Krishnaram Kenthapadi KENTHK@AMAZON.COM
Cédric Archambeau CEDRICA@AMAZON.COM
Amazon Web Services

Abstract

Given the increasing importance of machine learning in our lives and the need for algorithmic fairness, several methods have been proposed to measure and mitigate biases in machine learning models. Commonly, these techniques are specialized approaches applied to a single type of model and a specific definition of fairness, limiting their effectiveness in practice. In this paper, we present a general constrained Bayesian optimization (BO) framework to optimize the performance of any black-box machine learning model while enforcing fairness constraints. BO is a class of global optimization algorithms that has been successfully applied to automatically tune the hyperparameters of machine learning models. We apply BO with fairness constraints to a range of popular models, including random forests, gradient boosting and neural networks, showing that we can obtain accurate and fair solutions by acting solely on the hyperparameters. We also show empirically that our approach is competitive with specialized techniques that explicitly enforce fairness constraints during training, and outperforms preprocessing methods that learn unbiased representations of the input data.

1. Introduction

With the increasing use of machine learning (ML) in domains such as lending, hiring, criminal justice, and college admissions, there has been a major concern for the potential for ML to unintentionally encode societal biases and result in systematic discrimination (Angwin et al., 2016; Barocas et al., 2018; Bolukbasi et al., 2016; Buolamwini and Gebru, 2018; Caliskan et al., 2017). For example, a classifier that is only tuned to maximize performance can unfairly predict a high credit risk for some subgroups of the population applying for a loan. Extensive work has been done to measure and mitigate biases during different stages of the ML life-cycle (Barocas et al., 2018).

In many practical ML settings, there is a need to optimize the performance of black-box ML models while enforcing fairness constraints. For example, several cloud platforms allow their customers to bring their own proprietary model training code and datasets, perform model training, and then tune the hyperparameters of the models treating them as a black-box (Golovin et al., 2017; Clark and Hayes, 2019). However, being tailored to specific models and fairness definitions, most existing fairness techniques are inapplicable to these settings.

Motivated by this, we present a general constrained Bayesian Optimization (BO) framework to tune the performance of black-box ML models while satisfying fairness constraints. We demonstrate its effectiveness on several classes of ML models, including random forests, gradient boosting, and neural networks, showing that we can obtain accurate and fair models simply by acting on their hyperparameters. Figure 1 illustrates this idea by plotting the ac-

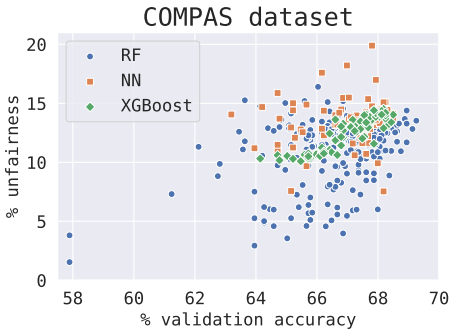

Figure 1: Unfairness-accuracy trade-off by varying the hyperparameters of XGBoost, RF and NN on the COMPAS benchmark. Each dot corresponds to a different hyperparameter configuration.

curacy and unfairness achieved by trained gradient boosted tree ensembles (XGBoost, Chen and Guestrin (2016)), random forests (RF) and a simple feedforward neural network (NN), with each dot corresponding to a random hyperparameter configuration. The key observation is that, given a level of accuracy, one can reduce unfairness just by tuning the model hyperparameters. As an example, for RF, paying 0.02 points of accuracy (-2.5%, from 0.69 to 0.67) can lead to a model with 0.08 fewer unfairness points (-70%, from 0.13 to 0.04).

Our approach supports arbitrary fairness constraints, allows for multiple constraints simultaneously, and is complementary to existing bias mitigation techniques. In experiments over a set of popular sensitive classification tasks, we show that it is more effective than pre-processing techniques that learn unbiased representations of the input data, and is competitive with methods that have access to the model internals and incorporate fairness constraints as part of the objective during model training.

