FOOL SHAP WITH STEALTHILY BIASED SAMPLING.

Anonymous authors
Paper under double-blind review

ABSTRACT

SHAP explanations aim at identifying which features contribute the most to the difference in model prediction at a specific input versus a background distribution. Recent studies have shown that they can be manipulated by malicious adversaries to produce arbitrary desired explanations. However, existing attacks focus solely on altering the black-box model itself. In this paper, we propose a complementary family of attacks that leave the model intact and manipulate SHAP explanations using stealthily biased sampling of the data points used to approximate expectations w.r.t the background distribution. In the context of fairness audit, we show that our attack can reduce the importance of a sensitive feature when explaining the difference in outcomes between groups while remaining undetected. More precisely, experiments performed on real-world datasets showed that our attack could yield up to an 90% relative decrease in amplitude of the sensitive feature attribution while remaining undetected. These results highlight the manipulability of SHAP explanations and encourage auditors to treat them with skepticism.

1 INTRODUCTION

As Machine Learning (ML) gets more and more ubiquitous in high-stake decision contexts (e.g., healthcare, finance, and justice), concerns about its potential to lead to discriminatory models are becoming prominent. The use of auditing toolkits (Adebayo et al., 2016; Saleiro et al., 2018; Bellamy et al., 2018) is getting popular to circumvent the use of unfair models. However, although auditing toolkits can help model designers in promoting fairness, they can also be manipulated to mislead both the end-users and external auditors. For instance, a recent study of Fukuchi et al. (2020) has shown that malicious model designers can produce a benchmark dataset as fake "evidence" of the fairness of the model even though the model itself is unfair.

Another approach to assess the fairness of ML systems is to explain their outcome in a *post hoc* manner (Guidotti et al., 2018). For instance, `SHAP` (Lundberg & Lee, 2017) has risen in popularity as a means to extract model-agnostic local feature attributions. Feature attributions are meant to convey how much the model relies on certain features to make a decision at some specific input. The use of feature attributions for fairness auditing is desirable for cases where the interest is on the direct impact of the sensitive attributes on the output of the model. One such situation is in the context of causal fairness (Chikahara et al., 2021). In some practical cases, the outputs cannot be independent from the sensitive attribute unless we sacrifice much of prediction accuracy. For example, any decisions based on physical strength are statistically correlated to gender due to biological nature. The problem in such a situation is not the statistical bias (such as demographic parity), but whether the decision is based on the physical strength or gender, *i.e.* the attributions of each feature.

The focus of this study is on manipulating the feature attributions so that the dependence on sensitive attributions is hidden and the audits are misled as if the model is fair even if it is not the case. Recently, several studies reported that such a manipulation is possible, *e.g.* by modifying the black-box model to be explained (Slack et al., 2020; Begley et al., 2020; Dimanov et al., 2020), by manipulating the computation algorithms of feature attributions (Aïvodji et al., 2019), and by poisoning the data distribution (Baniecki et al., 2021; Baniecki & Biecek, 2022). With these findings in mind, the current possible advice to the auditors is not to rely solely on the reported feature attributions for fairness auditing. A question then arises about what "evidence" we can expect in addition to the feature attributions, and whether they can be valid "evidence" of fairness.

In this study, we show that we can craft fake "evidence" of fairness for `SHAP` explanations, which provides the first negative answer to the last question. In particular, we show that we can produce not

only manipulated feature attributions but also a benchmark dataset as the fake "evidence" of fairness. The benchmark dataset ensures the external auditors reproduce the reported feature attributions using the existing `SHAP` library. In our study, we leverage the idea of stealthily biased sampling introduced by Fukuchi et al. (2020) to cherry-pick which data points to be included in the benchmark. Moreover, the use of stealthily biased sampling allows us to keep the manipulation undetected by making the distribution of the benchmark sufficiently close to the true data distribution. Figure 1 illustrates the impact of our attack in an explanation scenario with the Adult Income dataset.

Our contributions can be summarized as follows:

- Theoretically, we formalize a notion of foreground distribution that can be used to extend Local Shapley Values (LSV) to Global Shapley Values (GSV), which can be used to decompose fairness metrics among the features (Section 2.2). Moreover, we formalize the task of manipulating the GSV as a Minimum Cost Flow (MCF) problem (Section 4).
- Experimentally (Section 5), we illustrate the impact of the proposed manipulation attack on a synthetic dataset and four popular datasets, namely Adult Income, COMPAS, Marketing, and Communities. We observed that the proposed attack can reduce the importance of a sensitive feature while keeping the data manipulation undetected by the audit.

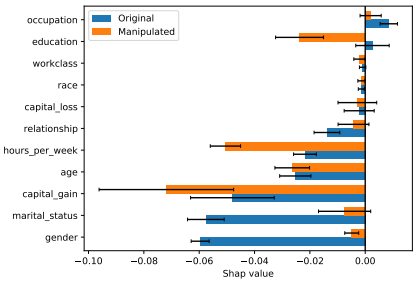


Figure 1: Example of our attack on the Adult Income dataset. After the attack, the feature `gender` moved from the most negative attribution to the 6th, hence hiding some of the explicit model bias.

Our results indicate that `SHAP` explanations are not robust and can be manipulated when it comes to explaining the difference in outcomes between groups. Even worse, our results confirm we can craft a benchmark dataset so that the manipulated feature attributions are reproducible by external audits. Henceforth, we alert auditors to treat post-hoc explanations methods with skepticism even if it is accompanied by some additional evidence.

