FixOut: an ensemble approach to fairer models

Guilherme Alves, Vaishnavi Bhargava, Fabien Bernier, Miguel Couceiro, Amedeo Napoli

FixOut: an ensemble approach to fairer models*

Guilherme Alves[1] [0000-0002-5004-4429], Vaishnavi Bhargava[1], Fabien Bernier[1],
Miguel Couceiro[1] [0000-0003-2316-7623], and Amedeo Napoli[1]

Université de Lorraine, CNRS, Inria Nancy G.E., LORIA
Vandoeuvre-les-Nancy, France
{guilherme.alves-da-silva, fabien.bernier, miguel.couceiro,
amedeo.napoli}@loria.fr, vaishnavi.bhargava2605@gmail.com

Abstract. In this paper, we address the question of process and model fairness. We propose FixOut, a human-centered and model-agnostic framework, that uses any explanation method (based on feature importance) to assess model's reliance on sensitive features. Given a pre-trained classifier, FixOut first checks whether it relies on user defined sensitive features. If it does, then FixOut employs feature dropout to produce a pool of simplified classifiers that are then aggregated into an ensemble classifier. We present empirical results using different models on several real-world datasets, that show a consistent improvement in terms of widely used fairness metrics, decreased reliance on sensitive features, and without compromising the classifier's accuracy.

Keywords: SHAP · LIME · Fairness metrics· Feature importance · Feature-dropout · Ensemble classifier.

1 Introduction

Machine Learning (ML) models are increasingly present in decision support systems with critical societal impacts, for instance, in job recruitment, loan applications and criminal recidivism prediction. In spite of the objective character of these algorithmic decisions, recent studies raised fairness concerns by revealing discriminating outcomes against minorities and unprivileged groups[1][2] [9, 2]. In 2016, the European Union has enforced the GDPR Law[3] across all organizations and firms. The law entitles European citizens the right to have a basic knowledge w.r.t. the inner workings of ML models and their outcomes.

Two main approaches have been proposed to address algorithmic (un)fairness based on decision outcomes. One is to use fairness measures and impose fairness

* This research was partially supported by TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA No 952215, and the Inria Project Lab "Hybrid Approaches for Interpretable AI" (HyAIAI)

[1] https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G

[2] https://www.bbc.com/news/business-50365609

[3] General Data Protection Regulation: https://gdpr-info.eu/

constraints during training [21, 20] whereas the other aims to reduce the reliance of ML models on salient or sensitive features [8, 4, 1]. For the latter, a natural approach is to train models on datasets with these sensitive features removed, however this may entail a reduction in the model's accuracy [21].

Following the same tracks, Bhargava et al. [4] proposed a human centered, model-agnostic framework called LimeOut to reduce classifiers' reliance on sensitive features without compromising their accuracy. Essentially, LimeOut receives a triple (M, D, F) consisting of a classifier M, a dataset D and a set F of sensitive features, as input, and it outputs a classifier M' that is less reliant on the sensitive features in F. To assess the reliance of M on the sensitive features in F, LimeOut uses a *global* variant of LIME explanations [15]. If sensitive features are shown to contribute globally to M's outcomes, then M is deemed unfair. In this case, LimeOut employs *feature dropout* to build a pool of classifiers that are then aggregated to obtain an ensemble classifier. Otherwise, M is deemed fair, and no action is taken.

Empirical studies [4, 1] showed that LimeOut's ensemble models are less dependent on sensitive features, and with improved (or, at least, maintained) accuracy when compared to the original models. However, several issues concerning the use of explanation methods for assessing process fairness have been recently raised. For instance, [5, 19] questioned the usefulness of explanations to assess fairness by showing that it is possible to perform "adversarial attacks" to modify explanations in order to conceal unfairness issues. This led to a thorough empirical investigation [1] beyond process fairness, and where LimeOut showed consistent improvements with respect to widely used fairness metrics such as disparate impact, equal opportunity, demographic parity, equal accuracy, and predictive equality. In [1] it was also claimed the adaptability of LimeOut to other data types as well as to other explanation methods. This is particularly relevant given the drawbacks of LIME explanations that have been pointed out in the literature [6, 14].