2. Blackbox Bayesian Optimization

Bayesian optimization (BO) is a well-established methodology to optimize expensive blackbox functions (see Shahriari et al., 2016, for an overview). It relies on a probabilistic model of the unknown target $f(\mathbf{x})$ one wishes to optimize and which is repeatedly queried until one runs out of budget (e.g., time). Queries consist in evaluations of f at hyperparameter configurations $\mathbf{x}^1, \ldots, \mathbf{x}^n$ selected according to an explore-exploit trade-off criterion or *acquisition function* (Jones et al., 1998). The hyperparameter configuration corresponding to the best query is then returned. One popular approach is to impose a Gaussian process (GP) prior over f and, in light of the observed queries $f(\mathbf{x}^1), \ldots, f(\mathbf{x}^n)$, compute the posterior GP (Rasmussen and Williams, 2006). The posterior GP is characterized by a posterior mean function and a posterior variance function that are required when evaluating the acquisition function for each new query of f.

A widely used acquisition function is the Expected Improvement (EI) (Mockus et al., 1978). For a Gaussian predictive distribution, EI is defined in closed-form as the following: $EI(\mathbf{x}) = \mathbf{E}[\max(0, f(\mathbf{x}_{min}) - f(\mathbf{x}))] = \sigma^2(\mathbf{x})(z(\mathbf{x})\Phi_n(z(\mathbf{x})) + \phi_n(z(\mathbf{x})))$, where $z(\mathbf{x}) := \frac{\mu(\mathbf{x}) - f(\mathbf{x}_{min})}{\sigma^2(\mathbf{x})}$. μ and σ^2 are respectively the posterior GP mean and variance, and Φ_n and ϕ_n respectively the CDF and PDF of the standard normal. Alternative acquisition functions

based on information gain criteria have been developed (Hennig and Schuler, 2012). Standard acquisitions only focus on the objective $f(\mathbf{x})$ and do not account for additional constraints. In this work, we aim to optimize a black-box function $f(\mathbf{x})$ subject to fairness constraints $c(\mathbf{x}) \leq \epsilon$, $\epsilon \in \mathbb{R}$, with ϵ determining how strictly the fairness constraint should be enforced.

3. Fairness definitions

Today, there is no consensus on a unique definition of fairness, and some of the most common definitions are conflicting (Verma and Rubin, 2018). In our blackbox framework, we can incorporate different definitions. Our goal is not to introduce yet another fairness definition, but to propose a flexible methodology that is able to output fair models regardless of the selected criterion we want to enforce. Let Y be the binary label in $\{0, 1\}$, S the protected (or sensitive) attribute, and $\hat{Y}$ our predicted label. Let a, b be in $\{0, 1\}$. We summarize the most common definitions, grouping those that result in equivalent mathematical formalizations.

Predicted outcome given the true label. The probability of making a mistake is the same regardless of the value taken by the protected attribute $P(\hat{Y} = a|Y = b, S = 0) = P(\hat{Y} = a|Y = b, S = 1)$. For example, Equal Opportunity (EO) is part of this family, where we fix $a = 1$ and $b = 1$, i.e., $P(\hat{Y} = 1|Y = 1, S = 0) = P(\hat{Y} = 1|Y = 1, S = 1)$.

True label given predicted outcome. A different way of weighting mistakes, still requiring the probability of error being independent of the value taken by the protected attribute: $P(Y = a|\hat{Y} = b, S = 0) = P(Y = a|\hat{Y} = b, S = 1)$.

Predicted outcome only. The prediction is unaffected by the protected attribute, regardless of the actual true decision $P(\hat{Y} = a|S = 0) = P(\hat{Y} = a|S = 1)$. Statistical Parity (SP) is part of this family, where we fix $a = 1$, i.e., $P(\hat{Y} = 1|S = 0) = P(\hat{Y} = 1|S = 1)$.

Our goal is to have accurate models with a controlled (small) violation of the fairness constraint. Following Donini et al. (2018), we consider the family of the ϵ-fair models. A model $\hat{Y}$ is ϵ-fair if its violation of the fairness definition is less or equal to $\epsilon \geqslant 0$. In the case of EO, a model $\hat{Y}$ is ϵ-fair if the difference in EO (DEO) is smaller or equal to ϵ:

$$|P(\hat{Y} = 1|Y = 1, S = 0) - P(\hat{Y} = 1|Y = 1, S = 1)| \leqslant \epsilon. \tag{1}$$

