Explaining Fairness Violations using Machine Unlearning

Tanmay Surve
Purdue University
West Lafayette, IN, USA
tsurve@purdue.edu

Romila Pradhan
Purdue University
West Lafayette, IN, USA
rpradhan@purdue.edu

ABSTRACT

Given the increasing prevalence of machine learning in critical domains, debugging machine-learning-based systems for discriminatory behavior is crucial. Discriminatory decisions in such systems can often be traced back to the data that the system has been trained upon. Recent advances in debugging fairness violations in machine learning models use influence functions that limits their applicability to a niche class of machine learning models – with loss functions that are convex and twice-differentiable. We focus on explaining an instance of fairness violation in non-parametric models by identifying the top-k predicate-based training data subsets attributable to the violation. We quantify the attribution of a subset to fairness violation in terms of the change in model fairness when the subset is removed. We introduce FUME, a system that **E**xplains a **F**airness violation by leveraging **M**achine **U**nlearning to efficiently estimate the change in model fairness when parts of the underlying training data are removed. To prioritize informative subsets in the huge search space of training data subsets, FUME navigates the subsets in the form of a hierarchically ordered space. Several pruning rules are adopted to avoid estimating subset attribution of unnecessary subsets. We showcase our solution on random forest classifiers which are one of the most widely used non-parametric machine learning models. We empirically evaluate the effectiveness and efficiency of FUME on several real-world and synthetic datasets, and demonstrate that the subsets determined by FUME are consistent with insights from prior studies on these datasets.

1 INTRODUCTION

Machine learning (ML) is fast becoming the standard choice for data science applications that involve automated decision-making in sensitive domains such as finance, healthcare, crime prevention, and justice management. Designed carefully, ML-based systems have the potential to eliminate the undesirable aspects of human decision-making such as biased judgments. However, concern continues to mount that these systems reinforce systemic biases and discrimination often reflected in their training data. For example, technology giants routinely come under the radar for discriminating against people based on their race, zip codes and perceived gender [6, 24, 44]; self-driving cars are less accurate at detecting pedestrians with darker skin tones [7]. Such discriminatory outcomes are harmful because they not only violate human rights but also impede and undermine societal trust in machine learning.

The need to *debug* and *explain* the causes for unexpected and discriminatory model behavior has propelled advances in the field of Explainable Artificial Intelligence (XAI) [10, 37, 53] that refers to the ability of an AI-based system to explain its decisions and actions in a way that is understandable, accountable, and transparent to humans. Much of XAI research has centered on generating *feature-based explanations* that explain the behavior of a machine learning model in terms of the input features or attributes of its training data [33, 51, 54, 59, 60]. With the advent of *data-centric* revolution in AI, *example-based explanations* have emerged as another powerful way to explain a model's behavior in terms of particular data instances that the model has been trained on [12, 34, 45, 58, 64]. Several recent approaches [18, 58] leveraged the concept of influence functions [45] to generate example-based explanations for discrimination or *bias* in model decisions and highlight training data instances responsible for model bias. However, due to the use of influence functions, these systems are limited to parametric machine learning models that have the added requirement of incorporating a loss function that is convex and twice-differentiable.

Our goal is to explain sources of bias in non-parametric machine learning models such as decision trees and ensemble tree-based models (e.g., random forest classifiers, gradient boosted decision trees) by identifying parts of the training data that can be attributed to the observed bias. We showcase our approach on random forest classifiers that are widely used in classification and regression tasks because they are successful in predictions and are computationally inexpensive compared to deep learning models. However, being an ensemble method, random forests are often considered difficult to interpret [26]. Because a random forest is a collection of trees, knowing the path that led to a decision is not possible (as opposed to a decision tree). While feature-based explanations determine features of a dataset important for a particular decision made by a random forest classifier, they cannot pinpoint to *data instances* that can be attributed to the decision. While demonstrated on random forest classifiers, the intuition behind the solutions developed in this paper is easily extensible to other non-parametric machine learning models (Section 5).

Discriminatory behavior, captured through *fairness* in the algorithmic literature, is broadly categorized as *individual fairness*, *group fairness* and *causal fairness*. Individual fairness [28] states that similar individuals must be treated similarly. Group fairness [52, 68] mandates parity between individuals belonging to different sensitive groups termed as *privileged* and *unprivileged* groups (e.g., males vs. non-males, Asians vs. non-Asians). Causal fairness, on the other hand, considers whether features have a causal effect on the fairness of outcomes [20, 46]. These notions of fairness are orthogonal to each other; we focus on group fairness which we detail further in Section 2.

Consider the following example that illustrates the need for determining causes of bias in a non-parametric model.

Example 1.1 (German Credit dataset). Consider an ML classification pipeline that ingests demographic and financial information about individuals (as in Figure 1(a)) and learns a random forest classifier that predicts whether an individual is a good credit risk (should be granted a loan) or a bad credit risk (denied a loan). The classifier has a high accuracy on unseen test data

© 2025 Copyright held by the owner/author(s). Published in Proceedings of the 28th International Conference on Extending Database Technology (EDBT), 25th March-28th March, 2025, ISBN 978-3-89318-099-8 on OpenProceedings.org.
Distribution of this paper is permitted under the terms of the Creative Commons license CC-by-nc-nd 4.0.

but exhibits disparity in predictions for individuals belonging to different age groups — those aged 45 and higher are 10% more likely to be classified as good credit risks compared to younger individuals. Note that this is an example of discrimination if, based on the model, younger applicants are routinely denied loans compared to older individuals.

A data scientist notes that this disparity is caused by the presence of unprivileged group in the training data receiving unfavorable outcomes. To explain the disparity, she therefore considers identifying paths in the trees of the random forest that mention the unprivileged group (younger individuals) and predict the unfavorable label (bad credit risk). Table 1 below presents the paths identified within the first few levels of three trees in the forest along with the number of samples denoted by the path.

Tree	Patterns	Size
1	(Savings > 500 DM) and (Age < 45)	9%
2	(Housing = Not renting) and (Status of checking account > 200 DM) and (Age < 45)	12.7%
	(Housing = Renting) and (Gender = Male) and (Purpose = Furniture) and (Age < 45)	0.02%
3	None found in the first five levels	-

Table 1: Patterns potentially explaining model bias.

This form of explanation is inadequate because of multiple reasons. First, we need to enlist the (possibly multiple) paths for all trees in the forest which may be expensive to compute. Second, it is difficult to summarize these paths. The two paths of tree 2 are disjoint and cannot be consolidated. The instances represented by paths of tree 1 and tree 2 may overlap; however, the overlapping pattern is not guaranteed to predict the unfavorable label. Moreover, one of the paths in tree 2 has just one sample (0.02%) and may not be relied upon as a potential cause of bias. Third, the combination of unprivileged group and unfavorable outcome may not occur in the first few levels of all trees; a pattern occurring at a deeper level becomes challenging for the data scientist to interpret.

The goal of this work is to highlight problematic subpopulations for the data scientist to subsequently inspect. Note that problematic instances are not guaranteed to be concentrated in certain subspaces; however, prior research has shown that data errors are inadvertently introduced for certain groups [42]. Presenting problematic data in the form of coherent subsets instead of individual data points will allow the data scientist to formulate hypotheses about potential data quality issues in the earlier stages of the data science pipeline (e.g., mislabeled instances in the unprivileged group), fixing which may improve the performance of the downstream model.

Determining parts of the training data that are attributable to model bias is a computationally expensive task: data instances must be evaluated for their effect on the downstream model bias and this computation must be performed in an efficient manner. A data scientist may judiciously peruse the data identified as attributable to model bias if presented in a format that is compact, easy to understand, and quantifies its contribution to model bias. Assuming access to a data scientist or data curator who can inspect the data instances for potential errors/issues, we are interested in the following question: what are the top-k training data subsets that are attributable to model bias? (Note that k is a user-defined parameter). This task is challenging because of a number of reasons. First, the subsets should be easily comprehensible. Second, we need to *efficiently* compute the attribution of each identified subset on the downstream model bias. Lastly, since the space of training data subsets is potentially large, we must prioritize computing the attribution of a small fraction of all subsets.

Human-understandable subset identification: To tackle the first challenge, we identify subsets in the form of predicates that are conjunctions of literals $\wedge_j(X_j \ op \ v_j)$ where X_j is an attribute, v_j is a corresponding attribute value and op can be one of $=, \neq, <, \leq, \geq, >$. For example, a possible subset for the scenario in Example 1.1 is (Age > 45) ∧ (Gender=Female). Such coherent subsets not only present informative data summaries compared to individual data instances but also highlight potential systemic issues in the earlier stages of the data science pipeline for specific subsets (e.g., possibly incorrect labels), and therefore, may reveal underlying discriminatory issues with how the data instances in these subsets are pre-processed.

Attributable subset and attribution to fairness violation: We need a way to determine which subset of the training data instances can be attributed to the discriminatory outcome. We call such subsets *attributable* to fairness violation violation and quantify their attribution by computing the change in model fairness when a new model is trained after removing the subset from the training data. Identifying responsible subsets is challenging because of two reasons. First, retraining the model to compute the attribution of each subset is computationally expensive. Instead, we need a faster way to estimate or approximate subset attribution without retraining the model. We realize that this problem is akin to the problem of efficiently learning an updated model when parts of the data that the model has been trained upon are deleted. Toward this goal, we utilize the concept of *machine unlearning* [14, 72] that refers to the ability of a model to *unlearn* a few training data instances and remove their impact on model predictions without having to retrain the model. Second, we need to evaluate all possible predicates to determine which of them can be attributed the most to model unfairness. It is practically infeasible to consider the huge search space over predicates which is exponential in the number of attributes and their distinct attribute values. To address this challenge, we utilize the idea of a hierarchically ordered lattice tree structure used in the apriori algorithm for frequent itemset mining [9], and adopt several pruning rules to reduce the subset search space.