In this paper we tackle the latter issues by showing that LimeOut can be adapted to different explanation methods. More precisely, we propose FixOut[4], an explainer-agnostic framework that generalizes LimeOut. To illustrate, we consider FixOut instantiated by SHAP[5] [13], an explanation method that is based on coalitional game theory, to assess model fairness. Also, instead of a simple average as aggregation rule, FixOut employs a weighted average that takes into account the global contribution of sensitive features, to construct the final ensemble model.

The main contributions of this paper are thus the following: (1) the introduction of the FixOut framework, which is explainer-agnostic, (2) the consideration of model ensembles that take into account global contributions of sensitive features, and (3) an empirical study of FixOut on different datasets and with respect to several fairness metrics, that illustrate the adaptability of FixOut to different explanation methods.

[4] FixOut stands for **Fa**I**r**ness through e**X**planations and feature drop**Out**.

[5] SHAP stands for **SH**apley **A**dditive ex**P**lanations.

2 Related Work

In this section we recall the main concepts used in this work. We start with the notions of fairness and then we describe SHAP explanations which is used to assess fairness in our framework FixOut.

2.1 Assessing Model Fairness

Fairness of ML models can be addressed in several ways, but most fairness notions focus on models' outcomes. In this setting, there are two main approaches: one that proposes certain *fairness metrics* [20], while the other focuses on *process fairness* that assesses, for instance, the model's reliance on discriminatory or sensitive features [7], such as race, ethnicity, gender, or sexual orientation.

Fairness metrics usually rely on well known scores measured with respect to privileged and unprivileged groups. For instance, with respect to "race", white people are usually the privileged group and the nonwhites the unprivileged group. Among the best known fairness metrics, we will consider the following ones: *demographic parity* (DP) [10] that relies on predicted positive rates, *disparate impact*[6] (DI) [20] consider the proportion of these positive rates, *equal opportunity*[7] (EO) that is based on recall scores [20], and *equal accuracy* (EA) relies on accuracy scores [10]. In addition, we will also use *predictive equality* (PE) that assesses fairness based on false positive rates [1].

2.2 Explanations to assess fairness: the case of SHAP

Several works have been advocating that explanations can be used to assess model fairness since they provide insights into ML models' outcomes. Explanation methods differ mainly in the form of explanations or in the approach they use to generate them. For instance, Anchors provides rule-based explanations [16], while LIME [15], SHAP [12] and DeepLIFT [18] explain the outcome for a given instance by computing the contributions of feature to the outcome. In this paper, we focus on model-agnostic explanation methods, such as LIME and SHAP. As LIME has been already considered [1, 4], in this paper we will mainly use SHAP explanations.

SHapley **A**dditive ex**P**lanations [13] is a local model-agnostic explanation method based on coalitional game theory. SHAP provides explanations in the form of a linear surrogate model that (unlike LIME) is defined on a simplified representation space (a "coalition" of simplified features), and whose coefficients correspond to the contributions of the corresponding (selected) features. In the case of SHAP these coefficients coincide with Shapley values [17]. In this work, we focus on KernelSHAP [13] that is a variant of SHAP. KernelSHAP receives as input an instance x, the prediction model f, and the number of coalitions m. It then learns a linear model g defined on a simplified subset of features

[6] It is also referred to as *group fairness*

[7] It is also referred to as *disparate mistreatment*

("coalition" that defines the representation space) by optimizing the following loss function:

$$L(f, g, \pi_x) = \sum_{z \in Z} [f(h_x(z)) - g(z)]^2 \pi_x(z),$$

where $h_x(z)$ converts z from the representation space to the feature space, Z is a set of points that are representations of neighbors of x in the representation space, and $\pi_x(z)$ is the kernel defined as:

$$\pi_x(z) = \frac{M-1}{\binom{M}{|z|} |z| (M - |z|)},$$

where $|z|$ is the number of present features in the coalition z and M is the maximum coalition size.