2 Shapley Values

2.1 Local Shapley Values

Shapley values are omnipresent in post-hoc explainability because of their fundamental mathematical properties (Shapley, 1953) and their implementation in the popular `SHAP` Python library (Lundberg & Lee, 2017). `SHAP` provides local explanations in the form of feature attributions i.e. given an input of interest $\boldsymbol{x}$, `SHAP` returns a score $\phi_i \in \mathbb{R}$ for each feature $i = 1, 2, \ldots, d$. These scores are meant to convey how much the model f relies on feature i to make its decision $f(\boldsymbol{x})$. Shapley values have a long background in coalitional game theory, where multiple players collaborate toward a common outcome. In the context of explaining model decisions, the players are the input features and the common outcome is the model output $f(\boldsymbol{x})$. In coalitional games, players (features) are either present or absent. Since one cannot physically remove an input feature once the model has already been fitted, `SHAP` removes features by replacing them with a baseline value $\boldsymbol{z}$. This leads to the *Local Shapley Value* (LSV) $\phi_i(f, \boldsymbol{x}, \boldsymbol{z})$ which respect the so-called efficiency axiom (Lundberg & Lee, 2017)

$$\sum_{i=1}^{d} \phi_i(f, \boldsymbol{x}, \boldsymbol{z}) = f(\boldsymbol{x}) - f(\boldsymbol{z}). \tag{1}$$

Simply put, the difference between the model prediction at $\boldsymbol{x}$ and the baseline $\boldsymbol{z}$ is shared among the different features. Additional details on the computation of LSV are presented in Appendix B.1.

2.2 Global Shapley Values

LSV are local in the sense that they explain the prediction at a specific $\boldsymbol{x}$ and rely on a single baseline input $\boldsymbol{z}$. Since model auditing requires a more global analysis of model behavior, we must understand the predictions at multiple inputs $\boldsymbol{x} \sim \mathcal{F}$ sampled from a distribution $\mathcal{F}$ called the

foreground. Moreover, because the choice of baseline is somewhat ambiguous, the baselines are sampled $\boldsymbol{z} \sim \mathcal{B}$ from a distribution $\mathcal{B}$ colloquially referred to as the *background*. Taking inspiration from Begley et al. (2020), we can compute *Global Shapley Values* (GSV) by averaging LSV over both foreground and background distributions.

Definition 2.1.

$$\Phi_i(f, \mathcal{F}, \mathcal{B}) := \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim\mathcal{F}\\ \boldsymbol{z}\sim\mathcal{B}}}\big[\phi_i(f, \boldsymbol{x}, \boldsymbol{z})\big], \quad i = 1, 2, \ldots, d. \tag{2}$$

Proposition 2.1. *The GSV have the following property*

$$\sum_{i=1}^{d} \Phi_i(f, \mathcal{F}, \mathcal{B}) = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{F}}[f(\boldsymbol{x})] - \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{B}}[f(\boldsymbol{x})]. \tag{3}$$

2.3 Monte-Carlo Estimates

In practice, computing expectations w.r.t the whole background and foreground distributions may be prohibitive and hence Monte-Carlo estimates are used. For instance, when a dataset is used to represent a background distribution, many explainers in the `SHAP` library such as the `ExactExplainer` and `TreeExplainer` will subsample this dataset[1] by selecting 100 instances uniformly at random when the size of the dataset exceeds 100. More formally, let

$$\mathcal{C}(S, \boldsymbol{\omega}) := \sum_{\boldsymbol{x}^{(j)}\in S} \omega_j \delta(\boldsymbol{x}^{(j)}) \tag{4}$$

represent a categorical distribution over a finite set of input examples S, where $\delta(\cdot)$ is the dirac probability measure, $w_j \geqslant 0\ \forall j$, and $\sum_j \omega_j = 1$. Estimating expectations with Monte-Carlo amounts to sampling M instances

$$S_0 \sim \mathcal{F}^M \qquad S_1 \sim \mathcal{B}^M, \tag{5}$$

and computing the plug-in estimate

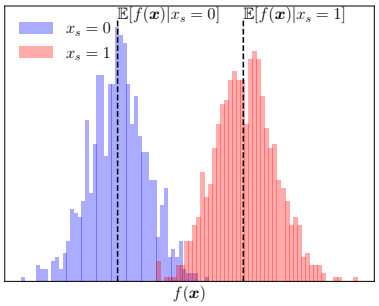
$$\begin{aligned}\widehat{\boldsymbol{\Phi}}(f, S_0, S_1) &:= \boldsymbol{\Phi}(f,\, \mathcal{C}(S_0, \mathbf{1}/M),\, \mathcal{C}(S_1, \mathbf{1}/M)) \\ &= \frac{1}{M^2} \sum_{\boldsymbol{x}^{(i)}\in S_0} \sum_{\boldsymbol{z}^{(j)}\in S_1} \boldsymbol{\phi}(f, \boldsymbol{x}^{(i)}, \boldsymbol{z}^{(j)}).\end{aligned} \tag{6}$$

When a set of samples is a singleton (*e.g.* $S_1 = \{\boldsymbol{z}^{(j)}\}$), we shall use the convention $\widehat{\boldsymbol{\Phi}}(f, S_0, \{\boldsymbol{z}^{(j)}\}) \equiv \widehat{\boldsymbol{\Phi}}(f, S_0, \boldsymbol{z}^{(j)})$ to improve readability. In Appendix B.2, $\widehat{\boldsymbol{\Phi}}(f, S_0, S_1)$ is shown to be a consistent and asymptotically normal estimate of $\boldsymbol{\Phi}(f, \mathcal{F}, \mathcal{B})$ meaning that one can compute approximate confidence intervals around $\widehat{\boldsymbol{\Phi}}$ to capture $\boldsymbol{\Phi}$ with high probability. In practice, the estimates $\widehat{\boldsymbol{\Phi}}$ are employed as the model explanation which we see as a vulnerability. As discussed in Section 4, the Monte-Carlo estimation is the key ingredient that allow us to manipulate the GSV in favor of a dishonest entity.