Summary of contributions. Our contributions are as follows:

- We present FUME, a system that facilitates debugging of training data for explaining instances of fairness violations in non-parametric machine learning models by identifying training data subsets that can be attributed the most to model unfairness. (Overview of FUME in Figure 1).
- We formalize the notion of attribution of subsets to model unfairness and define the problem of determining training data subsets with the highest attribution to model unfairness (Section 2).
- We leverage concepts from *machine unlearning* to estimate the attribution of training data subsets to model unfairness (Section 3). To the best of our knowledge, FUME is the first system that uses machine unlearning for the problem of fairness debugging.

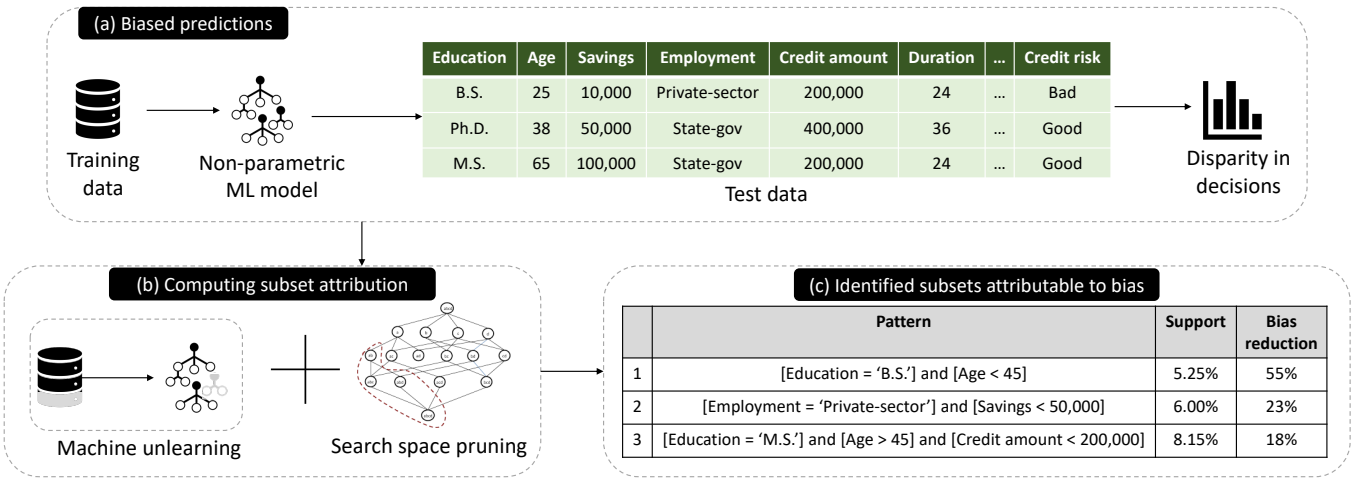


Figure 1: An overview of FUME. (a) Given a classifier that generates biased predictions on test data, (b) FUME uses machine unlearning and subset search space pruning to compute the attribution of each subset toward model fairness and determine the (c) top-k predicate-based training data subsets attributable to the fairness violation along with the attribution.

- To efficiently navigate the huge subset search space, we present an algorithm that utilizes a hierarchically ordered lattice structure and adopts several pruning strategies (Section 4).
- We provide experimental evaluation on several real-world and synthetic datasets, and show that for random forest classifiers, the training data subsets attributed to model unfairness are consistent with insights from prior studies (Section 6).

2 PROBLEM DEFINITION

We introduce the notations used throughout the paper, present relevant background information on classification and algorithmic fairness, and formally introduce the problem we solve in this paper.

2.1 Preliminaries

Classification. We consider the problem of binary classification and assume an instance space $\mathcal{X} \subseteq \mathbb{R}^p$ and binary labels $\mathcal{Y} = \{0, 1\}$. Let $\mathbf{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ be a training dataset where each instance $\mathbf{x}_i \in \mathcal{X}$ has p attributes and $y_i \in \mathcal{Y}$. The set of attributes in $\mathbf{D}$ is denoted by $\mathbf{X}$. Let $\mathcal{A} : \mathcal{X} \times \mathcal{Y} \rightarrow \mathcal{H}$ represent a learning algorithm defined as a function from a labeled dataset to a model in the hypothesis space $\mathcal{H}$. Let $h \in \mathcal{H}$ be the learned model obtained by training learning algorithm $\mathcal{A}$ on $\mathbf{D}$, and $\hat{Y}$ be the output space such that $\hat{y} = h(\mathbf{x})$ is its prediction on test data instance $\mathbf{x} \in \mathcal{X}$. In this paper, we consider non-parametric learning algorithms and focus on random forest classifiers as an example of such a model.

Group fairness. Given a binary classifier $h \in \mathcal{H}$ with output $\hat{Y}$ and a protected attribute $S \in \mathbf{X}$ (such as gender, race, age etc.), we interpret $\hat{Y} = 1$ as a favorable (positive) prediction and $\hat{Y} = 0$ as an unfavorable (negative) prediction. We assume the domain of S, $\text{Dom}(S) = \{0, 1\}$ where $S = 1$ indicates a privileged and $S = 0$ indicates a protected group (e.g., males and non-males, respectively). Group fairness mandates that individuals belonging to different groups must be treated similarly. The notion of similarity in treatment is captured by different associative notions of fairness [21, 52, 68]. We focus on the following widely used notions of group fairness:

- **Statistical parity:** A classifier h satisfies statistical parity if both protected and privileged groups have the same probability of being predicted the positive outcome i.e., $P(\hat{Y} = 1|S = 0) = P(\hat{Y} = 1|S = 1)$.
- **Equalized odds:** A classifier h satisfies equalized odds if the predictions $\hat{Y}$ and the sensitive attribute S are independent conditional on the true labels Y, i.e., $P(\hat{Y} = 1|S = 0, Y = y) = P(\hat{Y} = 1|S = 1, Y = y)$, where $y \in \{0, 1\}$. This definition states that the protected and privileged groups should have equal true positive rate and equal false positive rate.
- **Predictive parity:** A classifier h satisfies predictive parity if $P(Y = 1|S = 0, \hat{Y} = 1) = P(Y = 1|S = 1, \hat{Y} = 1)$ i.e., the likelihood of a positive label among individuals predicted as having a positive outcome is the same regardless of group membership.

These fairness metrics can be computed on both training data predictions and test data predictions. Fairness metric $\mathcal{F} : \mathcal{H} \times \mathbf{D} \rightarrow \mathbb{R}$ quantifies a given notion of group fairness computed over dataset $\mathbf{D}$. For example, per statistical parity, $\mathcal{F}(h, \mathbf{D}) = P(\hat{Y} = 1|S = 0) - P(\hat{Y} = 1|S = 1)$ where $P(\hat{Y} = 1|S = s), s \in \{0, 1\}$ is estimated on the prediction probabilities obtained by applying classifier h on $\mathbf{D}$. To satisfy group fairness, we introduce the notion of *group fairness violation* as follows:

Definition 2.1. **Group fairness violation.** Given dataset $\mathbf{D}$, classifier h, and fairness metric $\mathcal{F}$, group fairness violation occurs when $\mathcal{F}(h, \mathbf{D}) \neq 0$.

We refer to an instance of group fairness violation as *bias*; if $\mathcal{F}(h, \mathbf{D}) < 0$, h is biased against the protected group. The magnitude of the bias is denoted by $|\mathcal{F}(h, \mathbf{D})|$; the higher the magnitude of bias, the more biased the classifier's decisions are. Typically, in machine learning applications, fairness violations of h are measured on a test dataset $\mathbf{D}_{test} \in \mathcal{X} \times \mathcal{Y}$.

We are interested in determining subsets of the training data that are attributable to fairness violation on $\mathbf{D}_{test}$. Training data subset $\mathsf{T} \subseteq \mathbf{D}$ is attributed to the observed fairness violation considered if removing T from the training dataset and retraining a new classifier on the updated training data reduces bias. We say subset T is *attributable* to the violation and formally define it below:

Definition 2.2. **Attributable subset.** Given classifier h exhibits group fairness violation over predictions on dataset $\mathbf{D}_{test}$, training data subset $\mathsf{T} \subseteq \mathbf{D}$ is considered attributable to the violation if:

$$|\mathcal{F}(h_{\mathsf{T}}, \mathbf{D}_{test}| < |\mathcal{F}(h, \mathbf{D}_{test})|$$

where h_{T} is the classifier trained on subset $\mathbf{D} \setminus \mathsf{T}$ obtained after removing T from $\mathbf{D}$.

To determine if training data subset T is attributable to fairness violation, we need to quantify its attribution toward classifier bias, which is defined as follows:

Definition 2.3. **Subset attribution toward bias.** Given training data subset $\mathsf{T} \subseteq \mathbf{D}$ and classifier h, the contribution of T toward fairness violation $\mathcal{F}(h, \mathbf{D}_{test})$ is defined as:

$$\phi_{\mathsf{T}} = \frac{|\mathcal{F}(h_{\mathsf{T}}, \mathbf{D}_{test})| - |\mathcal{F}(h, \mathbf{D}_{test})|}{|\mathcal{F}(h, \mathbf{D}_{test})|}$$

In words, the attribution of a subset toward bias is the relative difference between bias of the original model and that of the new model obtained by training without the subset. We say that T is attributed to the fairness violation whenever $\phi_{\mathsf{T}} < 0$. The magnitude of ϕ_{T} quantifies the attribution of subset T to the classifier's bias on $\mathbf{D}_{test}$ predictions. The higher the magnitude, the higher the attribution of a subset toward bias.