KernelSHAP first samples coalitions of features and it then asks for prediction of each coalition[8]. This produces a new dataset of coalitions along with predictions which is used by KernelSHAP to fit a linear model g as described in the formula:

$$g(z) = \phi_0 + \sum_{j=1}^{M} \phi_j z_j,$$

where z_j indicates the presence/absence of the j-th feature.

To illustrate, let us consider the example of the Adult dataset where the goal is to predict if a person earns ≥50k dollars a year. Figure 1 presents a SHAP explanation for a prediction using Logistic Regression classifier, where the Shapley value for "Capital Gain = 2,174" is around -0.15 that indicates this feature contribute to move the prediction towards the negative class.

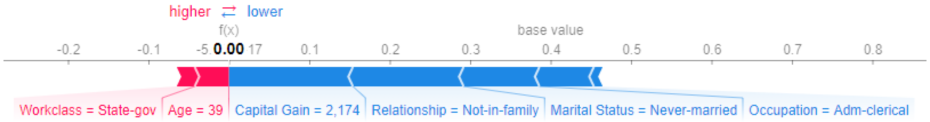

Fig. 1. SHAP explanation of the prediction of an instance in the Adult dataset.

3 Fairness Through Explanations and Feature Dropout

In this section, we introduce our proposed framework FixOut and highlight the differences between FixOut and LimeOut. Similarly to LimeOut [1, 4],

[8] Before asking for predictions, KernelSHAP converts a coalition z from the representation space to the original space using $h_x(z)$.

FixOut has two main components: $\text{Exp}_{\text{Global}}$ that provides global explanations[9] in order to assess fairness of a given pre-trained model M, and $\text{EnsembleOut}_{\text{Out}}$ that builds a fairer model M_{final} if M is deemed unfair. However, unlike LimeOut, FixOut receives as input a quadruple (M, D, F, E) where M is a pre-trained model, D is a dataset, F is a set of sensitive features, and E is an explanation method based on feature importance.

FixOut's workflow can be summarized as follows. Given (M, D, F, E), FixOut applies the component $\text{Exp}_{\text{Global}}$ using E as the explanation method. For instance, it can employ either SHAP or LIME to measure feature importance and so to evaluate the dependence of M on sensitive features. The output of $\text{Exp}_{\text{Global}}$ is a list F^k of the k most important features $a_1, a_2, \ldots, a_k$. As in LimeOut, FixOut applies the following rule to decide whether M is fair: if F^k contains sensitive features $a_{j_1}, a_{j_2}, \ldots, a_{j_i}$ in F with $i > 1$, then M is deemed unfair and the FixOut's second component applies; otherwise, it is considered fair and no action is taken.

In the former case (i.e., M is considered unfair), FixOut employs *feature dropout* [4] and uses the i features $a_{j_1}, a_{j_2}, \ldots, a_{j_i} \in F$ to build a pool of $i+1$ classifiers in the following way: for each $1 \leq t \leq i$, FixOut trains a classifier M_t after removing a_{j_t} from D, and an additional classifier M_{i+1} trained after removing all sensitive features F from D. As in LimeOut, this pool of classifiers is used to construct an ensemble classifier M_{final}. However, instead of a simple average, FixOut employs a weighted average using weights that take into account feature's contributions. Let $c'_{j_t} \in [0,1]$ be the normalized[10] global feature contribution associated with a_{j_t}, and define the weights w_t of M_t and the weight w_{i+1} of M_{i+1} as

$$w_t = \frac{c'_{j_t}}{1 + \sum_{u=1}^{i} c'_{j_u}}, \ 1 \leq t \leq i, \quad \text{and} \quad w_{i+1} = \frac{1}{1 + \sum_{u=1}^{i} c'_{j_u}}.$$

For a data instance x and a class C, the ensemble classifier M_{final} uses the following rule to predict the probability of x being in class C,

$$P_{M_{final}}(x \in C) = \sum_{t=1}^{i+1} w_t P_{M_t}(x \in C), \tag{1}$$

where $P_{M_t}(x \in C)$ is the probability predicted by model M_t.