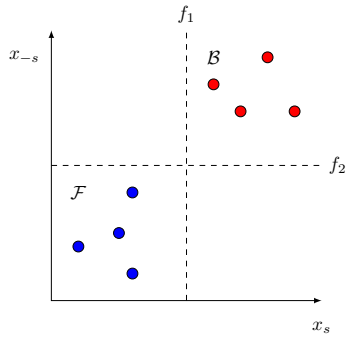
3 Audit Scenario

This section introduces an audit scenario to which the proposed attack of `SHAP` can apply. This scenario involves two parties: a company and an audit. The company has a dataset $D = \{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$ with $\boldsymbol{x}^{(i)} \in \mathbb{R}^d$ and $y^{(i)} \in \{0, 1\}$ that contains N input-target tuples and also has a model $f : \mathcal{X} \to [0, 1]$ that is meant to be deployed in society. The binary feature with index s (*i.e* $x_s \in \{0, 1\}$) represents a sensitive feature with respect to which the model should not explicitly discriminate. Both the data D and the model f are highly private so the company is very careful when providing information about them to the audit. Hence, f is a black box from the point of view of the audit. At first, the audit asks the company for the necessary data to compute fairness metrics *e.g.* the Demographic Parity (Dwork et al., 2012), the Predictive Equality (Corbett-Davies et al., 2017), or the Equal Opportunity (Hardt et al., 2016). Note that our attack would apply as long as the

[1] `https://github.com/slundberg/shap/blob/0662f4e9e6be38e65812007990489 9cccda59ff8/shap/maskers/_tabular.py#L54-L55`



(a) The data initially provided by the company to the audit is $f(D_0)$ and $f(D_1)$ *i.e* the model predictions for all instances in the private dataset for different values of x_s. This dataset can later be used by the audit to assess whether or not the subsets S_0', S_1' provided by the company where cherry-picked.



(b) Two models f_1 and f_2 (decision boundaries in dashed lines) with perfect accuracy exhibit a disparity in outcomes w.r.t groups with $x_s < 0$ and $x_s > 0$. Here, $\Phi_s(f_1, \mathcal{F}, \mathcal{B}) = -1$ while $\Phi_s(f_2, \mathcal{F}, \mathcal{B}) = 0$. Hence, f_2 is **indirectly** unfair toward x_s because of correlations in the data.

Figure 2: Illustrations of the audit scenario.

fairness metric is a difference in model expectations over subgroups. For simplicity, the audit decides to compute the Demographic Parity

$$\mathbb{E}[f(\boldsymbol{x})|x_s = 0] - \mathbb{E}[f(\boldsymbol{x})|x_s = 1], \tag{7}$$

and henceforth demands access to the model outputs for all inputs with different values of the sensitive feature *i.e* $f(D_0)$ and $f(D_1)$, where $D_0 = \{\boldsymbol{x}^{(i)} : x_s^{(i)} = 0\}$ and $D_1 = \{\boldsymbol{x}^{(i)} : x_s^{(i)} = 1\}$ are subsets of the input data of sizes N_0 and N_1 respectively. Doing so does not force the company to share values of features other than x_s nor does it requires direct access to the inner workings of the proprietary model. Hence this demand respects privacy requirements and the company will accept to share the model outputs across all instances, see Figure 2a. At this point, the audit confirms that the model is indeed biased in favor of $x_s = 1$ and puts in question the ability of the company to deploy such a model. Now, the company argues that, although the model exhibits a disparity in outcomes, it does not mean that the model explicitly uses the feature x_s to make its decision. If such is the case, then the disparity could be explained by other features statistically associated with x_s. Some of these other features may be acceptable grounds for decisions. To verify such a claim, the audit decides to employ post-hoc techniques to explain the disparity. Since the model is a black-box, the audits shall compute the GSVs. The foreground $\mathcal{F}$ and background $\mathcal{B}$ are chosen to be the data distributions conditioned on $x_s = 0$ and $x_s = 1$ respectively

$$\mathcal{F} := \mathcal{C}(D_0, \mathbf{1}/N_0) \qquad \mathcal{B} := \mathcal{C}(D_1, \mathbf{1}/N_1). \tag{8}$$

According to Equation 3, the resulting GSVs will sum up of the demographic parity (cf. Equation 7). If the sensitive feature has a large negative GSV Φ_s, then this would mean that the model is **explicitly** relying on x_s to make its decisions and the company would be forbidden from deploying the model. If the GSV has a small amplitude, however, the company could still argue in favour of deploying the model in spite of having disparate outcomes. Indeed, the difference in outcomes by the model could be attributed to other more acceptable features. See Figure 2b for a toy example illustrating this reasoning.