Predicate-based subsets. We are interested in training data subsets that are represented by predicates because of their ease in interpretation. Subset T is represented by a conjunction of literals, i.e., $\mathsf{T} = \wedge_j X_j \; op \; v_j$ where $X_j \in \mathbf{X}, v_j \in \mathrm{Dom}(X_j)$ and $op \in \{=, \neq, <, \leq, \geq, >\}$. In Example 1.1, a possible subset would be T = (Age > 45) $\wedge$ (Gender=Female). Moreover, we are interested in subsets that contain a minimum number of data instances, denoted by τ. We define the *support* of subset $\mathsf{T} \subseteq \mathbf{D}$ as the fraction of data instances in $\mathbf{D}$ that are contained in T, i.e., $sup(\mathsf{T}) = |\mathsf{T}|/|\mathbf{D}|$.

Given these preliminaries, we are interested in identifying the top-k training data subsets attributable to a classifier's fairness violation. Formally, we seek to answer the following question:

Problem Statement. Given classifier h trained on $\mathbf{D}$ and evaluated on $\mathbf{D}_{test}$, fairness metric $\mathcal{F}$, support threshold τ and parameter k, we address the problem of identifying the top–k predicate-based subsets $\{\mathsf{T}_i\}_{i=1}^{k} \in \mathbf{D}$ such that $\forall i \; sup(\mathsf{T}_i) > \tau$, $\phi(\mathsf{T}_i) < 0$ and for $1 \leq i < j \leq k, |\phi(\mathsf{T}_i)| > |\phi(\mathsf{T}_j)|$.

3 ESTIMATING ATTRIBUTION TO BIAS

The naïve way of computing the attribution of a subset to model bias involves removing the subset from the training data, learning a new model with the modified training data and comparing the fairness of this new model with that of the original model. However, this approach constitutes retraining the model with each subset deletion, which is a time-consuming task. To address this challenge, we observe that learning the new model from scratch without the subset is akin to removing the effect of the subset from the trained model without retraining — a concept introduced by the recent field of *machine unlearning* [72].

Machine unlearning. The goal of machine unlearning is to *unlearn* or *forget* particular training data instances by updating a trained model to completely remove the effect of those instances. Given training dataset $\mathbf{D}$, model $\mathcal{A}(\mathbf{D})$, and data instance $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ that we want to remove from $\mathbf{D}$, removal method $\mathcal{R} : \mathcal{A}(\mathbf{D}) \times \mathbf{D} \times (\mathcal{X} \times \mathcal{Y}) \rightarrow \mathcal{H}$ is a function that maps the dataset without $(\mathbf{x}, y)$ to a new model in the hypothesis space $\mathcal{H}$ [15, 35]. The naïve retraining approach learns a new model in the hypothesis space $\mathcal{H}$ from scratch by retraining $\mathcal{A}$ on the modified dataset $\mathbf{D} \setminus (\mathbf{x}, y)$. For *exact unlearning*, the removal method must be equivalent to applying the learning algorithm to the dataset after removing training data instance $(\mathbf{x}, y)$. On the other hand, *approximate learning* ensures that the distribution of the *unlearned* model and that of a retrained model are approximately indistinguishable [72]. In that case, equivalence is defined as having identical probabilities for each model in $\mathcal{H}$ i.e., $P(\mathcal{A}(\mathbf{D} \setminus (\mathbf{x}, y))) = P(\mathcal{R}(\mathcal{A}(\mathbf{D}), \mathbf{D}, (\mathbf{x}, y)))$ In our problem setting, we are interested in removal methods that consider removal of data instances in subset $\mathsf{T} \subseteq \mathcal{D}$ i.e.,

$$P(\mathcal{A}(\mathbf{D} \setminus \mathsf{T})) = P(\mathcal{R}(\mathcal{A}(\mathbf{D}), \mathbf{D}, \mathsf{T})) \tag{1}$$

More details on various machine unlearning desiderata and mechanisms can be found in recent surveys on the topic [56, 72].

Computing subset attribution using machine unlearning. Given the definition of subset attribution to bias (Definition 2.3), we propose to compute the attribution of a subset using machine unlearning (Equation 1) as:

$$\phi_{\mathsf{T}} = \frac{|\mathcal{F}(\mathcal{R}(\mathcal{A}(\mathbf{D}), \mathbf{D}, \mathsf{T}), \mathbf{D}_{test})| - |\mathcal{F}(h, \mathbf{D}_{test})|}{|\mathcal{F}(h, \mathbf{D}_{test})|} \tag{2}$$

where $\mathcal{R}$ represents the removal method. Quantifying the attribution of a subset does not remove it from training data but estimates the change in model fairness *were the model trained without the subset.* This computation of subset attribution to model unfairness does not depend on the knowledge of the internals of the model and any model-agnostic machine unlearning approach [13, 16, 31, 36, 38, 40, 55, 65] can be used for the removal method $\mathcal{R}$.

Determining the top-k training data subsets that can be attributed to model bias then involves computing the attribution of each subset and ranking them in decreasing order of their attributions. This process is computationally expensive because of the huge subset search space ($O(2^p \times d)$ for p attributes with d distinct values per attribute). In the following section, we present our strategy to reduce the huge subset search space.

4 PRUNING THE SUBSET SEARCH SPACE

Utilizing DaRE-RF, we efficiently compute the attribution of a subset to model bias. However, there are a large number of training data subsets (exponential in the number of attributes and their distinct values), which makes this computation prohibitively expensive. FUME renders the problem tractable by employing several pruning techniques that reduce the subset search space.

To navigate the search space of all possible training data subsets, we employ the lattice structure (e.g., Figure 2) borrowed from the concept of the apriori algorithm in frequent itemset mining [9]. The lattice is a hierarchically ordered space where each lattice node corresponds to a unique subset in the training dataset represented by a conjunction of literals. We generate the lattice starting at level 1 that has subsets represented by just one literal (i.e., all attribute-value pairs in the training dataset). For p attributes with d distinct values per attribute, level 1 will have $d \times p$ subsets represented by nodes. Subsequent levels follow this rule: lattice level l has subsets represented by l literals that are generated by merging two nodes of level $l - 1$ having exactly $(l - 2)$ literals in common.

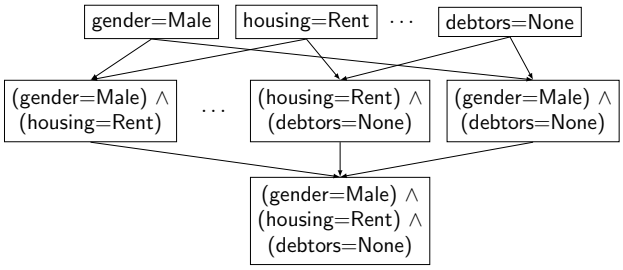


Figure 2: Visualization of the hierarchically ordered lattice structure for subset generation. At level 1, all nodes consist of a single literal. At each level, literals are merged two at a time, as illustrated, to generate subsequent subsets.

Greedy expansion of the lattice. The aforementioned process generates all possible subsets in a dataset, which are then pruned using the following rules to yield fewer subsets:

Rule 1: Prune irrelevant subsets. While generating subsets by navigating the lattice structure, we ensure that impractical subsets e.g., (Age < 50) ∧ (Age > 70) are not generated.

Rule 2: Filter subsets depending upon a support threshold. Depending on the domain, we might want the subsets attributable to bias to lie within some specific *support* range only. For example, a data scientist might want to inspect larger subsets (e.g., individuals in California state resulting in ∼ 40% of data) to highlight potential systemic biases whereas sometimes it might be beneficial to identify smaller subsets (e.g., younger individuals with divorced marital status that correspond to ∼ 3% of the data) to highlight potential data errors in that cohort. This choice is especially important while attributing subsets to model bias. Deleting subsets having large support may be undesirable due to the resulting large reduction in training dataset size. Upon expanding levels in the lattice structure, progressively complex subsets (with more literals) are generated that have smaller support compared to subsets higher up in the lattice. If a node is encountered such that the subset represented by it has a support smaller than the minimum support threshold, we do not expand its subtree as its children (with more literals and stricter conditions) will have an even smaller support. For example, for a support threshold of $(5-15)\%$, assuming the node (Debtors = None) has support of 4% in Figure 2, the node will not be used to generate nodes at the next level. In such a scenario, nodes (Gender = Male) ∧ (Debtors = None) and (Housing = Rent) ∧ (Debtors = None) will never be created as they would have support <= 4%. On the other hand, a subset with support more than the maximum user-defined support level is excluded from the set of identified subsets attributable to bias. However, such a subset is considered for further expansion of the lattice structure because it may generate subsets in the required support range in subsequent levels.

Rule 3: Prune complex subsets. The complexity of a subset is indicated by its interpretability which is reflected by the number of literals used to represent the subset. The higher the number of literals, the less interpretable the subset. A subset represented by three literals is more comprehensible than one having ten literals. Limiting the number of literals desired in the identified subsets effects a stopping condition for expanding the lattice structure. In Figure 2, enforcing interpretability at 2 literals in a subset, the lattice tree will only be expanded till level 2; nodes at level 3 e.g., (Gender = Male) ∧ (Housing = Rent) ∧ (Debtors = None) will never be generated.

Rule 4: Prune subsets with lower attribution to bias than its parent subsets. Even with the aforementioned pruning techniques, we must ensure that no redundant branches of the lattice structure are traversed that might result in expanding subsets that eventually are not attributable to model bias. Consider subset S representing a node in the lattice structure. Let S_1 and S_2 represent the subsets that S was merged from in the lattice structure. Let ϕ_S, ϕ_{S1} and ϕ_{S2} represent the subset contribution of S, S_1 and S_2 respectively computed using Equation 2. Unlike accuracy and empirical loss, fairness metrics are not additive. Therefore, we cannot leverage the downward closure property typical of the apriori algorithm to reason about ϕ_S from ϕ_{S1} and ϕ_{S2}. To address this limitation, we adopt the following heuristic: if the attribution of subset S to model bias is lower than either of its parents S_1 and S_2, i.e., $\phi_S < \phi_{S1}$ or $\phi_S < \phi_{S2}$, then S is considered to be of a worse quality and the node representing subset S is not expanded. The intuition behind this heuristic is that a merged subset with a lower attribution to bias compared to its parents' is devoid of influential pockets of training data (which are present in the subsets represented by its parents), and hence is not considered a beneficial route to follow.