[9] Essentially, we use sampling techniques to choose representative data instances: LimeOut uses submodular pick [15], whereas FixOut uses a simple bootstrapping approach available in the implementation of SHAP [11]. These are then aggregated to obtain a global ranking of feature contributions.

[10] We standardize feature contributions by $c'_{jt} = \frac{c_{j_t} - min(F^k)}{max(F^k) - min(F^k)}$, where $min(F^k)$ and $max(F^k)$ are the lowest and the highest feature contribution among F^k, respectively.

4 Experiments

In this section, we briefly present the datasets and the experimental setting that we used to perform our experiments. We then examine the obtained results in the following way. First, we report and compare the obtained accuracy on several classifiers and on FixOut's ensembles in Subsection 4.2. We then assess process fairness using SHAP explanations in Subsection 4.3. Finally, we evaluate fairness using standard metrics in Subsection 4.4.

4.1 Datasets & Experimental Setup

The experiments were conducted on 5 datasets used in [1], namely, German[11], Adult[12], HMDA[13], LSAC[14], and Default[15]. All datasets share common characteristics that allow us to run our experiments: a binary target feature and the presence of sensitive features. Table 1 summarizes basic information about these datasets.

Table 1. Datasets employed in the experiments.

Dataset	# features	# instances	Sensitive features
German	20	1000	"statussex", "telephone", "foreign worker"
Adult	14	32561	"MaritalStatus", "Race", "Sex"
HMDA	28	92793	"sex", "race", "ethnicity"
LSAC	11	26551	"race", "sex", "family_income"
Default	23	30000	"sex", "marriage"

We split each dataset into 70% training set and 30% testing. As the datasets are imbalanced, we used Synthetic Minority Oversampling Technique (SMOTE[16]) over training data to generate the samples synthetically. We train original and FixOut's ensemble models on the balanced (augmented) datasets using five classifiers. We used Scikit-learn implementation of the following algorithms: AdaBoost(**ADA**), Bagging (**BAG**), Random Forest (**RF**), and Logistic Regression (**LR**). We kept the default parameters of Scikit-learn documentation. In order to estimate Shapley values faster, especially in the presence of continuous features, we use K-means clustering with the number of clusters $n = 10$ to reduce feature domains; otherwise, the full domain is considered.

[11] https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)

[12] http://archive.ics.uci.edu/ml/datasets/Adult

[13] https://www.consumerfinance.gov/data-research/hmda/

[14] http://www.seaphe.org/databases.php

[15] https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients

[16] https://machinelearningmastery.com/threshold-moving-for-imbalanced-classification/

4.2 Accuracy Assessment

Table 2 shows the average accuracy obtained throughout the performance of the same experiment 30 times. For each dataset, we have the average accuracy of the original model and of the FixOut ensemble model. We also have the level of statistical significance. We performed the t-test to assess whether the average accuracy of original and FixOut's ensemble models are statistically different, with the exception of the Default dataset for which bagging, random forest and logistic regression were considered fair, whereas the improvement in the case of AdaBoost was negligible as discussed below.

Our analysis is based on the comparison between the accuracy of original and ensemble models. It is evident that FixOut ensemble models improve (or at least maintain) the accuracy level compared to original models. We can also observe that experiments in which FixOut's ensemble models improve in accuracy have a level of significance $p < 0.05$ (see experiments with one or two stars). The differences are not statistically significant in the following cases: Logistic Regression on the German, the Adult and the HMDA datasets, and AdaBoost on the LSAC dataset. We only notice an improvement in accuracy in Random Forest on the German dataset. Moreover, there was a slight improvement in the case of AdaBoost on the Default dataset (0.817 vs 0.819) but the level of significance was greater than 0.05, and thus we do not consider it statistically significant.