In the case of SP, we can follow the same approach with the difference in SP (DSP):

$$|P(\hat{Y} = 1|S = 0) - P(\hat{Y} = 1|S = 1)| \leqslant \epsilon. \tag{2}$$

3.1 Black-box Algorithmic Fairness

Blackbox approaches to enforce fairness have been proposed in the literature, usually consisting in data pre-processing techniques. For example, Zemel et al. (2013) learn a fair representation of the data on top of the training procedure. Another common practice consists in applying the following two-step procedure (Kamiran and Calders, 2012): (i) removing the sensitive attribute from the feature set; (ii) rebalancing the dataset, i.e., increasing the number of observations using synthetic oversampling by using SMOTE (Chawla et al., 2002). These methodologies are blackbox, but the hyperparameters of the underlying base methods

Algorithm 1 Fair Bayesian Optimization (FairBO)

Input: Initial design budget M, total budget N, unfairness upper bound ϵ.
Evaluate $f(\mathbf{x}_i)$ and $c(\mathbf{x}_i)$ for $i = 1, ..., M$ random hyperparameters $\mathbf{x}_i$ from the search space.
Define set of evaluated hyperparameters $\mathcal{C} = \{(\mathbf{x}_i, f(\mathbf{x}_i), c(\mathbf{x}_i)\}_{i=1}^{M}$
Update constraint and objective models based on $\mathcal{C}$.
while $i < N - M$ **do**
  $\mathbf{x}_{\text{new}} = \arg\max_{\mathbf{x}} EI(\mathbf{x})P(c(\mathbf{x}) \leq \epsilon)$.
  Evaluate $f(\mathbf{x}_{\text{new}})$ and $c(\mathbf{x}_{\text{new}})$.
  Update $\mathcal{C} = \mathcal{C} \cup \{(\mathbf{x}_{\text{new}}, f(\mathbf{x}_{\text{new}}), c(\mathbf{x}_{\text{new}}))\}$
  Update objective and constraint models based on $\mathcal{C}$.
  $i = i + 1$
end while
return Best fair hyperparameter configuration in $\mathcal{C}$.

still need to be tuned for performance and, as we will show shortly, can have an impact on the degree of fairness obtained. Another possible approach is described in Agarwal et al. (2018), where a fair classification task is reduced to a sequence of cost-sensitive classification problems. The solutions to these problems yield a randomized classifier with a low empirical error subject to fairness constraints. This is a blackbox method with respect to the base model, but still needs specific implementations based on the fairness definition at hand and outputs an ensemble of models.[1] In contrast, we show that our constrained BO approach is agnostic to the selected fairness constraint.

4. Fair Bayesian Optimization

The most established technique to extend BO to the constrained case is the constrained EI (cEI) (Gardner et al., 2014; Gelbart et al., 2014; Snoek et al., 2015). An additional GP is placed on the constraint and the EI weighted with the posterior probability $P(\mathbf{x})$ of the constraint being satisfied, giving $cEI(\mathbf{x}) = P(\mathbf{x})EI(\mathbf{x})$. In our setting, feasible hyperparameter configurations are those satisfying the desired fairness constraint (e.g., the difference in statistical parity across subgroups should be lower than 10%). We define $cEI(\mathbf{x})$ with respect to the current *fair* best, which may not be available in the first iterations. Therefore, we start by greedily optimizing $P(\mathbf{x})$ and then switch to $cEI(\mathbf{x})$ when the first fair hyperparameter configuration is found. Algorithm 1 describes FairBO, the approach to optimize the hyperparameters of a blackbox function while satisfying an arbitrary fairness constraint. The procedure is straightforward to extend to handle multiple definitions of fairness simultaneously: $P(\mathbf{x})$ becomes a product, each term being the probability of satisfying each fairness definition. FairBO can also be implemented through alternative, entropy-based acquisition functions (Hernández-Lobato et al., 2015; Perrone et al., 2019). We leave this for future work.