Rule 5: Prune subsets that cannot be attributed to model bias. This rule ensures that a node in the lattice structure is expanded only if the attribution of the S represented by the node is positive, i.e., $\phi_S > 0$. The intuition behind this rule is that we are only interested in subsets that, when removed from the training data, improve upon the fairness of the original model.

5 FUME ALGORITHM

Now that the subset search space is reduced (Section 4), in this section we outline our algorithm to generate the top-k training data subsets attributable to model bias. We present FUME, a framework that combines the computation of subset attribution using machine unlearning (Section 3) with the subset pruning phase (Section 4) to determine training data subsets that can explain the bias. FUME expands the lattice structure in a top-down manner, generates subsets at levels 1 and 2, and computes their subset attribution. These subsets are considered *candidate* subsets attributable to model bias, their attribution to bias is computed and the space of candidate subsets is pruned using the pruning rules in Section 4. Subsets represented by nodes in the subsequent levels are generated in accordance with the pruning rules. The algorithm stops when either the maximum depth of the lattice is reached (governed by Rule 3) or there are no more subsets within the desired support range. The algorithm stops early if there are not enough nodes present to merge for the next level. The candidate subsets are then sorted in decreasing order of their attribution to bias, and the top-k subsets with the highest attribution to bias are generated as output. Algorithm 1 presents the pseudocode for this process of identifying the top-k training data subsets attributable to model bias.

5.1 Subset attribution for non-parametric models

While model-agnostic machine unlearning techniques are applicable to a wide range of machine learning algorithms, there are schemes designed to leverage the special traits of particular learning algorithms but are not universally compatible [72].

Algorithm 1: FUME: Generation of top-k training data subsets attributable to model bias

Input: maxLiterals η, supportRange, $\tau = [\tau_{min}, \tau_{max}]$
Output: top-k subsets $\mathcal{E} = \{\mathsf{T}_i\}_{i=1}^{k}$

```
C = [ ]                                          ▹ candidate subsets
level = 1
E ← ExpandSubsets(C, level)                      ▹ Rule 1
ℰ = [ ]
                                                 ▹ Rule 3
while level ≤ η do
    for subset ∈ E do
                                                 ▹ Rule 2
        if support(subset) > τ_max then
            C.append(subset)
            continue
        if support(subset) < τ_min then
            continue
                                                 ▹ Rules 4 and 5
        selected ← EstimateAttribution(subset)
        if selected then
            C.append(subset)
            ℰ.append(subset)
    level = level + 1                            ▹ Rule 1
    E ← ExpandSubsets(C, level)
    if E is empty then
        BREAK
return ℰ
```

For example, for non-parametric models such solutions include Hedgecut [63] and DaRE-RF [15] that are specifically designed for tree-based models. Hedgecut focuses on improving the efficiency of the unlearning process by proposing a classification model based on extremely randomized trees (ERTs) [19] and assumes that a tiny fraction of data instances can be deleted (more details in [63]). DaRE-RF, on the other hand, is an *exact* unlearning method specific to random forest classifiers. In the following, we explore the applicability of one such machine unlearning method for computing the attribution of a subset toward unfairness in a random forest classifier which is one of the simplest non-parametric model.

Machine unlearning for random forests. Specifically designed for random forests, Data Removal-Enabled Random Forests (DaRE-RF) [15] proposes a random forest variant that enables the efficient removal of training data instances for *exact* unlearning. The key intuition is to retrain subtrees in each tree of forest only as needed.

DaRE-RF construction. To implement the unlearning process, DaRE-RF leverages two primary techniques. First, it ensures that only those portions of the random forest are retrained where the structure must change to match the updated database. This property is made possible by introducing *randomness* at the top of each tree in the forest such that the splitting attribute and threshold (at most k per attribute) are chosen randomly. Doing so ensures that these nodes minimally depend on the data. and hence, in the event of a deletion, they rarely need to be retrained. Second, DaRE-RF utilizes *caching* to store data statistics throughout each tree. For decision nodes, information on the number of data instances and data instances with positive labels are stored along with the corresponding information for a set of k' thresholds per attribute. Similar information is stored and updated for leaf nodes along with a list of data instances contained in that leaf node. These data statistics are initialized when the tree is trained for the first time.

DaRE-RF unlearning. The saved statistics ensure that we know which subtrees to focus on during unlearning. When a training data instance $(x, y) \in \mathcal{D}$ is deleted, the saved statistics are updated and used to check if a particular subtree needs to be retrained. The statistics for internal tree nodes that are affected by the deletion of (x, y) are updated, and the splitting criterion for each attribute-threshold pair is recomputed. The thresholds are then checked for validity: if a different threshold provides an improved splitting criterion over the currently chosen threshold, the subtree rooted at this node is retrained by obtaining the list of data instances from all leaf node descendants of this subtree. If an internal node does not require retraining and a leaf node is reached instead, then its label counts and list of data instances are updated and the deletion operation is complete. Therefor, the saved statistics are sufficient to recompute the splitting criterion of each threshold and construct the updated tree without iterating through the data.

Effectiveness of DaRE-RF. By compromising the space complexity (required to store data statistics on the internal and leaf nodes), DaRE-RF greatly reduces the recomputation cost associated with unlearning. It must be highlighted that the **data deletion process for DaRE-RF is exact** — which implies that removing instances from a DaRE model yields exactly the same model as retraining from scratch on updated data. Empirically, the impact of deleting a single training data instance from a trained model on the model's predictive performance using has been shown to be within a test error difference of 1% [15].

Estimating subset attribution to bias using DaRE-RF. Because DaRE-RF is an *exact* unlearning technique, the DaRE-RF model is effective in updating the original random forest classifier after a few training data instances have been deleted with minimal effect on model accuracy. The resultant DaRE-RF model, therefore, is the same as retraining a random forest classifier after removing a few training data instances. We extend this idea to measure the difference in bias on test data instances and propose using the DaRE-RF model to compute the impact of deleting training data instances in a subset on model bias.

DaRE-RF Complexity for deleting subsets. The time complexity of updating the DaRE-RF model for deletion of a training data subset follows that of updating the tree for deletion of a single data instance. Given that the constructed DaRE-RF model has random nodes up to d_{max} depth, $\tilde{p}$ random attributes to be considered at each splitting node, and k' randomly chosen thresholds, if the tree structure does not change, the time complexity to delete subset $\mathsf{T} \in \mathcal{D}$ is $O(\tilde{p}k'd_{max})$. If deletion results in a node with invalid thresholds, then the time to choose new thresholds is $O(|D| \log |D|)$ where $|D|$ represents the number of instances in the node. If a node with $|D|$ instances at level l needs to be retrained, then the additional retraining time is $O(\tilde{p}|D|(d_{max} - l))$.

Updated DaRE-RF unlearning for training data subsets. Consider a subset $\mathsf{T} = \{(x, y)\} \subset \mathbf{D}$ for deletion from a given random forest

classifier. The removal of a subset of training data instances might incur changes in more internal nodes than when a single data instance is removed. The unlearning process then iterates over all internal nodes that are affected by the deletion of T. The statistics for each of the nodes are updated and each such node is checked for improvement in splitting criterion with different attribute-threshold pairs. Each node is retrained and leaf nodes are updated as needed.

Given a subset $\mathsf{T} \subset \mathbf{D}$, the updated DaRE-RF model is used as the removal method $\mathcal{R}$ to compute ϕ_{T} in Equation 2. We demonstrate that this model correctly estimates the subset attribution toward bias for smaller subsets (with support $0 - 5\%$) and incurs a difference of up to 25% for subsets with $5 - 15\%$ support (Section 6.2, Figure 3).

Complexity. Given p attributes and at most d attribute values per attribute, FUME considers pd subsets and performs DaRE-RF model update operations pd times for level 1 and presents them as subsets attributable to model bias if their attribution is positive (i.e., bias is reduced). Subsets at the subsequent levels are generated according to the rules outlined in Section 4.

Extensibility to other non-parametric models. As discussed in Section 2, the computation of subset attribution in Equation 2 does not require the knowledge of the internals of the machine learning model. Our approach can be easily extended to any parametric or non-parametric machine learning model by changing the EstimateAttribution( ) function (line 15) in Algorithm 1 to incorporate either one of the model-agnostic machine unlearning approaches [13, 16, 31, 36, 38, 40, 55, 65] or a model-specific machine unlearning approach [56, 72].

6 EXPERIMENTAL EVALUATION

In this section, we answer the following research questions. **RQ1:** How effective is machine unlearning in capturing the effect of subset removal on the fairness of a DaRE-RF model? **RQ2:** How effective and interpretable are the top-k subsets that can be attributed to model unfairness as identified by FUME? What is the quality of the identified subsets? **RQ3:** How efficient is FUME in identifying attributable subsets over datasets with varying characteristics?