Table 2. Average accuracy assessment, where FixOut stands for the ensemble model built by our proposed framework. Numbers in parentheses indicate standard deviation. Hyphen indicates no accuracy values are reported (no statistical test is performed either). Stars indicate level of statistical significance, $**$ means $p < 0.001$, $*$ means $p < 0.05$, and no stars indicates $p > 0.05$.

		ADA	BAG	RF	LR
German	Original	0.754 (.017)	0.742 (.021)	0.765 (.018)	0.764 (.020)
	FixOut	0.758 (.018)	0.761 (.018)	0.766 (.014)	0.761 (.020)
	t-test	*	*		
Adult	Original	0.854 (.003)	0.841 (.003)	0.846 (.003)	0.807 (.006)
	FixOut	0.856 (.003)	0.845 (.003)	0.848 (.003)	0.805 (.003)
	t-test	**	**	*	
HMDA	Original	0.880 (.001)	0.883 (.001)	0.882 (.001)	0.878 (.001)
	FixOut	0.880 (.001)	0.884 (.001)	0.883 (.001)	0.878 (.001)
	t-test		**	**	
LSAC	Original	0.857 (.003)	0.860 (.003)	0.853 (.003)	0.818 (.005)
	FixOut	0.857 (.003)	0.862 (.002)	0.858 (.003)	0.820 (.004)
	t-test		**	**	*

4.3 Process Fairness Assessment

We now address process fairness, namely, the reliance of FixOut's ensemble outputs on sensitive features. To demonstrate the ability of FixOut to reduce

Table 3. Global explanation of RF on German dataset.

Original (SHAP)		Ensemble (SHAP)	
Feature	**Contrib.**	**Feature**	**Contrib.**
residencesince	9.124647	property	7.867011
job	-8.293296	credithistory	7.290505
statussex	-6.704196	residencesince	7.122002
existingchecking	-6.659944	job	-6.649668
savings	6.598886	installmentrate	5.405495
purpose	6.567743	existingchecking	-4.993191
property	6.454444	existingcredits	4.630803
telephone	5.69921	duration	3.574884
housing	-4.141756	otherinstallmentplans	-3.503103
installmentrate	3.990249	housing	-3.492053

Table 4. Global explanation of AdaBoost on Adult dataset.

Original (SHAP)		Ensemble (SHAP)	
Feature	**Contrib.**	**Feature**	**Contrib.**
MaritalStatus	-2.570228	Relationship	-2.487373
Education	1.841389	Education	1.895012
Age	1.823791	Age	1.602135
Hoursperweek	1.449913	Hoursperweek	1.406758
Relationship	1.145855	CapitalGain	-0.85136
CapitalGain	-0.841991	Occupation	-0.613416
Occupation	-0.572992	**Sex**	0.251091
Sex	0.453937	CapitalLoss	-0.159068
Education-Num	-0.294207	**MaritalStatus**	-0.156663
CapitalLoss	-0.158168	Education-Num	0.121778

Table 5. Global explanation of Bagging on HMDA dataset.

Original (SHAP)		Ensemble (SHAP)	
Feature	**Contrib.**	**Feature**	**Contrib.**
derived_loan_product_type	131.970375	derived_loan_product_type	152.401853
intro_rate_period	55.185479	intro_rate_period	78.719783
derived_race	-35.760327	applicant_credit_score_type_desc	29.645684
derived_sex	12.466373	loan_to_value_ratio	21.613061
applicant_credit_score_type	12.222446	loan_amount	8.680493
debt_to_income_ratio	6.906509	applicant_credit_score_type	7.692752
applicant_age_above_62	6.140392	debt_to_income_ratio	7.539624
loan_amount	5.489757	loan_term	-7.261209
loan_to_value_ratio	5.421845	income	5.350675
income	5.189954	applicant_age_above_62	4.415154

Table 6. Global explanation of Bagging on LSAC dataset.