1. Fairlearn code at: `https://github.com/fairlearn/fairlearn`.

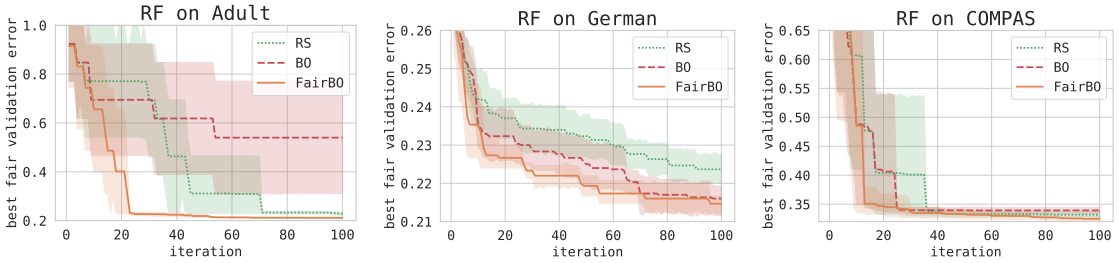

Figure 2: Comparison of RS, BO, and FairBO over the validation error of the best feasible solution, tuning RF. The fairness constraint is DSP ≤ 0.05.

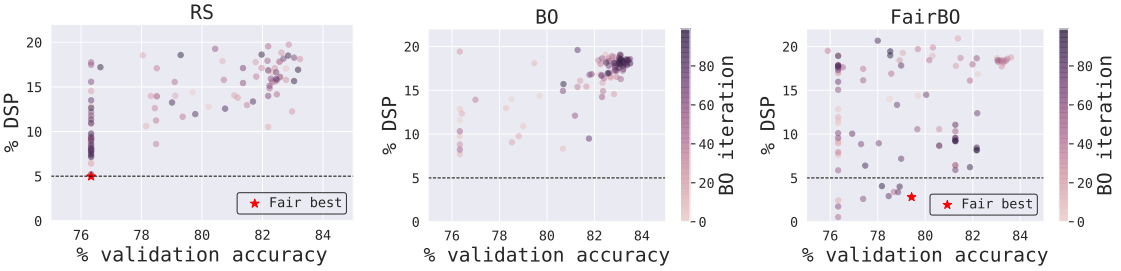

Figure 3: Comparison of RS, BO, and FairBO on the task of tuning RF on Adult. The horizontal line is the fairness constraint, set to DSP ≤ 0.05.

5. Experiments

We consider 3 benchmark datasets widely used in the context of fairness: (i) Adult – Census Income (Dua and Graff, 2017), a binary classification task with binary gender as sensitive attribute, where the task is to predict if income exceeds \$50K/yr based on census data; (ii) German Credit Data (Dua and Graff, 2017), a binary classification problem with binary gender as sensitive attribute, where the goal is to classify people described by a set of attributes as good or bad credit risks; (iii) COMPAS, a binary classification problem concerning recidivism risk, with binarized ethnic group as sensitive attribute (one group for "white" and one for all other ethnic groups).[2] We tune 4 popular ML algorithms implemented in `scikit-learn` (Pedregosa et al., 2011): XGBoost, Random Forest (RF), a simple feedforward neural network (NN), and Linear Learner (LL), optimizing the hyperparameters listed in Appendix A. We optimize for validation accuracy, with a random 70%/30% split into train/validation. We place an upper bound on DSP (inequality (2)). We obtained analogous results with DEO (inequality (1)), noting that other fairness definitions can be plugged in.

5.1 Results

We first compare FairBO to Random Search (RS) and to standard BO based on the expected improvement. Figure 2 compares the validation error of the fair solution on Adult, COMPAS, and German data, tuning RF as a base model (analogous results are obtained on XGBoost,

2. COMPAS link: `https://github.com/propublica/compas-analysis`.

Method	Adult	German	COMPAS
FERM	0.164 ± 0.010	0.185 ± 0.012	0.285 ± 0.009
Zafar	0.187 ± 0.001	0.272 ± 0.004	0.411 ± 0.063
FERM preprocess	0.228 ± 0.013	0.231 ± 0.015	0.343 ± 0.002
SMOTE	0.178 ± 0.005	0.206 ± 0.004	0.321 ± 0.002
FairBO (ours)	0.175 ± 0.007	0.196 ± 0.005	0.307 ± 0.001

Table 1: Validation error of the best fair models for model-specific (first two rows) and model-agnostic fairness methods. Fairness is defined as DSP $<= 0.1$.