6.1 Experimental Setup

6.1.1 Datasets. We demonstrate the effectiveness of FUME on the following real-world datasets:
German Credit [27] contains financial information of 1,000 individuals; sensitive attribute: *age*. Prediction task determines whether an individual is a good/bad credit risk.
Adult Census Income [48] contains demographic and financial information of 48,844 individuals. Prediction task: determine whether an individual has annual income $\leq 50k$ or $> 50k$; sensitive attribute: *sex*.
Stop-Question-Frisk (SQF) [3] contains demographic and stop-related information for 72,548 individuals who were stopped and questioned (and possibly frisked) by the NYC Police Department (NYPD). Classification task: predict if a stopped individual will be frisked; sensitive attribute: *race*.
ACS Income [4, 25] is extracted from the 2015 US-wide American Community Survey (ACS) Public Use Microdata Sample (PUMS) census data and contains demographic and employment-related information. We used a subset of the dataset pertaining to 139,866 individuals from the state of California. Classification task: predict whether an individual's income is greater than \$50,000; sensitive attribute: *sex*.
Medical Expenditure Panel Survey (MEPS) [1, 2] is a large-scale survey of individuals and families that collects data regarding healthcare usage, cost, and health insurance coverage. The data consists of medical information for 11,081 individuals from 2015 Panel 19. Classification task: predict whether an individual has high utilization of medical care and services; sensitive attribute: *race*.

More details on the datasets are seen in Table 2. For all the experiments, the numerical columns in each dataset have been discretized to explore subsets. Details on other preprocessing steps after data loading has been made available in the code repository.

6.1.2 ML model. We use the data removal-enabled random forest (DaRE-RF) [15], which is a random forest classifier that supports exact unlearning of data instances, for all the classification tasks.

6.1.3 Metrics. As discussed in Section 2, FUME supports three fairness metrics: statistical parity, predictive parity and equalizing odds parity [52, 68]. We compute the corresponding fairness metric difference between the privileged and protected groups as the fairness criteria where a zero difference indicates an unbiased model. We report the **parity reduction** effected by a training data subset in terms of the percent by which model fairness is reduced after removing the subset from the training dataset and training a new model on the updated training dataset.

6.1.4 Baseline. We use DropUnprivUnfavor, the baseline considered in Example 1.1, that trains a random forest classifier on a modified training dataset obtained after removing training data instances where the unprivileged group have unfavorable outcome. Removing such instances reduces the dependency of the classifier on the sensitive attribute and the learned model is expected to exhibit lower bias than when trained on the entire data.

Hyperparmeter settings. We set $k = 5$ and report the top-5 identified attributable subsets for each dataset. A higher value of k will overwhelm the practitioner since they will have to inspect more subgroups for potential errors while for $k < 5$, the subsets identified would be in the same order as listed for $k = 5$. As the value of k increases, the parity reduction of further subsets keeps decreasing and the benefit of parity reduction is outweighed by the effort of inspecting those additional subsets.

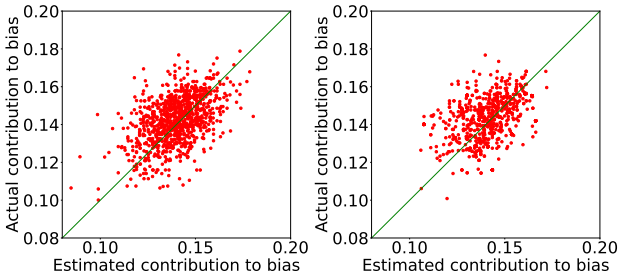
Hardware and Platform. The experiments were conducted on a 64-bit Windows OS with a 3.20 GHz AMD Ryzen 7 5800H processor and 32.0 GB memory. The algorithms were implemented in Python in Jupyter Notebook environment.
Source code. The code for FUME is publicly available at: https://github.com/responsible-data-science-lab/fume.

6.2 Effectiveness of machine unlearning in estimating subset attribution

The effect of DaRE-RF unlearning capabilities on model accuracy is known [15]: as *support* of a subset to be unlearned from the model increases, the accuracy of the updated model decreases. To test the applicability of DaRE-RF unlearning process to the use-case of debugging causes of fairness violations, we first need

Dataset	# instances	# features	Sensitive attribute	$\frac{\lvert Protected \rvert}{\lvert Dataset \rvert}$	Privileged base rate	Protected base rate
German Credit	1,000	21	age	41.10%	74.19%	63.99%
Adult Census Income	45,222	10	sex	32.50%	31.24%	11.35%
SQF	72,546	16	race	35.94%	38.32%	30.16%
ACS Income	139,833	10	sex	48.55%	43.53%	31.06%
MEPS	11,081	42	race	64.07%	25.49%	12.36%

Table 2: Summary of datasets.



(a) Random **(b) Coherent**

Figure 3: Effectiveness of DaRE-RF in estimating subset attribution to bias. Dots represent training data subsets. x-axis and y-axis respectively represent the actual and DaRE-RF-estimated attribution of a subset to bias for (a) random subsets, and (b) coherent subsets in the training data.

to validate its impact on model fairness. Toward this goal, we generate 1,000 *random* and 1,000 *coherent* subsets from the German credit dataset [27]. A *random* subset has data instances chosen randomly from the dataset while a *coherent* subset is defined in the form of conjunction of literals. We consider three support ranges: (0% – 5%) denoting small-sized subsets, [5% – 15%] denoting medium-sized subsets, and ≥ 30% denoting large subsets, and three fairness metrics: statistical parity, predictive parity and equalizing odds. Figure 3 shows the plots for *random* and *coherent* subsets for predictive parity fairness metric for the support range of [5% – 15%]. We see that most of the points in these plots lie on or clustered around the $y = x$ line (shown in green). This alignment means that the fairness metric calculated via retraining a model from scratch after deleting a subset from the training dataset is almost the same as that estimated using machine unlearning. Similar results were observed for different support ranges and across different fairness metrics (not shown here due to space constraints). Note that even with an increase in subset size, the fairness of the unlearned model, unlike accuracy, is approximately the same as that of the retrained model. This experiment highlights that model fairness is preserved (i.e., the fairness of the unlearned model is the same as that of the retrained model) by DaRE-RF unlearning across various support ranges and hence, can be used to estimate model fairness.

6.3 Effectiveness of subsets attributable to bias

We analyze the top-k subsets attributable to bias by evaluating how effective they are in explaining instances of group fairness violations. Our main goal is to assess if the identified attributable subsets are justified in the domain with respect to the underlying training data. Toward this goal, we analyze the attributable subsets using base rate difference and change in feature importance scores. Base rate differences: For each identified attributable subset, we compute the base rates (proportion of training data with positive labels) for both the sensitive groups. A higher base rate for the privileged group compared to the protected group indicates that a reasonably accurate model will reflect this inherent bias in its predictions causing the subset to be present in the top-k attributable subsets. Model feature importance deviations: We used scikit-learn's permutation_importance functionality to rank features in order of their importance toward a model's predictions. Subset deletion might impact feature importance rankings; comparing rankings over a model trained with and without a particular subset highlights the impact the feature has on model predictions. A decrease in the feature importance of the sensitive attributes after deleting the subset from the training data indicates that the correlation between the sensitive attribute and the outcome was reduced, causing a reduction in model bias.

Identified attributable subsets for German Credit: From Table 3 we can clearly see that the top-5 subsets (indicated by indices GS1 - GS5) remove almost all of the model bias, thus validating the effectiveness of FUME in successfully identifying training data subsets that can be attributed to model bias. These subsets represent a very small fraction of the entire training data and are represented by just two literals and, therefore, are easy to interpret. On inspection of subsets indicated by indices GS1 and GS2, we observed that the base rate of the privileged group was much higher than that of the protected group in both the cases (70% for privileged vs. 57% for protected group in GS1, and 70.1% for privileged vs. 62% for protected group in GS2). Similar behavior was also observed in the remaining attributable subsets. By removing these training data subsets, the correlation between the sensitive group and the target variable reduced, thus reducing model bias. In comparison, DropUnprivUnfavor removes 14.75% data points which reduces parity by 85.5%. We also observed that the attributable subsets on the same dataset are different for different fairness metrics. This observation highlights the fact that *the underlying training data subset does not contain a single (or more) subset can be solely attributed to the bias across multiple fairness metrics.*

Identified attributable subsets for Adult: We report the top-5 attributable subsets for this dataset in Table 4. We observe that the identified subsets (AS1 – AS5) exhibit considerable reduction in bias (34% - 52%). On closer inspection, it was found that the base rate difference in the sensitive groups for two of these identified subsets (AS2 and AS3) are not high. However, the feature importance of attributes relevant for income, such as occupation, increased respectively by 30.53% and 41.79% in these subsets. Although the correlation between the sensitive attribute

Index	Patterns	Support	Parity Reduction
GS1	Status of checking account = < 0 DM, Number of people liable = High	5.00%	97.79%
GS2	Savings = 100 ≤ . . . < 500 DM, Job = Skilled employee / official	7.13%	95.58%
GS3	Installment plans = Bank, Debtors = None	12.00%	93.38%
GS4	Status of checking account = No checking account, Property = Unknown / no property	5.25%	91.17%
GS5	Housing = Rent, Status and sex = Female divorced/separated/married	10.00%	89.91%

Table 3: Top-5 subsets attributable to statistical disparity in German Credit dataset in the support range 5% - 15%.

Index	Patterns	Support	Parity Reduction
AS1	Sex = Male, Education = Bachelors	11.67%	51.89%
AS2	Occupation = Sales, Age = Middle-aged	6.54%	36.43%
AS3	Occupation = Clerical administration	12.33%	35.53%
AS4	Age = Middle-aged, Workclass = Self employed no income	6.01%	34.39%
AS5	Relationship = Unmarried	10.64%	34.37%

Table 4: Top-5 subsets attributable to statistical disparity in Adult Census Income dataset in the support range 5% - 15%.