To compute the GSV, the audit demands the two datasets of inputs D_0 and D_1, as well as the ability to query the black box f at arbitrary points. Because of privacy concerns on sharing values of $\boldsymbol{x}$ across the whole dataset, and because GSV must be estimated with Monte-Carlo, both parties agree that the company shall only provide subsets $S_0 \subset D_0$ and $S_1 \subset D_1$ of size M to the audit so they can compute a Monte-Carlo estimate $\widehat{\boldsymbol{\Phi}}(f, S_0, S_1)$. The company first estimate GSV on their own by choosing S_0, S_1 uniformly at random from $\mathcal{F}$ and $\mathcal{B}$ (cf. Equation 5) and observe that $\widehat{\Phi}_s$ indeed has a large negative value. They realize they must carefully select which data points will be sent, otherwise, the audit may observe the explicit bias toward $x_s = 1$ and the model will not be deployed. Moreover, the company understands that the audit currently has access to the data $f(D_0)$ and $f(D_1)$ representing the model predictions on the whole dataset (see Figure 2a). Therefore, if the company

does not share subsets S_0, S_1 that where chosen uniformly at random from D_0, D_1, it is possible for the audit to detect this fraud by doing a statistical test comparing $f(S_0)$ to $f(D_0)$ and $f(S_1)$ to $f(D_1)$. The company needs a method to select **misleading subsets** S'_0, S'_1 in a manner that manipulates the GSV in their favour, while remaining undetected by the audit. Such a method is the subject of the next section.

4 FOOL SHAP WITH STEALTHILY BIASED SAMPLING

4.1 MANIPULATION

To fool the audit, the company can decide to indeed sample S'_0 uniformly at random $S'_0 \sim \mathcal{F}^M$. Then, given this choice of foreground sub-sample, they can manipulate the background distribution to cherry-pick M instances. Formally, the company must compute a non-uniform background distribution $\mathcal{B}'_{\boldsymbol{\omega}} := \mathcal{C}(D_1, \boldsymbol{\omega})$ with $\boldsymbol{\omega} \neq \mathbf{1}/N_1$ from which to sample M points $S'_1 \sim \mathcal{B}'^M_{\boldsymbol{\omega}}$. To make the model look fairer, the company needs the $\widehat{\Phi}_s$ computed with these cherry-picked points to have a small magnitude.

Proposition 4.1. *Let S'_0 be **fixed**, and let $\xrightarrow{p}$ represent convergence in probability as the size M of the set $S'_1 \sim \mathcal{B}'^M_{\boldsymbol{\omega}}$ increases, we have*

$$\widehat{\Phi}_s(f, S'_0, S'_1) \xrightarrow{p} \sum_{\boldsymbol{z}^{(j)} \in D_1} \omega_j \, \widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)}). \tag{9}$$

We note that the coefficients $\widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)})$ in Equation 9 are tractable and can be computed and stored by the company. We discuss in more detail how to compute them in Appendix B.3. Remember that to manipulate the GSV, the company must tune the weights $\boldsymbol{\omega}$ such that the explanation $\widehat{\Phi}_s$ is manipulated while ensuring that the non-uniform distribution $\mathcal{B}'_{\boldsymbol{\omega}}$ remains similar to the original $\mathcal{B}$. Otherwise, the fraud could be detected by the audit. In this work, the notion of similarity between distributions will be captured by the Wasserstein distance in output space.

Definition 4.1 (Wassertein Distance)**.** *Any probability measure π over $D_1 \times D_1$ is called a coupling measure between $\mathcal{B}$ and $\mathcal{B}'_{\boldsymbol{\omega}}$, denoted $\pi \in \Delta(\mathcal{B}, \mathcal{B}'_{\boldsymbol{\omega}})$, if $1/N_1 = \sum_j \pi_{ij}$ and $\omega_j = \sum_i \pi_{ij}$. The Wassertein distance between $\mathcal{B}$ and $\mathcal{B}'_{\boldsymbol{\omega}}$ mapped to the output-space is defined as*

$$\mathcal{W}(\mathcal{B}, \mathcal{B}'_{\boldsymbol{\omega}}) = \min_{\pi \in \Delta(\mathcal{B}, \mathcal{B}'_{\boldsymbol{\omega}})} \sum_{i,j} |f(\boldsymbol{z}^{(i)}) - f(\boldsymbol{z}^{(j)})| \pi_{ij}, \tag{10}$$

representing the cost of the optimal transport plan that distributes the probability mass from one distribution to the other.

We propose the Algorithm 1 to compute the non-uniform weights $\boldsymbol{\omega}$ by minimizing the magnitude of the GSV while maintaining a small Wasserstein distance. The trade-off between attribution manipulation and proximity to the data is tuned via a hyper-parameter $\lambda > 0$. We show in the Appendix A.2 that the optimization problem at line 5 of Algorithm 1 can be reformulated as a Minimum Cost Flow (MCF) and hence can be solved in polynomial time (more precisely $\widetilde{\mathcal{O}}(N_1^{2.5})$ as in Fukuchi et al. (2020)).