NN and LL). All methods are initialized with 5 random hyperparameter configurations and results are averaged across 10 repetitions, with 95% confidence intervals obtained via bootstrapping. The fairness constraint is DSP $\leqslant 0.05$. As expected, FairBO finds a well-performing fair model more quickly than EI and RS. Figure 3 shows that standard BO with EI can get stuck in high-performing yet unfair regions, failing to return a feasible solution. RS is more robust than standard BO, but needs significantly higher resources to find an accurate and fair solution (e.g., five times slower than constrained BO on Adult).

In the context of algorithmic fairness several *ad-hoc* methods have been proposed. We compare to the method of Zafar (Zafar et al., 2017) and Fair Empirical Risk Minimization (FERM) (Donini et al., 2018), the state-of-the-art to yield fair linear models.[3] These methods enforce fairness during training and optimize the linear model parameters to make it both fair and accurate with respect to a fixed fairness definition. In this sense these are not black-box approaches, and to be able to compare to them we apply FairBO to LL. As alternative black-box approaches we compare to SMOTE, which preprocesses data by removing the sensitive feature and rebalancing observations, and FERM preprocess (Donini et al., 2018), which learns a fair representation of the data before fitting a linear model. We allocate 100 hyperparameter tuning iterations for all approaches.

Table 1 shows the best fair model found by FairBO on LL compared to the best fair model found by each baseline. As expected, FERM achieves higher accuracy, due to the constraint applied directly while training the parameters (as opposed to the hyperparameters) of the linear model. On the other hand, the performance gap with FairBO is modest, and FairBO outperforms Zafar. While simple, we found that constrained BO is a competitive baseline that can outperform or compete against highly specialized techniques. FairBO is also the best model-agnostic method, outperforming both SMOTE and FERM preprocessing. Constrained BO is thus more effective than pre-processing techniques, which remove bias from the data at a high price in accuracy.

6. Conclusions

We showed that tuning model hyperparameters is a surprisingly effective strategy to mitigate unfairness in ML, and proposed a constrained BO framework to jointly tune ML models for accuracy and fairness. On top of being model agnostic, constrained BO works with arbitrary fairness definitions and allows for any type of fairness constraints. The proposed methodology empirically finds more accurate fair solutions than data-debiasing techniques, while being competitive with state-of-the-art algorithm-specific fairness techniques.

3. Code for Zafar from `https://github.com/mbilalzafar/fair-classification`.

References

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudik, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69, 2018.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica*, 2016.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. Fairness and machine learning. fairmlbook. org, 2018. *URL: http://www.fairmlbook.org*, 2018.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In *NIPS*, 2016.

Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *FAT**, 2018.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334), 2017.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16: 321–357, 2002.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, 2016.

Scott Clark and Patrick Hayes. SigOpt Web page. 2019. URL `https://sigopt.com`.

Michele Donini, Luca Oneto, Shai Ben-David, John S Shawe-Taylor, and Massimiliano Pontil. Empirical risk minimization under fairness constraints. In *Advances in Neural Information Processing Systems*, pages 2796–2806, 2018.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Jacob Gardner, Matt Kusner, Zhixiang Xu, Kilian Weinberger, and John Cunningham. Bayesian optimization with inequality constraints. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 937–945, 2014.

Michael A. Gelbart, Jasper Snoek, and Ryan P. Adams. Bayesian optimization with unknown constraints. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 250–259, 2014.

D. Golovin, B. Solnik, S. Moitra, G. Kochanski, J. E. Karro, and D. Sculley. Google Vizier: A service for black-box optimization. In *KDD*, 2017.

P. Hennig and C. J. Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(1):1809–1837, 2012.

José Miguel Hernández-Lobato, Michael A. Gelbart, Matthew W. Hoffman, Ryan P. Adams, and Zoubin Ghahramani. Predictive entropy search for Bayesian optimization with unknown constraints. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1699–1707, 2015.

F. Hutter, H. Hoos, and K. Leyton-Brown. An efficient approach for assessing hyperparameter importance. In *Proceedings of International Conference on Machine Learning (ICML)*, page 754–762, June 2014.

Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.

Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of Bayesian methods for seeking the extremum. *Towards Global Optimization*, 2(117-129):2, 1978.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research (JMLR)*, 12:2825–2830, 2011. ISSN 1532-4435.