Index	Patterns	Support	Parity Reduction
SS1	Sex = Female	6.51%	100%
SS2	Weight = Light, Casing a victim = False	6.44%	39.95%
SS3	Build = Heavy, Fits a relevant description = False	6.87%	35.99%
SS4	Suspect acting as a lookout = False, Actions indicative of a drug transition = True	6.01%	33.83%
SS5	Weight = Light	7.81%	31.24%

Table 5: Top-5 subsets attributable to statistical disparity in Stop-Question-Frisk dataset in the support range 5% - 15%.

and target attribute is not completely broken, there is a drop in this correlation as indicated by the drop in feature importance of the sensitive attribute (by 34.42% and 34.87% upon deletion of subsets AS2 and AS3 respectively), thus reducing model bias. In contrast, DropUnprivUnfavor removes a large fraction of data (40.34%) resulting in a higher parity reduction (74.4%) but close to 15% drop in accuracy.

Identified attributable subsets for SQF: We report the identified attributable subsets for this dataset in Table 5 and observe that bias is reduced substantially by removal of each of the subsets. Note that the model bias is completely removed upon removal of the subset where Sex=Female. While the sensitive attribute in this dataset is race, we observed that in this identified subset, sex is highly correlated with race. By removing training data instances that are female, we break the model's dependence on sex and consequently on race, thus reducing the model bias. We also observed that the base rate difference between the sensitive groups is not very high in all of the subsets. However, the feature importance changes after deletion of these subsets, thus providing a clear explanation on why these subsets can be attributed to bias. Deleting SS1 and SS5 caused the importance of the feature Reason for stop: actions indicative of a drug transition to increase by 116.23% and 58.46%, respectively. Similarly, the importance of Reason for stop: casing a victim increased by 75.30% and 42.50% for SS1 and SS5, respectively. The importance of Reason for stop: suspect acting as a lookout also increased to 118.74% and 32.45% for SS1 and SS5, respectively. Similarly, the highest loss in feature importance was seen in Sex=Male, dropping by 100% and 30.45% in SS1 and SS5, respectively (which relates to our intuition that gender should not be an important criteria in determining whom to frisk). We also observed that training data instances where Weight=Light are attributable to bias in two of the top-5 identified subsets, thus indicating the importance of these attributes toward model bias. We found that DropUnprivUnfavor results in an 8x increase in disparity in the other direction by removing 44.8% of training data.

Identified attributable subsets for ACSIncome: We report the attributable subsets identified by FUME on this dataset in Table 6 and observe some reduction in bias for all identified subsets. Removing these subsets do not offer a huge reduction in bias. This behavior is largely explained by the dataset size — since the dataset contains more than 100, 000 data instances, it is unlikely that a small fraction (5 − 15%) of the data can be attributed to much of the model bias. With an increased support range (> 30%), we are able to find larger subsets that, when removed, reduce model bias by around 70%. We also do not observe any overlaps of data instances over the top-5 identified subsets. While the base rates were found to be comparable for the privileged and protected groups in these subsets, the reduction in bias is due to the difference in model predictions. However, considering the feature importance changes provides better insights: the importance of discriminatory attributes, such as race and sex, decreased across all subsets, thus explaining the reduction in bias, and the small magnitude of their changes (on an average by 53.10% for race and 6% for sex) explains the low reduction in bias. In contrast, attributes such as WorkClass show improved importance across all subsets (around 141% on an average) indicating that instead of race and sex, a non-discriminatory attribute is now deemed important for positive model predictions.

Attributable subsets for MEPS: We report the top-5 attributable subsets for this dataset in Table 7. We observe a high

Index	Patterns	Support	Parity Reduction
AC1	Hours worked per week = Overtime, WorkClass = Private	14.74%	27.32%
AC2	Age = Senior	10.41%	20.30%
AC3	Age = Middle-aged, School = ≥ 1 college credit but no degree,	9.59%	15.01%
AC4	Hours worked per week = Part-time	14.31%	14.21%
AC5	WorkClass = Local government	8.58%	12.30%

Table 6: Top-5 subsets attributable to statistical disparity in ACSIncome dataset in the support range 5% - 15%.

Index	Patterns	Support	Parity Reduction
ME1	Chronic bronchitis = No, Cancer diagnosis = True	5.86%	80.75%
ME2	Health insurance coverage = True, Employment Status = Unemployed	10.73%	76.01%
ME3	Emphysema diagnosis = No, Cancer diagnosis = True	5.81%	74.34%
ME4	Cognitive limitations = No, Cancer diagnosis = True	5.36%	72.92%
ME5	Cancer diagnosis = True	6.17%	71.49%

Table 7: Top-5 subsets attributable to statistical disparity in MEPS dataset in the support range 5% - 15%.

reduction in bias for all of the identified subsets, ME1 - ME5. Unsurprisingly, a positive cancer diagnosis was the common pattern in four of the five subsets; individuals with a positive cancer diagnosis routinely have a high usage and cost of medical services. Upon inspection of data instances in these subsets, it was found that a high expenditure was invariably related to the protected group. By removing these subsets, the feature importance of race decreased by around 20.51% indicating a decrease in the target attribute's dependence on race toward model prediction, and thus reducing model bias. The importance of the attribute ACTLIM (Any limitation - work/household/school), on the other hand, increased by 49.01% on an average, indicating that any kind of medical limitation should be causing an increase in expenditure. The reason for this disparity in model predictions might be due to substandard data collection or unjustifiable discrimination against the protected group. FUME identifies training data subsets that can be attributed to model bias; the next step toward mitigating bias would be highlighting potential data errors and biases inherent in those specific subsets that manifested themselves in an increased model bias. DropUnprivUnfavor removes a much larger fraction of data (55.8%) with a lower reduction in parity (72.2%).

Quality of identified attributable subsets. To further understand and evaluate the quality of the attributable subsets identified by FUME, we present the average and maximum bias reductions (Figure 4) achieved by the top-5 attributable subsets for each of the datasets for various support ranges. From experiments on specific datasets, it was found that for some datasets (e.g., German), the bias reduction is high (> 90%) while the least bias reduction of 12.3% was obtained in ACSIncome. In the German dataset, on an average, the identified attributable subsets reduce a high fraction of bias across all support ranges. However, for some of the other datasets, we are able to identify attributable subsets in higher support ranges only. For example, in the support range of > 30% bias is reduced by around 70% for ACSIncome even though we are not able to achieve such high bias reduction in the $5-15\%$ support range. As the size of the subset to be deleted increases, a large number of training data instances are removed — one would expect an improvement in fairness but a greater loss in model accuracy. However, the model accuracy scores after removing the attributable subsets across all datasets for the $5-15\%$ support range were not found to decrease proportional to the bias reduction: the maximum accuracy reduction observed for this scenario was around 4%. These observations indicate that it is easier to determine tiny attributable training data subsets for the smaller datasets while for larger datasets, we need to look for larger attributable subsets.

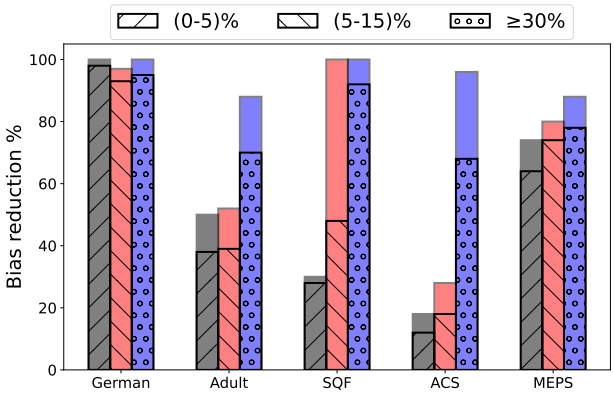


Figure 4: Quality of attributable subsets identified for different support ranges. Solid bars in the background and patterned bars in the foreground respectively indicate the maximum and average bias reduction for a given support.

6.4 Efficiency of FUME

We observed from the previous sections that the attributable subsets generated by FUME are effective and interpretable. However, it is also important to consider the efficiency of FUME in generating the attributable subsets. In this set of experiments, we measure efficiency by reporting the runtime of FUME, and consider how the runtime is impacted by various dataset characteristics e.g., dataset dimensions ($= n \times p$ where n represents the number of data instances and p is the number of attributes in the dataset), number of attributes, and number of unique attribute values.

In Table 8, we observe that as the dataset dimension increases exponentially from German Credit to ACSIncome (by a factor

Dataset	Dimension	Time (sec)
German credit	21,000 **(1x)**	130.96 **(1x)**
Adult Income	452,220 **(21.5x)**	687.92 **(5.3x)**
MEPS	465,402 **(22.16x)**	2388.48 **(18.23x)**
SQF	1,160,736 **(55x)**	5268.25 **(40.2x)**
ACS Income	1, 398, 330 **(66.58x)**	8615.04 **(65.78x)**

Table 8: FUME runtime on real-world datasets. Dimension = |Dataset| × |Attributes|.

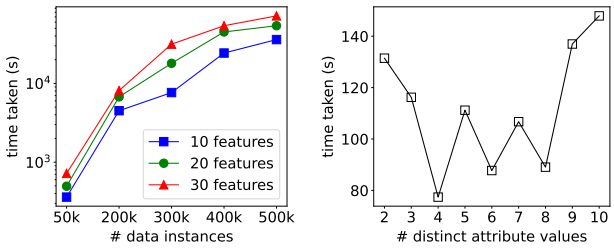
of 66.58x), the runtime of FUME increases quite linearly initially, increasing exponentially MEPS onward. We conclude that FUME works efficiently for datasets having fewer dimensions, but does not scale well as the dataset dimensions increase. To verify this hypothesis, we evaluated FUME on synthetic datasets with varying characteristics. In Figure 5a, we report the time taken by FUME for training data sets with varying number of attributes and fixed (= 2) unique attribute values per attribute. We observed that as the number of instances increases, FUME runtime increases rapidly, which is expected as larger subset removals would incur more processing for updating the DaRE tree. Note that FUME is efficient for smaller datasets (<50k size). As the number of attributes in a dataset grows, FUME requires more time to retrieve the attributable subsets. In Figure 5b, we report the time taken by FUME to generate attributable subsets for a training data set with 30, 000 instances and 10 attributes, and vary the number of distinct values per attribute. We observe that increasing the number of distinct attribute values does not indicate any clear pattern with runtime. This behavior is attributed to the fact that even though there are more subsets to consider, a large fraction is pruned by the rules. FUME runtime, thus is governed by the number of subsets that invoke model unlearning and estimating subset attribution to bias.