Original (SHAP)		Ensemble (SHAP)	
Feature	**Contrib.**	**Feature**	**Contrib.**
zfygpa	21.085383	DOB_yr	-15.453783
zgpa	19.393242	zgpa	11.221133
DOB_yr	-17.997389	zfygpa	9.450359
sex	-9.409471	ugpa	-5.384145
family_income	-4.030357	cluster_tier	-5.37509
lsat	-3.86481	weighted_lsat_ugpa	4.671315
ugpa	3.172791	lsat	-4.408225
weighted_lsat_ugpa	-3.118103	isPartTime	3.522834
isPartTime	3.034175	race	-1.937712
cluster_tier	-1.777019	**family_income**	0.854036

the reliance of classifiers on sensitive features regardless of the choice of explanation method, we made several experiments using SHAP explanations. Due to

Table 7. Global explanation of AdaBoost on Default dataset.

Original (SHAP)		Ensemble (SHAP)	
Feature	**Contrib.**	**Feature**	**Contrib.**
PAY_0	-0.970901	PAY_0	-0.711716
PAY_AMT2	-0.533411	PAY_AMT1	-0.482153
EDUCATION	0.40984	PAY_AMT3	-0.347062
PAY_AMT3	-0.388931	EDUCATION	0.290634
PAY_5	-0.183335	AGE	0.229882
PAY_6	-0.138245	PAY_6	-0.188315
MARRIAGE	-0.056857	PAY_3	-0.122395
PAY_2	-0.048885	PAY_5	-0.103442
PAY_3	-0.028558	PAY_2	-0.08513
SEX	-0.001967	PAY_AMT2	0.07256

lack of space, we do not provide the list of feature contributions for all combinations of datasets and classifiers. Instead, for each dataset, we select the classifier that obtained the highest accuracy.

Tables 3, 4, 5, 6, and 7 present the list of the $k = 10$ most important features with their respective global contribution of both original and ensemble models. In all cases, we notice that FixOut's ensemble classifiers are less reliant on sensitive features. For instance, in the experiment using RF on the German dataset (see Table 3), the global contribution of "statussex" and "telephone" decreased so that both features disappeared of the list of most important features of the ensemble model. Also, we can observe that the sensitive features that still in the top-10 most important features of the ensemble model contributed less to the global prediction compared to the original model. For instance, in the experiment using AdaBoost on the Adult dataset (see Table 4), the absolute value of "MaritalStatus" decreased from -2.5702288 (original model) to -0.156663 (ensemble model). Like in LimeOut we notice the same behavior when we compare both lists (original and ensemble) of top-k important features. However, unlike in LimeOut where Lime$_{\text{Global}}$ deemed fair these 4 classifiers trained on the HMDA [1], FixOut (using SHAP) deems them unfair on the HMDA dataset.

4.4 Fairness Metrics Assessment

In this section, we assess fairness using the standard metrics introduced in Subsection 2.1 in order to have a different perspective of the fairness of FixOut's ensemble models. We compute Demographic Parity (DP), Equal Opportunity (EQ), Equal Accuracy (EA) and Disparate Impact (DI) using IBM AI Fairness 360 Toolkit[17] [3]. We also consider Predictive Equality (PE) [1] to measure the false positive differences between privileged and unprivileged groups. The metrics DP, EO, EA, and PE give values in the interval [-1,1] where 0 indicates a perfect fair model, while the optimal value for DI is 1.

Fairness metrics are depicted in Figures 3 and 4. In this analysis, we compare the original and ensemble models based on fairness metrics for each combination of classifier and sensitive feature. Red points indicate the values for FixOut ensemble models while blue points indicate values for original models. The dashed

[17] https://github.com/Trusted-AI/AIF360

line is the reference for a fair model (optimal value). Results for the German dataset are depicted in Figure 3. Like in LimeOut experiments, FixOut produces ensemble models that are fairer according to metrics DP, EQ and DI, since red points are closer to zero compared to blue points (pre-trained model). However, unlike in LimeOut's case, we can notice that ensemble models behaved almost in the opposite way of original models according to PE metric. In the case of EA metric, ensemble models keep the same fairness level, except for Random Forest on "foreign worker" attribute, while the same combination (classifier,sensitive feature) shows fairness improvement in DP and DI metrics. Figure 4 shows the results on fairness metrics for the Adult dataset. In this dataset, FixOut ensemble models keep values of all metrics in almost scenarios. We only see a deterioration of fairness when we compute EQ for Logistic Regression focuses on marital status. This behaviour means that FixOut at least maintain the value of fairness metrics when it reduces the dependence on sensitive features, but it cannot ensure fairness metrics closer to 0.