Valerio Perrone, Iaroslav Shcherbatyi, Rodolphe Jenatton, Cedric Archambeau, and Matthias Seeger. Constrained Bayesian Optimization with Max-Value Entropy Search. *arXiv preprint arXiv:1910.07003*, 2019.

Carl Rasmussen and Chris Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando de Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

Jasper Snoek, Oren Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Mostofa Patwary, Mr Prabhat, and Ryan Adams. Scalable Bayesian optimization using deep neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2171–2180, 2015.

Sahil Verma and Julia Rubin. Fairness definitions explained. In *IEEE/ACM International Workshop on Software Fairness (FairWare)*, pages 1–7, 2018.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pages 962–970, 2017.

Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333, 2013.

Appendix A. Optimized hyperparameters

A.1 Algorithms

We considered the problem of tuning four popular ML algorithms: XGBoost (XGB), random forest (RF), feedforward neural network (NN), linear learner (LL). In this section, we give more details on the search space over which each hyperparameter was optimized.

XGBoost We consider a 7-dimensional search space: number of boosting rounds in {1, 2, ..., 256} (log scaled), learning rate in [0.01, 1.0] (log scaled), minimum loss reduction to partition leaf node `gamma` in [0.0, 0.1], L1 weight regularization `alpha` in [1e-3, 1e+3] (log scaled), L2 weight regularization `lambda` in [1e-3, 1e+3] (log scaled), subsampling rate in [0.01, 1.0], maximum tree depth in {1, 2, ..., 16}.

Random Forest We consider a 4-dimensional search space: number of trees in {1, 2, ..., 64} (log scaled), tree split threshold in [0.01, 0.5] (log scaled), tree maximum depth in {1, 2, 3, 4, 5}, criterion for quality of split in {Gini, Entropy}.

NN We consider an 11-dimensional search space: number of layers in {1, 2, 3, 4}, each layer size in {2, 3, ..., 32} (log scaled), activation in {Logistic, Tanh, ReLU}, tolerance in [1e-5, 1e-2] (log scaled), L2 regularization in [1e-6, 0.1] (log scaled), and Adam parameters: initial learning rate `eps` in [1e-6, 1e-2] (log scaled), beta1 and beta2 in [1e-3, 0.99] (log scaled).

Linear Learner We consider a 6-dimensional search space: iteration count in {1, 2, ..., 128}, regularization type in {L1, L2, ElasticNet}, Elastic Net mixing parameter in [0, 1], regularization factor `alpha` in [1e-3, 1e3] (log scaled), initial learning rate `eta0` in [1e-4, 1e-1] (log scaled), learning rate schedule in {Constant, Optimal, Invscaling, Adaptive}.

A.2 Baselines

In addition to the hyperparameters of the tuned algorithms, we allocated 100 iterations of random search to tune each baseline, namely SMOTE, Zafar and FERM.

SMOTE In addition to the 6 Linear Learner's hyperparameters, we jointly tuned 2 hyperparameters controlling the degree of dataset rebalancing: oversampling rate of the less frequent class in [0.3, 1.0] and number of neighbors to generate synthetic examples in {1, 2, ..., 20}.

FERM preprocess FERM preprocessing learns a fair representation of the dataset, which is then fed to Linear Learner. Hence, we tuned the same six hyperparameters as per the standard Linear Learner.

Zafar We tuned 3 hyperparameters: L1 regularization coefficient in [0.001, 10.0] (log scaled), L2 regularization coefficient in [0.001, 10.0] (log scaled), and Zafar's epsilon-fairness threshold in [0.0, 1.0].

FERM We consider 2 hyperparameters for FERM. The L2 regularization coefficient C in [0.001, 10.0] (log scaled), and FERM's epsilon-fairness threshold in [0.0, 1.0].