Effect of pruning. FUME adopts a number of pruning rules that reduces the search space of subsets to explore instead of estimating the effect of removing each subset and ranking them in decreasing order of the estimated effect. To evaluate the impact of pruning rules, we report the gain in subset exploration as a lattice structure is expanded level by level. In Table 9, we report

Level	1	2	3	4
Possible subsets	69	2, 346	873, 181	125, 751
Subsets explored	59	1, 322	502	416
Subsets pruned (%)	10	43.65	99.94	99.67

Table 9: Effect of pruning on subset exploration.

the observed statistics for the German Credit dataset. The first level represents the most general subsets, most of which satisfy the support threshold and are not pruned. Level 2 joins two subsets at a time and prunes close to half of them resulting in a large number of subsets to explore in level 3 (and level 4). Subsets at deeper levels are more restrictive (in terms of satisfying the pattern) and naturally a large fraction of them do not meet the support threshold. A considerable number of subsets that do meet this threshold are further pruned by rules 4 and 5, resulting in less than 1% of subsets that need to be evaluated.



(a) Effect of dataset size **(b) Effect of attribute values**

Figure 5: Efficiency of FUME.

7 RELATED WORK

Our work is broadly related to the following lines of recent research:

Explainable AI. Our research is mainly related to the broad field of explainable artificial intelligence [10, 37, 53] that is aimed at ensuring that the decisions made by an AI-based system are transparent to and understandable by different stakeholders of the system. XAI techniques primarily generate explanations for model decisions in terms of features or examples of the underlying training data. *Feature-based* explanations [33, 51, 54, 59, 60, 69] identify input features of the training data that are deemed the most important by the ML model for predicting positive outcomes. *Example-based* explanations focus on identifying training data instances that are the most responsible for particular ML model decisions. These explanations hinge on the *valuation* of data instances through techniques such as influence functions [12, 23, 45, 58], their variants [64] and data Shapley values [34]. Recent approaches [18, 58] leveraged influence functions to generate example-based explanations for debugging instances of fairness violations in parametric ML models. The proposed solutions are not directly applicable to the problem of debugging fairness violations in non-parametric models.

Algorithmic Bias. Ensuring fairness is imperative in algorithmic decision-making systems that are prevalent in safety-critical applications. A number of fairness metrics have been proposed [52] to quantify bias and are broadly categorized as individual fairness [28], group fairness [52, 68] and causal fairness [20, 46]. Individual fairness states that similar individuals must be treated similarly. Group fairness mandates parity between individuals belonging to different sensitive groups. Causal fairness considers the causal effect of features on the fairness of outcomes. These metrics are orthogonal to each other and work under different underlying assumptions. We focus on group fairness; our solution of removing training data subsets may change the outcome for individuals, thus potentially violating individual fairness. Identifying subsets attributable to individual fairness violation is an interesting area of research that is deferred to future work. Several bias mitigation techniques have been introduced [52] that are categorized as *pre-processing*, *in-processing*, and *post-processing* [52], and typically involve modifying one of three components of an AI-based system, namely data, model, and model predictions, respectively. *Pre-processing* [43] is usually model agnostic and assumes access to and transforms the underlying training data such that a model learnt on the transformed data exhibits lower unfairness compared to the original model. *In-processing* [73] assumes access to the model's learning algorithm and either tweaks its objective function or adds fairness

constraints to satisfy fairness. *Post-processing* [41] assumes access to only the model predictions; methods to satisfy fairness include reassigning labels obtained by the model based on a function (e..g, changing thresholds for different sensitive groups). Our work is the most similar to pre-processing as we modify the underlying training data by deleting subsets. While pre-processing seeks to reduce bias, we diagnose the system for fairness violations and identify parts of the underlying data responsible for model unfairness. A related line of research proposed in [11] acquires some of the patterns identified as having inadequate representation to reduce model bias (which is not guaranteed [29]). In contrast, we estimate the exact change in model bias effected by removal of patterns and directly identify patterns that, if removed, will reduce model bias.

Machine Unlearning. With the introduction of international regulations, such as the European Union's General Data Protection Regulation (GDPR) [67], the California Consumer Privacy Act (CCPA) [5], and Canada's proposed Consumer Privacy Protection Act (CPPA) [8], that center around users' *right to be forgotten*, and concerns around the security and privacy of users' data, the field of *machine unlearning* [14, 15, 17, 22, 35, 39, 47, 49, 62, 63, 66, 72] has gained much popularity in recent years. The need for machine unlearning also stems from concerns around security and privacy of individuals whose data is present in the underlying training data for the ML model. Malicious actors might get access to individuals' sensitive data by exploiting system security vulnerabilities or by inference through model's predictions [66]; these dangers mandate that when an individual requests their data to be removed, it is not enough to remove it from the training data but also that their effect on the model be *unlearned* [72]. While machine unlearning is in nascent stages, the idea of using unlearning to forget the effect of data instances on a model has shown promising results. Researchers have only recently started to explore the fairness implications of unlearning techniques [74], and found that some frameworks (e.g., SISA [14]) have minimal impact on fairness for both uniform and non-uniform data deletions. We studied the fairness impacts of DaRE-RF and observed it to mostly preserve fairness with both random and coherent data deletions; this line of research needs further investigation. To the best of our knowledge, FUME is the first system that uses the concept of machine unlearning for the problem of fairness debugging. Although we focus on unlearning in random forest classifiers, the techniques presented are extensible to other ML models; we leave leveraging unlearning for debugging instances of fairness violations in other ML models for future work.

Debugging Data-based Systems. Debugging training data has long been considered a way to explain the performance of data-driven systems, especially ML-based systems [30, 32, 50, 57, 61, 70, 71]. Slice Finder [57] and SliceLine [61] identify *slices* of the training data where the model performs worse compared to the rest of the data. These works indirectly detect unfairness by identifying subpopulations where the model does not perform well. Our work goes beyond bias detection and addresses the problem of determining the root causes of the observed unfairness by tracing the disparity in test data back to the underlying training data. Furthermore, these works focus on model performance metrics (e.g., accuracy, log loss) which are additive in nature, and are not directly applicable to our problem that considers non-additive group fairness metrics. Our solutions can, however, be generalized beyond the fairness task and any operation that can *unlearn* could also benefit from the proposed work.

8 CONCLUSION

We proposed FUME, a system for identifying the top-k coherent training data subsets that can be attributed to instances of group fairness violations in the outcomes of a non-parametric machine learning model. FUME hinges on machine unlearning techniques to efficiently compute the attribution of subsets to model bias, and utilizes a hierarchically ordered lattice structure based on the apriori algorithm in frequent itemset mining and utilizes several pruning rules to prune the huge subset search space. To the best of our knowledge, FUME is the first system to leverage machine unlearning in the context of fairness to explain the discriminatory behavior of a machine learning model. Experimental evaluation on several real-world and synthetic datasets demonstrated that the subsets attributed to bias as identified by FUME incur a substantial reduction in model bias, and are consistent with prior studies on these datasets. While we focused on identifying subsets attributed to bias in random forest classifiers, the proposed approach can easily be applied to other parametric and non-parametric classifiers. We consider further investigations on utilizing machine unlearning for fairness debugging of black-box ML models for future work.

Acknowledgments

We would like to thank the anonymous reviewers for their insightful feedback. Funding for this research was provided by the National Science Foundation (NSF) under Grant# 2237149, Google Research Scholar program, and the UL Research Institutes through the Center for Advancing Safety of Machine Intelligence.

REFERENCES

[1] 2015 full year consolidated data file meps https://meps.ahrq.gov/mepsweb/data_stats/download_data_files_detail.jsp?cboPufNumber=HC-181.
[2] Aif360 meps documentation https://aif360.readthedocs.io/en/latest/modules/generated/aif360.datasets.MEPSDataset19.html#aif360.datasets.MEPSDataset19.
[3] Nypd stop, question and frisk data. https://www1.nyc.gov/site/nypd/stats/reports-analysis/stopfrisk.page. [Online; accessed 19-October-2021].
[4] United states census bureau. 2018 acs pums documentation. 2019. https://www.census.gov/programs-surveys/acs/microdata/documentation.2018.html.
[5] California consumer privacy act (CCPA). https://oag.ca.gov/privacy/ccpa, 2018. [Online; accessed 24-Nov-2023].
[6] Housing department slaps facebook with discrimination charge. lhttps://www.npr.org/2019/03/28/707614254/hud-slaps-facebook-with-housing-discrimination-charge, 2019.
[7] Self-driving cars more likely to hit blacks. lhttps://www.technologyreview.com/2019/03/01/136808/self-driving-cars-are-coming-but-accidents-may-not-be-evenly-distributed/, 2019.
[8] Consumer privacy protection act (CPPA). https://blog.didomi.io/en-us/canada-data-privacy-law, 2022. [Online; accessed 24-Nov-2023].
[9] Agrawal, R., and Srikant, R. Fast algorithms for mining association rules in large databases. In *VLDB* (1994), pp. 487–499.
[10] Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion 58* (2020), 82–115.
[11] Asudeh, A., Jin, Z., and Jagadish, H. V. Assessing and remedying coverage for a given dataset. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)* (2019), pp. 554–565.
[12] Basu, S., You, X., and Feizi, S. On second-order group influence functions for black-box predictions. In *ICML* (2020), pp. 715–724.
[13] Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C., Jia, H., Travers, A., Zhang, B., Lie, D., and Papernot, N. Machine unlearning. In *Proceedings - 2021 IEEE Symposium on Security and Privacy, SP 2021*, pp. 141–159.
[14] Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C. A., Jia, H., Travers, A., Zhang, B., Lie, D., and Papernot, N. Machine unlearning. *arXiv preprint arXiv:1912.03817* (2019).