Algorithm 1 Compute non-uniform weights

1: **procedure** COMPUTE_WEIGHTS($D_1, \{\widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)})\}_j, \lambda$)
2: $\quad \beta \;:= \text{sign}[\sum_{\boldsymbol{z}^{(j)} \in D_1} \widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)})]$
3: $\quad \mathcal{B} \;:= \mathcal{C}(D_1, \mathbf{1}/N_1)$ ▷ Unmanipulated background
4: $\quad \mathcal{B}'_{\boldsymbol{\omega}} := \mathcal{C}(D_1, \boldsymbol{\omega})$ ▷ Manipulated background as a function of $\boldsymbol{\omega}$
5: $\quad \boldsymbol{\omega} = \arg\min_{\boldsymbol{\omega}} \; \beta \sum_{\boldsymbol{z}^{(j)} \in D_1} \omega_j \, \widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)}) + \lambda \mathcal{W}(\mathcal{B}, \mathcal{B}'_{\boldsymbol{\omega}})$ ▷ Optimization Problem
6: $\quad$ **return** $\boldsymbol{\omega}$;

4.2 DETECTION

We now discuss ways the audit can detect manipulation of the sampling procedure. Recall that the audit has previously been given access to $f(D_1), f(D_0)$ representing the model outputs across all instances in the private dataset. The audit will then be given sub-samples S'_1, S'_0 of D_1, D_0 on which they can compute the output of the model and compare with $f(D_1), f(D_0)$. To assess whether or not the sub-samples provided by the company were sampled uniformly at random, the audit has to conduct statistical tests. The null hypothesis of these tests will be that S'_1, S'_0 were sampled uniformly at random from D_1, D_0. The detection Algorithm 2 with significance α uses both the Kolmogorov-Smirnov and Wald tests with Bonferonni corrections (i.e. the $\alpha/4$ terms in the Algorithm). The Kolmogorov-Smirnov and Wald tests are discussed in more details in appendix C.

Algorithm 2 Detection with significance α

procedure DETECT_FRAUD($f(D_0), f(D_1), f(S'_0), f(S'_1), \alpha, M$)
  for $i = 0, 1$ **do**
    $f(S_i) \sim \mathcal{C}(f(D_i), \mathbf{1}/N_i)^M$ ▷ Subsample without cheating.
    p-value-KS $=$ KS($f(S_i), f(S'_i)$) ▷ KS test comparing $f(S_i)$ and $f(S'_i)$
    p-value-Wald $=$ Wald($f(S'_i), f(D_i)$) ▷ Wald test
    if p-value-KS $< \alpha/4$ **or** p-value-Wald $< \alpha/4$ **then** ▷ Reject the null hypothesis
      return 1
  return 0;

4.3 WHOLE PROCEDURE

The procedure returning the subsets S'_0, S'_1 is presented in Algorithm 3. It conducts a log-space search between λ_{min} and λ_{max} for the λ hyper-parameter (line 6) in order to explore the space of possible attacks. For each value of λ, the attacker runs Algorithm 1 to obtain $\mathcal{B}'_{\boldsymbol{\omega}}$ (line 7-8), then repeatedly samples $S'_1 \sim \mathcal{B}'^M_{\boldsymbol{\omega}}$ (line 11) and attempts to detect the fraud via the detection algorithm (line 12). The attacker will choose $\mathcal{B}'_{\boldsymbol{\omega}}$ that reduces the magnitude of $\widehat{\Phi}_s$ the most while having a detection rate below some threshold τ (line 13). An example of search over λ on a real-world dataset is presented in Figure 3. One of the limitations of the proposed methodology is that we only manipulate a single sensitive attribute. In appendix E.1, we present a possible extension of Algorithm 1 to handle multiple sensitive attributes and present preliminary results of its effectiveness.

Algorithm 3 Fool SHAP

1: **procedure** FOOL_SHAP($f, D_0, D_1, M, \lambda_{\text{min}}, \lambda_{\text{max}}, \tau, \alpha$)
2: $S'_0 \sim \mathcal{C}(D_0, \mathbf{1}/N_0)^M$ ▷ S'_0 is sampled without cheating
3: Compute $\widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)}) \quad \forall \boldsymbol{z}^{(j)} \in D_1$ ▷ cf. Section B.3
4: $\mathcal{B}^\star = \mathcal{C}(D_1, \mathbf{1}/N_1)$
5: $\Phi^\star_s = 1/N_1 \sum_{\boldsymbol{z}^{(j)} \in D_1} \widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)})$ ▷ Initialize the solution
6: **for** $\lambda = \lambda_{\text{min}}, \dots, \lambda_{\text{max}}$ **do**
7: $\boldsymbol{\omega} =$ COMPUTE_WEIGHTS($D_1, \left\{\widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)})\right\}_j, \lambda$)
8: $\mathcal{B}'_{\boldsymbol{\omega}} = \mathcal{C}(D_1, \boldsymbol{\omega})$
9: `Detection_rate` $= 0$
10: **for** rep $= 1, \dots, 100$ **do** ▷ Detect the manipulation
11: $S'_1 \sim \mathcal{B}'^M_{\boldsymbol{\omega}}$
12: `Detection_rate` += DETECT_FRAUD($f(D_0), f(D_1), f(S'_0), f(S'_1), \alpha, M$)
13: **if** $|\sum_{\boldsymbol{z}^{(j)} \in D_1} \omega_j \widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)})| < |\Phi^\star_s|$ **and** `Detection_rate` $< 100\tau$ **then**
14: $\mathcal{B}^\star = \mathcal{B}'_{\boldsymbol{\omega}}$
15: $\Phi^\star_s = \sum_{\boldsymbol{z}^{(j)} \in D_1} \omega_j \widehat{\Phi}_s(f, S'_0, \boldsymbol{z}^{(j)})$ ▷ Update the solution
16: $S'_1 \sim \mathcal{B}^{\star M}$ ▷ Cherry-pick by sampling from the non-uniform background
17: **return** S'_0, S'_1

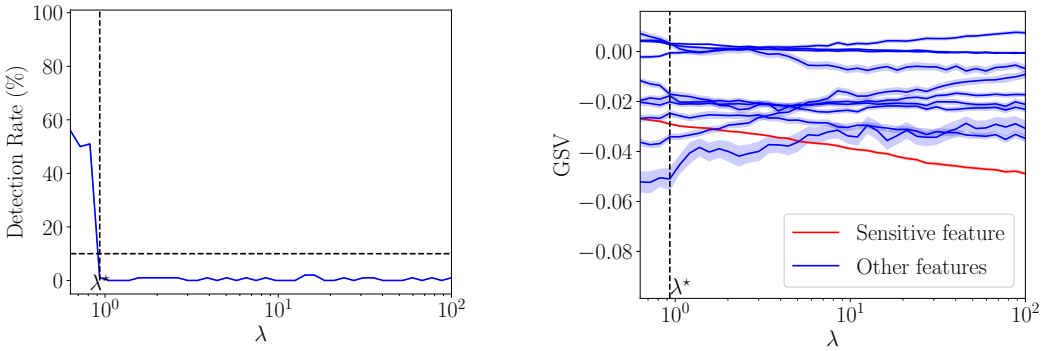


Figure 3: Example of log-space search over values of λ using a XGBoost classifier fitted on Adults. (a) The detection rate as a function of the parameter λ of the attack. The attacker uses a detection rate threshold $\tau = 10\%$. (b) For each value of λ, the vertical slice of the 11 curves is the GSV obtained with the resulting $\mathcal{B}'_\omega$. The goal here is to reduce the amplitude of the sensitive feature (red curve) in order to hide its direct effect when explaining the disparity in model outcomes.

4.4 Contributions

We clarify our contributions. The first technique to fool SHAP with perturbations of the background distribution was conducted by via a genetic algorithm Baniecki & Biecek (2022). However, the cross-over and mutation operations it employs to preturb data do not take into account feature correlations and can therefore generate unrealistic data. Moreover, the objective to minimize does not enforce a similarity between the original and manipulated backgrounds. We show in Appendix E.3 that these two limitations lead to systematic detections of the fraud. Hence, our contribution are two-fold

- By perturbing the background via non-uniform weights over pre-existing instances (*i.e.* $\mathcal{B}'_\omega := \mathcal{C}(D_1, \omega)$) rather than a genetic algorithm, we avoid the issue of non-realistic data.
- By adding the Wasserstein distance to the optimization objective, we can control the similarity between the original and manipulated backgrounds.

Since the technique introduced in Fukuchi et al. (2020) also leverages a non-uniform distribution over data points and the Wasserstein distance, it makes sense to adapt it for fooling SHAP. Still, the approach of Fukuchi et al. is different from ours. Indeed, in their work, they minimize the Wasserstein distance while enforcing a hard constraint on the number of instances that land on the different bins for the target and sensitive feature, That way, they can set the Demographic Parity to any given value while staying as close as to the original data. In our setting of manipulating the model explanation, we leave the Demographic Parity intact and instead manipulate its feature attribution. From an mathematical point of view, we now minimize a Shapley value with a soft constraint on the Wasserstein distance.

5 Experiments

5.1 Toy experiment

The task is predicting whether an individual will be hired for a job that requires carrying heavy objects. The causal graph for this toy data is presented in Figure 4 (left). We observe that sex (S) influences height (H), and that both these features influence the Muscular Mass (M). In the end, the hiring decisions (Y) are only based on the two attributes relevant to the job: H and M. Also, two noise features $N1, N2$ were added. More details and justifications for this causal graph are discussed in Appendix D.1. Since strength and height (the two important qualifications for applicants) are correlated with sex, any model f that fits the data will exhibit some disparity in hiring rates between sexes. Although, if the decisions made by the model do not rely strongly on feature S, the company can argue in favor of deployment. GSVs are used by the audit to measure the amount by which the model relies on the sex feature, see Figure 4 (Middle). By manipulating SHAP with $M = 200$, the company is able to reduce the amplitude of the GSV of feature S. More importantly, the audit is not able to detect that the provided samples S'_0, S'_1 were cherry-picked, see Figure 4 (Right).

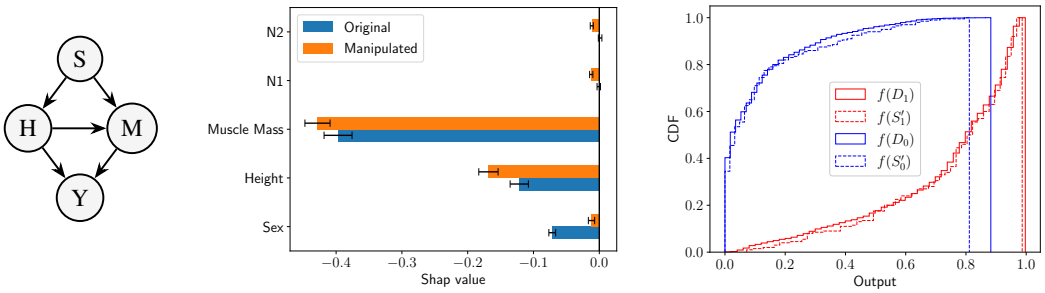


Figure 4: Toy example of hiring data for a job with specific physical requirements. Left: Causal graph. Middle: GSV before and after the attack with $M = 200$. Right: Comparison of the CDF of the misleading subsets $f(S'_0), f(S'_1)$ and the CDF over the whole data. $f(D_0), f(D_1)$. Here the audit is not able to detect the fraud using their detection algorithm.

5.2 Datasets

We consider four standard datasets from the FAccT literature, namely COMPAS, Adult-Income, Marketing, and Communities.

- **COMPAS** regroups 6,150 records from criminal offenders in Florida collected from 2013 and 2014. This binary classification task consists in predicting which individual will re-offend within two years. The sensitive feature s is `race` with values $x_s = 0$ for African-American and $x_s = 1$ for Caucasian.
- **Adult Income** contains demographic attributes of 48,842 individuals from the 1994 U.S. census. It is a binary classification problem with the goal of predicting whether or not a particular person makes more than 50K USD per year. The sensitive feature s in this dataset is `gender`, which took values $x_s = 0$ for female, and $x_s = 1$ for male.
- **Marketing** involves information of 41,175 customers of a Portuguese bank and the binary classification task is to predict who will subscribe to a term deposit. The sensitive attribute is `age` and took values $x_s = 0$ for age `30-60`, and $x_s = 1$ for age `not30-60`
- **Communities & Crime** contains per-capita violent crimes for 1994 different communities in the US. The binary classification task is to predict which communities have crimes below the median rate. The sensitive attribute is `PercentWhite` and took values $x_s = 0$ for `PercentWhite<90%`, and $x_s = 1$ for `PercentWhite>=90%`.

Three models were considered for the two datasets: Multi-Layered Perceptrons (MLP), Random Forests (RF), and eXtreme Gradient Boosted trees (XGB). One model of each type was fitted on each dataset for 5 different train/test splits seeds, resulting in 60 models total. Values of the test set accuracy and demographic parity for each model type and dataset are presented in Appendix D.2. We note that all demographic parity values are negative.

5.3 Detector Calibration

Detector calibration refers to the assessment that, assuming the null hypothesis to be true, the probability of rejecting it (i.e. false positive) should be bounded by the significance level α. Remember that the null hypothesis is that the sets S'_0, S'_1 provided by the company are sampled uniformly from D_0, D_1. Hence, to test the detector, the audit can sample their own subsets $f(S_0), f(S_1)$ uniformly from $f(D_0), f(D_1)$, run the detection algorithm, and count the number of detection over 1000 repeats. Table 1 presents the probabilities of false positives over the five train-test splits using a significance level $\alpha = 5\%$. We observe that the probability of false positives is indeed bounded by α for all model types and datasets implying that the detector employed by the audit is calibrated.

Table 1: Probability of False Positives (%) by the detector i.e. the probability that S_0, S_1 are considered cherry-picked when they are not. All probabilities should be under 5%.

	mlp	rf	xgb
COMPAS	3.6	3.4	3.1
Adult	3.5	3.7	3.8
Marketing		3.7	3.9
Communities		3.7	3.4

5.4 Attack Results and Discussion

The first step of the attack (line 3 of Algorithm 3) requires that the company run `SHAP` on their own and compute the necessary coefficients to run Algorithm 1. For the COMPAS and Adults datasets, the `ExactExplainer` of `SHAP` was used. Since Marketing and Communities contain more than 15 features, and since the `ExactExplainer` scales exponentially with the number of features, we were restricted to using the `TreeExplainer` (Lundberg et al., 2020) on these datasets. The `TreeExplainer` avoids the exponential cost of Shapley values but is only applicable to tree-based models such as RFs and XGBs. Therefore, we could not conduct the attack on MLPs fitted on Marketing and Communities.

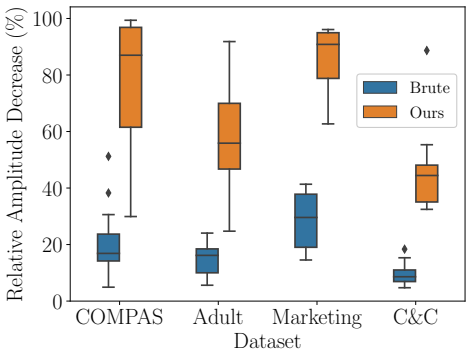


Figure 5: Relative decrease in amplitude of the sensitive feature attribution induced by the brute-force attack and Fool SHAP.

The following step is to solve the MCF for various values of λ (line 7 of Algorithm 3). As stated previously, solving the MCF can be done in polynomial time in terms of N_1, which was tractable for a small dataset like COMPAS and Communities, but not for larger datasets like Adult and Marketing. To solve this issue, as was done in Fukuchi et al. (2020), we compute the manipulated weights multiple times using 5 bootstrap sub-samples of D_1 of size 2000 to obtain a set of weights $\boldsymbol{\omega}^{[1]}, \boldsymbol{\omega}^{[2]}, \ldots, \boldsymbol{\omega}^{[5]}$ which we average to obtain the final weights $\boldsymbol{\omega}$.