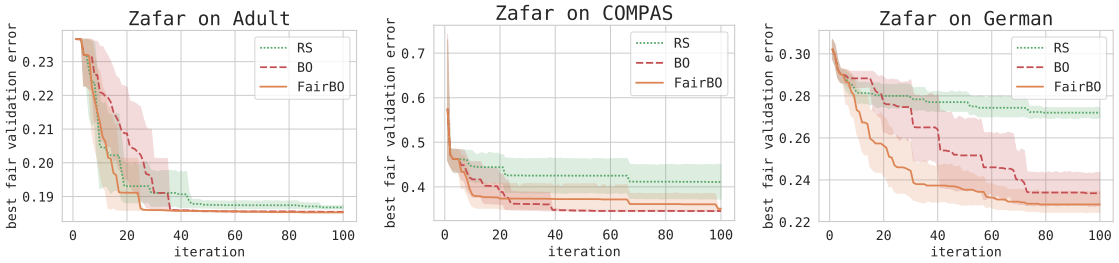

Figure 4: Zafar on Linear Learner. Compared are random search, BO and FairBO. The fairness constraint is DSP ≤ 0.1.

Appendix B. Additional experiments

B.1 Zafar with FairBO

By focusing on the hyperparameter level, FairBO is also a complementary approach to enforce fairness on top of model-specific techniques. These techniques often come with their own hyperparameters and their tuning can still negatively impact the fairness of the resulting solution. We demonstrate this idea by combining FairBO and Zafar in Figure 4. The results show that FairBO finds a better performing fair solution, and does so more quickly than random search and standard BO (EI). On top of being a competitive baseline, FairBO can be then naturally plugged on top of alternative fairness techniques, improving the accuracy of the final solution.

B.2 Hyperparameters and unfairness

We showed that tuning the hyperparameters can effectively mitigate unfairness while optimizing performance. In this section, we investigate more closely the role of each tuned hyperparameter on the unfairness of the resulting model. For every tuned algorithm, we apply fANOVA (Hutter et al., 2014) to study hyperparameter importance with respect to the fairness metric, fixed as statistical parity. Hyperparameter configurations and unfairness metrics are collected from 100 iterations of random search and 10 random seeds, for a total of 1000 data points per algorithm-dataset pair. The results are showed in Figure 5. It emerges that hyperparameters controlling the regularization level tend to have the largest impact on fairness.

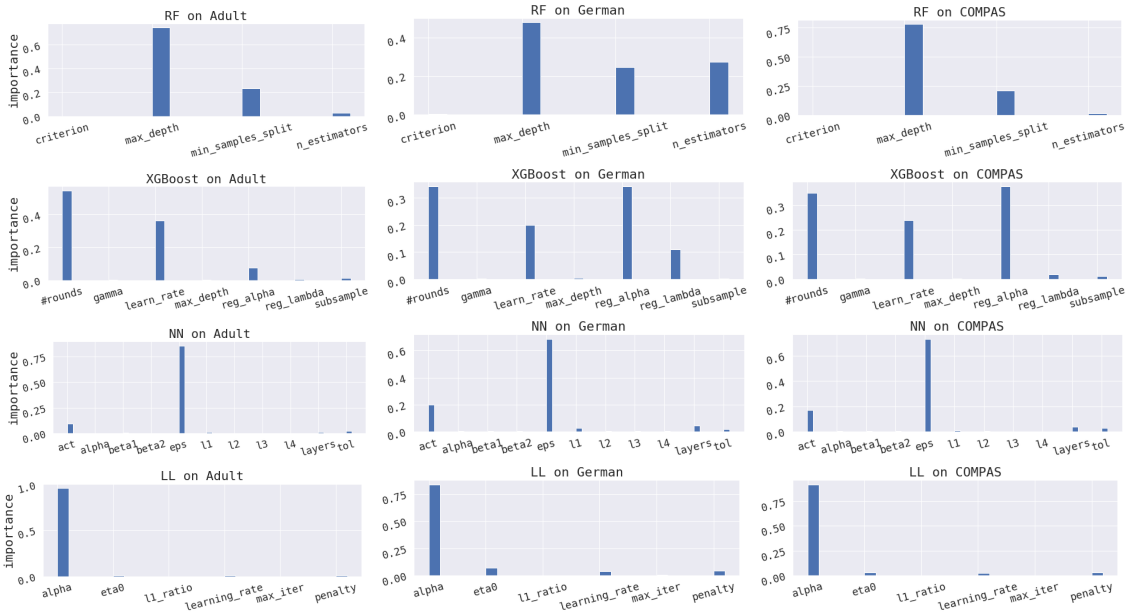

Figure 5: Hyperparameter importance on fairness.