[15] Brophy, J., and Lowd, D. Machine unlearning for random forests. In *Proceedings of the 38th International Conference on Machine Learning* (2021).

[16] Cao, Y., and Yang, J. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy* (2015), pp. 463–480.

[17] Chen, M., Zhang, Z., Wang, T., Backes, M., Humbert, M., and Zhang, Y. Graph unlearning. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security* (New York, NY, USA, 2022), CCS '22, Association for Computing Machinery, p. 499–513.

[18] Chen, R., Yang, J., Xiong, H., Bai, J., Hu, T., Hao, J., Feng, Y., Zhou, J. T., Wu, J., and Liu, Z. Fast model debias with machine unlearning, 2023.

[19] Cheng, H., Shi, Y., Wu, L., Guo, Y., and Xiong, N. An intelligent scheme for big data recovery in internet of things based on multi-attribute assistance and extremely randomized trees. *Information Sciences 557* (2021), 66–83.

[20] Chiappa, S. Path-specific counterfactual fairness. In *AAAI* (2019), vol. 33, pp. 7801–7808.

[21] Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *CoRR abs/1703.00056* (2017).

[22] Cohen, A., Smith, A., Swanberg, M., and Vasudevan, P. N. Control, confidentiality, and the right to be forgotten. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security* (2023), CCS '23, p. 3358–3372.

[23] Cook, D. R., and Weisberg, S. Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics 22* (1980).

[24] Dastin, J. Rpt-insight-amazon scraps secret ai recruiting tool that showed bias against women. *Reuters* (2018).

[25] Ding, F., Hardt, M., Miller, J., and Schmidt, L. Retiring adult: New datasets for fair machine learning. *Advances in neural information processing systems 34* (2021), 6478–6490.

[26] Dorador, A. Improving the accuracy and interpretability of random forests via forest pruning, 2024.

[27] Dua, D., and Graff, C. Uci machine learning repository, 2017.

[28] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. S. Fairness through awareness. In *ITCS* (2012), pp. 214–226.

[29] Ekta, and Pradhan, R. Valuation-based data acqusition for machine learning fairness. In *VLDB Workshops* (2024).

[30] Flokas, L., Wu, W., Liu, Y., Wang, J., Verma, N., and Wu, E. Complaint-driven training data debugging at interactive speeds. In *Proceedings of the 2022 International Conference on Management of Data* (2022), SIGMOD '22, p. 369–383.

[31] Fraboni, Y., Van Waerebeke, M., Scaman, K., Vidal, R., Kameni, L., and Lorenzi, M. SIFU: Sequential informed federated unlearning for efficient and provable client unlearning in federated optimization. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics* (2024), vol. 238 of *Proceedings of Machine Learning Research*, PMLR, pp. 3457–3465.

[32] Galhotra, S., Fariha, A., Lourenço, R., Freire, J., Meliou, A., and Srivastava, D. Dataprism: Exposing disconnect between data and systems. In *Proceedings of the 2022 International Conference on Management of Data* (2022), SIGMOD '22, Association for Computing Machinery, p. 217–231.

[33] Galhotra, S., Pradhan, R., and Salimi, B. Explaining black-box algorithms using probabilistic contrastive counterfactuals. In *Proceedings of the 2021 International Conference on Management of Data* (2021), pp. 577–590.

[34] Ghorbani, A., and Zou, J. Y. Data shapley: Equitable valuation of data for machine learning. In *ICML* (2019), K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97, pp. 2242–2251.

[35] Ginart, A. A., Guan, M. Y., Valiant, G., and Zou, J. *Making AI Forget You: Data Deletion in Machine Learning*. 2019.

[36] Golatkar, A., Achille, A., and Soatto, S. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Los Alamitos, CA, USA, jun 2020), IEEE Computer Society, pp. 9301–9309.

[37] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. A survey of methods for explaining black box models. *CSUR 51*, 5 (2018), 1–42.

[38] Guo, C., Goldstein, T., Hannun, A., and Van Der Maaten, L. Certified data removal from machine learning models. In *Proceedings of the 37th International Conference on Machine Learning* (2020), ICML'20, JMLR.org.

[39] Gupta, V., Jung, C., Neel, S., Roth, A., Sharifi-Malvajerdi, S., and Waites, C. Adaptive machine unlearning. *arXiv preprint arXiv:2106.04378* (2021).

[40] Gupta, V., Jung, C., Neel, S., Roth, A., Sharifi-Malvajerdi, S., and Waites, C. Adaptive machine unlearning. In *Proceedings of the 35th International Conference on Neural Information Processing Systems* (2024), NIPS '21.

[41] Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. In *NIPS* (2016), pp. 3315–3323.

[42] Jiang, H., and Nachum, O. Identifying and correcting label bias in machine learning. In *AISTATS* (2020), S. Chiappa and R. Calandra, Eds., vol. 108 of *Proceedings of Machine Learning Research*, PMLR, pp. 702–712.

[43] Kamiran, F., and Calders, T. Classifying without discriminating. In *2009 2nd International Conference on Computer, Control and Communication* (2009), pp. 1–6.

[44] Kay, M., Matuszek, C., and Munson, S. A. Unequal representation and gender stereotypes in image search results for occupations. In *CHI* (2015).

[45] Koh, P. W., and Liang, P. Understanding black-box predictions via influence functions. In *Proceedings of Machine Learning Research* (2017), D. Precup and Y. W. Teh, Eds., vol. 70, pp. 1885–1894.

[46] Kusner, M. J., Loftus, J. R., Russell, C., and Silva, R. Counterfactual fairness. In *NIPS* (2017), pp. 4069–4079.

[47] Lee, S., and Woo, S. S. Undo: Effective and accurate unlearning method for deep neural networks. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (New York, NY, USA, 2023), CIKM '23, Association for Computing Machinery, p. 4043–4047.

[48] Lichman, M. Uci machine learning repository, 2013.

[49] Lin, H., Chung, J. W., Lao, Y., and Zhao, W. Machine unlearning in gradient boosting decision trees. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2023), KDD '23, Association for Computing Machinery, p. 1374–1383.

[50] Lourenço, R., Freire, J., and Shasha, D. Bugdoc: Algorithms to debug computational processes. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 2020), SIGMOD '20, Association for Computing Machinery, p. 463–478.

[51] Lundberg, S. M., and Lee, S.-I. A unified approach to interpreting model predictions. In *NIPS* (2017), pp. 4765–4774.

[52] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. A survey on bias and fairness in machine learning. *ACM Comput. Surv. 54*, 6 (jul 2021).

[53] Molnar, C. *Interpretable Machine Learning*. Lulu. com, 2020.

[54] Mothilal, R. K., Sharma, A., and Tan, C. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (2020), pp. 607–617.

[55] Nguyen, Q. P., Oikawa, R., Divakaran, D. M., Chan, M. C., and Low, B. K. H. Markov chain monte carlo-based machine unlearning: Unlearning what needs to be forgotten. In *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security* (2022), ASIA CCS '22, p. 351–363.

[56] Nguyen, T. T., Huynh, T. T., Nguyen, P. L., Liew, A. W., Yin, H., and Nguyen, Q. V. H. A survey of machine unlearning. *CoRR abs/2209.02299* (2022).

[57] Polyzotis, N., Whang, S., Kraska, T. K., and Chung, Y. Slice finder: Automated data slicing for model validation. In *ICDE* (2019).

[58] Pradhan, R., Zhu, J., Glavic, B., and Salimi, B. Interpretable data-based explanations for fairness debugging. In *Proceedings of the 2022 International Conference on Management of Data* (New York, NY, USA, 2022), SIGMOD '22, Association for Computing Machinery, p. 247–261.

[59] Ribeiro, M. T., Singh, S., and Guestrin, C. "Why should I trust you?" explaining the predictions of any classifier. In *SIGKDD* (2016), pp. 1135–1144.

[60] Ribeiro, M. T., Singh, S., and Guestrin, C. Anchors: High-precision model-agnostic explanations. In *AAAI* (2018), vol. 18, pp. 1527–1535.

[61] Sagadeeva, S., and Boehm, M. Sliceline: Fast, linear-algebra-based slice finding for ml model debugging. In *SIGMOD* (2021), pp. 2290–2299.

[62] Schelter, S., Ariannezhad, M., and de Rijke, M. Forget me now: Fast and exact unlearning in neighborhood-based recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (New York, NY, USA, 2023), SIGIR '23, Association for Computing Machinery, p. 2011–2015.

[63] Schelter, S., Grafberger, S., and Dunning, T. Hedgecut: Maintaining randomised trees for low-latency machine unlearning. In *Proceedings of the 2021 International Conference on Management of Data* (2021), pp. 1545–1557.

[64] Sharchilev, B., Ustinovskiy, Y., Serdyukov, P., and de Rijke, M. Finding influential training samples for gradient boosted decision trees. In *Proceedings of the 35th International Conference on Machine Learning* (10–15 Jul 2018), J. Dy and A. Krause, Eds., vol. 80 of *Proceedings of Machine Learning Research*, PMLR, pp. 4577–4585.

[65] Thudi, A., Deza, G., Chandrasekaran, V., and Papernot, N. Unrolling sgd: Understanding factors influencing machine unlearning. In *2022 IEEE 7th European Symposium on Security and Privacy* (Los Alamitos, CA, USA, jun 2022), IEEE Computer Society, pp. 303–319.

[66] Ullah, E., Mai, T., Rao, A., Rossi, R. A., and Arora, R. