# Causal Graph Fuzzing for Fair ML Software Development

Verya Monjezi
vmonjezi@miners.utep.edu
University of Texas at El Paso
El Paso, TX, USA

Ashish Kumar
azk640@psu.edu
Penn State
State College, PA, USA

Gang Tan
gtan@psu.edu
Penn State
State College, PA, USA

Ashutosh Trivedi
Ashutosh.Trivedi@colorado.edu
University of Colorado Boulder
Boulder, CO, USA

Saeid Tizpaz-Niari
saeid@utep.edu
University of Texas at El Paso
El Paso, TX, USA

## ABSTRACT

Machine learning (ML) is increasingly used in high-stakes areas like autonomous driving, finance, and criminal justice. However, it often unintentionally perpetuates biases against marginalized groups. To address this, the software engineering community has developed fairness testing and debugging methods, establishing best practices for fair ML software. These practices focus on training model design, including the selection of sensitive and non-sensitive attributes and hyperparameter configuration. However, the application of these practices across different socio-economic and cultural contexts is challenging, as societal constraints vary.

Our study proposes a search-based software engineering approach to evaluate the robustness of these fairness practices. We formulate these practices as the first-order logic properties and search for two neighborhood datasets where the practice satisfies in one dataset, but fail in the other one. Our key observation is that these practices should be general and robust to various uncertainty such as noise, faulty labeling, and demographic shifts. To generate datasets, we sift to the causal graph representations of datasets and apply perturbations over the causal graphs to generate neighborhood datasets. In this short paper, we show our methodology using an example of predicting risks in the car insurance application.

**ACM Reference Format:**
Verya Monjezi, Ashish Kumar, Gang Tan, Ashutosh Trivedi, and Saeid Tizpaz-Niari. 2024. Causal Graph Fuzzing for Fair ML Software Development. In *2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion '24), April 14–20, 2024, Lisbon, Portugal.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3639478.3643530

## 1 INTRODUCTION

The software engineering community has proposed various practices in developing fair ML software [1]. Those practices cover different aspects of developments such as pre-processing, algorithm design, and fine-tuning. For example, they found that enlarging the feature space can improve fairness or selecting a particular hyperparameters may degrade fairness [2–4].

Are these practices locally robust in varying settings? This paper proposes a novel approach to evaluate the robustness of these fairness practices by focusing on their validity under some local perturbations. It focuses on the robustness of software development strategies on entire datasets, not just around individual samples. Since inferring neighborhood datasets is challenging, a key aspect is defining the similarity between generative models of datasets, using weighted causal models inferred from the data. We propose using search-based software engineering approaches [5, 6], leveraging graph mutation algorithms to perturb causal graphs and generate neighboring datasets. The perturbations aim to characterize factors such as noise in sampling, faulty labeling, and distribution shifts. The approach helps us understand conditions under which empirical findings about fairness may or may not hold.

## 2 OVERVIEW

In this section, we discuss our investigation on the impact of dropping features and selecting hyperparameters on the fairness of machine learning models using an insurance dataset.

**Incorporating Causal Graph.** In this section, the paper examines how feature selection and hyperparameter configuration during training impact the fairness of a model. We adapting a directed acyclic graph (DAG) from existing literature to represent the causal graph of car insurance risk [7]. To quantify the strength of these causal relationships, we use Bayesian methods via the `STAN` probabilistic programming language to infer the weight of each edge in the DAG. This inference process considers both features (nodes) and the coefficients of linear models connecting the nodes in the DAG. Following this, the study leverages the probabilistic program to introduce perturbations in the graph, effectively simulating slight shifts in the dataset distribution. This step is critical for generating data samples that reflect changes on the dataset, allowing for the examination of the robustness of the fairness properties under different scenarios.

**Understanding Impacts of Excluding Sensitive Attribute during Training.** We generate data samples from the probabilistic program of a base causal graph. Then, we train a logistic regression model using all features, including the sensitive attribute (race), and calculate the Equal Opportunity Difference (EOD. Next, we perform the same training with logistic regression without the sensitive attribute. Figure 1 (top) shows EOD bias differences


*ICSE-Companion '24, April 14–20, 2024, Lisbon, Portugal*

ACM ISBN 979-8-4007-0502-1/24/04.
https://doi.org/10.1145/3639478.3643530

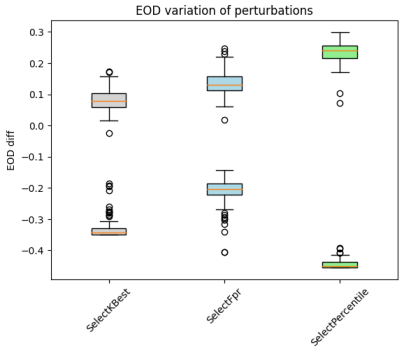


**Figure 2: Result of causal perturbation and feature selection on EOD.**

between training with the full feature set and without the sensitive attribute race, while F1 and accuracy differences are below 0.02. The results show that dropping the sensitive attribute aggravates the EOD bias by 0.13, consistent with prior findings. We hypothesize that a (slightly) different causal graph could challenge the negative effects of dropping sensitive attributes on fairness. To test this hypothesis, we employ a causal search-based algorithm to perturb the causal graph iteratively to identify a graph similar to the base graph that contradicts the findings of aggravating bias by dropping sensitive attributes. we repeat the same training as before, but this time we use the perturbed causal graph. We generate i.i.d samples as training data and perform two experiments with the full feature set and without the sensitive attribute race. Figure 1 (bottom) presents the results of the EOD change in the model trained by dropping the sensitive attribute race compared to the model trained with all features. Remarkably, the results show that the EOD of the model trained by dropping race decreased by 0.1 compared to the EOD of the model trained with all features (within 0.02 of F1 and accuracy scores). This finding reveals the importance of causal graphs for fairness when dropping sensitive attributes during training.

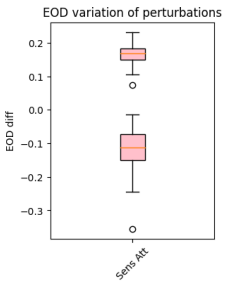


**Figure 1: Dropping sensitive attributes.**

**Understanding Impacts of Feature Selection on Fairness.** For the base causal graph, we train the logistic regression models by excluding different sets of non-sensitive features. Figure 2 shows the results where dropping some non-sensitive features mostly led to an increase in EOD bias which aligns with prior research. However, we repeat the same experiment of dropping non-sensitive features on this modified causal graph. The results suggest that feature importance is significantly different than the pattern in the base graph. Dropping different features consistently decreased the EOD for all cases. We also investigate standard feature selection operations in the ML pipeline. In particular, we consider three prevalent feature selection techniques: `SelectKBest` , `SelectFpr` , and `SelectPercentile`. Previous research [8] suggests that applying `SelectKBest` and `SelectPercentile` increased unfairness whereas `SelectFpr` did not impact fairness. However, our analysis showed that such empirical observations might not be locally robust. As before, we use the base causal graph, train logistic regression, and measure EOD after applying these operators. Then, we applied our mutation algorithm over the causal graph and identified perturbations that negate the empirical observations. The results of this experiment showcase the percentiles of EOD variations observed varying at most one edge of the base causal graph. The results indicate that each of these operators can increase or decrease fairness (based on EOD bias) depending on the causal relationships between variables. These findings suggest that the relationship between feature exclusion and fairness might not be locally robust and requires causal analysis.

**Understanding Impacts of Hyperparameter Selection on Fairness.** We leveraged a search-based algorithm [3] to explore the space of HP configurations. We ran the search for a fixed duration to explore a diverse range of HP configurations. Given a set of relevant HPs and their fairness characteristics, we leverage the Shapley Additive Explanations (SHAP) algorithm. The SHAP outcome for the base graph illustrates the importance of four HPs: tol, solver, fit_intercept, and intercept_scaling where the perturbed graph includes the following HPs: fit_intercept, penalty, solver, and warm_start.

## 3 CONCLUSION

Our study challenged the universality of these best practices, positing that the robustness of these practices across different settings is crucial to their validity. Over an example of car insurance, we show that some well-known practices are not valid under different setting. In the future work, we plan to investigate the validity of practices over a large number of datasets and their graphs.

## REFERENCES

[1] N. Yu, G. Tan, and S. Tizpaz-Niari, "Fairlay-ml: Intuitive remedies for unfairness in data-driven social-critical algorithms," 2023. [Online]. Available: https://arxiv.org/abs/2307.05029

[2] J. M. Zhang and M. Harman, ""ignorance and prejudice" in software fairness," in *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 2021, pp. 1436–1447.

[3] S. Tizpaz-Niari, A. Kumar, G. Tan, and A. Trivedi, "Fairness-aware configuration of machine learning libraries," in *Proceedings of the 44th International Conference on Software Engineering*, ser. ICSE '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 909–920. [Online]. Available: https://doi.org/10.1145/3510003.3510202

[4] T. Nguyen, S. Tizpaz-Niari, and V. Kreinovich, "How to make machine learning financial recommendations more fair: Theoretical explanation," 2023. [Online]. Available: https://scholarworks.utep.edu/cgi/viewcontent.cgi?article=2829&context=cs_techrep

[5] Y. Noller and S. Tizpaz-Niari, "Qfuzz: Quantitative fuzzing for side channels," in *Proceedings of the 30th ACM SIGSOFT International Symposium on Software Testing and Analysis*, 2021, pp. 257–269.

[6] V. Monjezi, A. Trivedi, G. Tan, and S. Tizpaz-Niari, "Information-theoretic testing and debugging of fairness defects in deep neural networks," in *Proceedings of the 45th International Conference on Software Engineering*, ser. ICSE '23. IEEE Press, 2023, p. 1571–1582. [Online]. Available: https://doi.org/10.1109/ICSE48619.2023.00136

[7] M. Kusner, J. Loftus, C. Russell, and R. Silva, "Counterfactual fairness," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 4069–4079.

[8] S. Biswas and H. Rajan, "Fair preprocessing: Towards understanding compositional fairness of data transformers in machine learning pipeline," ser. ESEC/FSE 2021. New York, NY, USA: Association for Computing Machinery, 2021, p. 981–993. [Online]. Available: https://doi.org/10.1145/3468264.3468536