These results indicate that, in general, FixOut consistently improves, or at least maintains, the fairness of the ensemble classifier. We also used an aggregation rule that takes into account the global contributions of sensitive features in each classifier. This resulted in a slight improvement of these metrics (when compared to those obtained by LimeOut in which were used simple aggregation). This fact seems to indicate that learning the aggregation rule should further improve these fairness metrics for FixOut's ensemble classifiers.

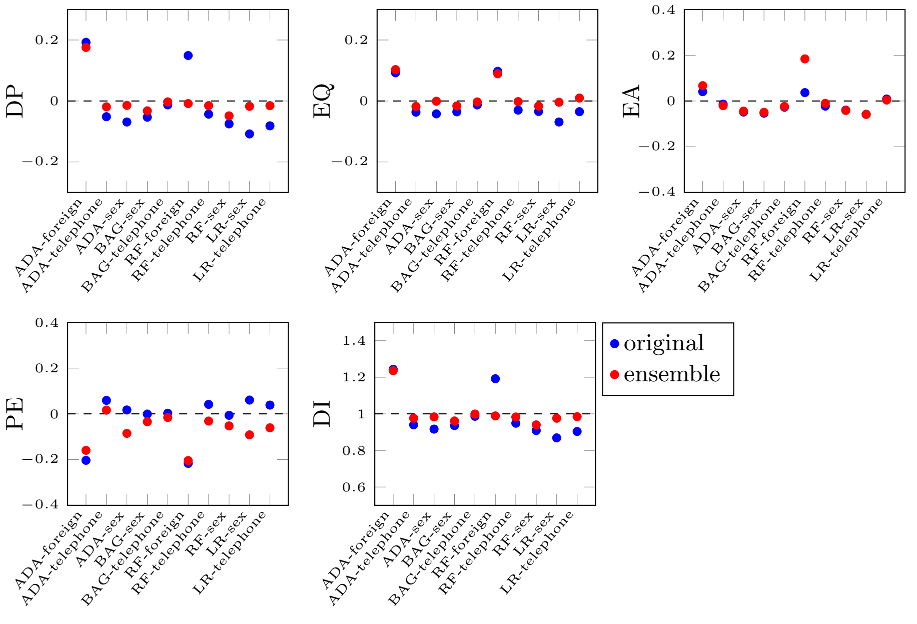

Fig. 3. Fairness metrics for German Credit Score Dataset

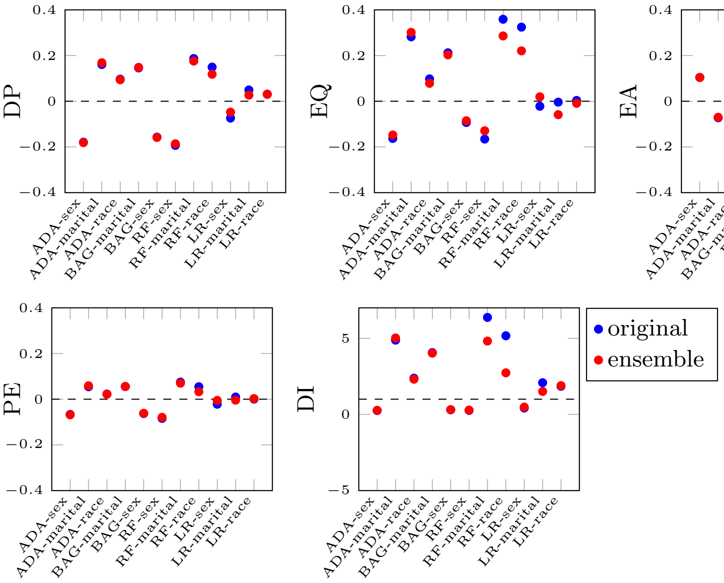

Fig. 4. Fairness metrics for Adult Dataset.