Results of 46 attacks with $M = 200$ are shown in Figure 5. Specific examples of the conducted attacks are presented in Appendix E.2. As a point of reference, we also show the results for a brute-force approach consisting of repeatedly sampling $S_1' \sim \mathcal{B}^M$ in a honest manner, and choosing the set S_1' leading to the smallest attribution of the sensitive attribute. To make comparisons to our attack more meaningful, the brute-force method was only allowed to run for the same amount of time it took to search for the non-uniform weights $\boldsymbol{\omega}$ (about 30-180 seconds). We note that, across all datasets, Fool SHAP leads to greater reductions of the sensitive feature attribution compared to brute-force search. This result highlights the benefit of perturbing the background distribution compared to brute-force sampling.

Now focusing on Fool SHAP, for the datasets COMPAS and Marketing, we observe median reductions in amplitudes of about 90%. This means that our attack can considerably reduce the perceived importance of the sensitive attribute. For the Adult and Communities datasets, the median reduction in amplitude is about 50% meaning that we typically reduce by half the importance of the sensitive attribute. Still, looking at the maximum reduction in amplitude for both datasets, we note that one attack managed to reduce the amplitude by 90%. This implies that there is a luck factor to the degree of success of the attack, which is to be expected from data-driven attacks.

Finally, the audit was consistently unable to detect the fraud using statistical tests. This observation raises concerns about the risk that `SHAP` explanations can be attacked to return not only manipulated attributions but also non-detectable fake evidence of fairness.

6 Conclusion

To conclude, we proposed a novel attack on Shapley values that does not require modifying the model but rather manipulates the sampling procedure that estimates expectations w.r.t the background distributions. We show on a toy example and four fairness datasets that our attack can reduce the importance of a sensitive feature when explaining the difference in outcomes between groups using `SHAP`. Moreover, the sampling manipulation is hard to detect by an audit that is given limited access to the data and model. These results raise concerns about the viability of using Shapley values to assess model fairness. We leave as future work the use of Shapley values to decompose other fairness metrics such as predictive equality and equal opportunity. Moreover, we wish to move to use cases beyond fairness, as we believe that the vulnerability of Shapley values that was demonstrated can apply to many other properties such as safety and security.

7 ETHICS STATEMENT

The main objective of this work is to raise awareness about the risk of manipulation of SHAP explanations and their undetectability. As such, it aims at exposing the potential negative societal impacts of relying on such explanations. It remains however possible that malicious model producers could use this attack to mislead end users or cheat during an audit. However, we believe this paper makes a significant step toward increasing the vigilance of the community and fostering the development of trustworthy explanations methods. Furthermore, by showing how fairness can be manipulated in explanation contexts, this work contributes to the research on the certification of the fairness of automated decision-making systems.

8 REPRODUCIBILITY STATEMENT

The source code of all our experiments is will be made available online.Moreover, experimental details are provided in appendix D.2 for the interested reader.

REFERENCES

Julius A Adebayo et al. Fairml: Toolbox for diagnosing bias in predictive modeling. Master's thesis, Massachusetts Institute of Technology, 2016. 1

Ulrich Aïvodji, Hiromi Arai, Olivier Fortineau, Sébastien Gambs, Satoshi Hara, and Alain Tapp. Fairwashing: the risk of rationalization. In *International Conference on Machine Learning*, pp. 161–170. PMLR, 2019. 1

Hubert Baniecki and Przemyslaw Biecek. Manipulating shap via adversarial data perturbations (student abstract). 2022. 1, 7

Hubert Baniecki, Wojciech Kretowicz, and Przemyslaw Biecek. Fooling partial dependence via data poisoning. *arXiv preprint arXiv:2105.12837*, 2021. 1, 28

Tom Begley, Tobias Schwedes, Christopher Frye, and Ilya Feige. Explainability for fair machine learning. *arXiv preprint arXiv:2010.07389*, 2020. 1, 3

Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, et al. Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *arXiv preprint arXiv:1810.01943*, 2018. 1

Yoichi Chikahara, Shinsaku Sakaue, Akinori Fujino, and Hisashi Kashima. Learning individually fair classifier with path-specific causal-effect constraint. In *International Conference on Artificial Intelligence and Statistics*, pp. 145–153. PMLR, 2021. 1

Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, pp. 797–806, 2017. 3

Botty Dimanov, Umang Bhatt, Mateja Jamnik, and Adrian Weller. You shouldn't trust me: Learning models which conceal unfairness from multiple explanation methods. In *SafeAI@ AAAI*, 2020. 1

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pp. 214–226, 2012. 3

Kazuto Fukuchi, Satoshi Hara, and Takanori Maehara. Faking fairness via stealthily biased sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 412–419, 2020. 1, 2, 5, 7, 9

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42, 2018. 1

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016. 3

Ian Janssen, Steven B Heymsfield, ZiMian Wang, and Robert Ross. Skeletal muscle mass and distribution in 468 men and women aged 18–88 yr. *Journal of applied physiology*, 2000. 20

A J Lee. *U-statistics: Theory and Practice*. Routledge, 2019. 18

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. 1, 2

Scott M Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable ai for trees. *Nature machine intelligence*, 2(1):56–67, 2020. 9, 17

Frank J Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951. 19

Pedro Saleiro, Benedict Kuester, Loren Hinkson, Jesse London, Abby Stevens, Ari Anisfeld, Kit T Rodolfa, and Rayid Ghani. Aequitas: A bias and fairness audit toolkit. *arXiv preprint arXiv:1811.05577*, 2018. 1

Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, pp. 307–317, 1953. 2

Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 180–186, 2020. 1

Larry Wasserman. *All of Statistics: A concise course in statistical inference*. Springer, 2004. 13