5 Conclusion

We have proposed FixOut a human-centered and model-agnostic framework to make ML models fairer. FixOut was proposed to address and tackle process fairness: it first assesses the dependence of given pre-trained ML model on sensitive features by global explanations in the form of feature contributions to the classifier's outcomes. If the ML model's outcomes are shown to rely on sensitive features, FixOut employs feature dropout followed by an ensemble approach to produce a new model.

In addition to process fairness, we also analysed FixOut empirically on different pre-trained ML models and using several well known fairness metrics. The empirical study performed on five real datasets showed that FixOut produces ensemble classifiers that are less reliant on sensitive features without compromising accuracy. Moreover, it also shows consistent improvements with respect to widely used fairness metrics.

Nonetheless, there is still room for several improvements in FixOut's workflow, in particular: (1) to determine the suitable number k of most important features for a given domain, (2) to learn the aggregation rule on the fly, and (3) to automate the detection of sensitive features. These are some of the topics of current on-going work.

References

1. Alves, G., et al.: Making ML models fairer through explanations: the case of Lime-Out. In: AIST'20
2. Angwin, J., et al.: Machine bias: There's software used across the country to predict future criminals. And it's biased against blacks. ProPublica **23** (2016)
3. Bellamy, R.K.E., et al.: AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. vol. abs/1810.01943 (2018)
4. Bhargava, V., et al.: LimeOut: An Ensemble Approach To Improve Process Fairness. In: ECML PKDD Int. Workshop XKDD (2020)
5. Dimanov, B., et al.: You shouldn't trust me: Learning models which conceal unfairness from multiple explanation methods. In: ECAI'20. pp. 2473–2480
6. Garreau, D., von Luxburg, U.: Explaining the explainer: A first theoretical analysis of LIME. vol. abs/2001.03447 (2020)
7. Grgić-Hlača, N., et al.: Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In: AAAI'18. pp. 51–60
8. Grgic-Hlaca, N., et al.: The case for process fairness in learning: Feature selection for fair decision making. In: NIPS Symposium on Machine Learning and the Law. vol. 1, p. 2 (2016)
9. Guegan, D., et al.: Credit risk analysis using machine and deep learning models. vol. 6, p. 38 pages (2018)
10. Hardt, M., et al.: Equality of opportunity in supervised learning. In: NIPS'16
11. Lundberg, S., et al.: Git repository of SHAP. `https://github.com/slundberg/shap` (2020), [Online; accessed 26-November-2020]
12. Lundberg, S.M., Lee, S.: A unified approach to interpreting model predictions. In: NIPS'17. pp. 4765–4774
13. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Advances in neural information processing systems. pp. 4765–4774 (2017)
14. Molnar, C.: Interpretable Machine Learning. Lulu. com (2020)
15. Ribeiro, M.T., et al.: "Why Should I Trust You?": Explaining the predictions of any classifier. In: SIGKDD'16. pp. 1135–1144
16. Ribeiro, M.T., et al.: Anchors: High-precision model-agnostic explanations. In: AAAI. vol. 18, pp. 1527–1535 (2018)
17. Shapley, L.S.: A value for n-person games. Contributions to the Theory of Games **2**(28), 307–317 (1953)
18. Shrikumar, A., et al.: Learning important features through propagating activation differences. arXiv preprint arXiv:1704.02685 (2017)
19. Slack, D., et al.: Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In: AIES'20. pp. 180–186
20. Speicher, T., et al.: A unified approach to quantifying algorithmic unfairness: Measuring individual & group unfairness via inequality indices. In: SIGKDD'18. pp. 2239–2248
21. Zafar, M.B., et al.: Fairness constraints: Mechanisms for fair classification. In: Artificial Intelligence and Statistics. pp. 962–970. PMLR (2017)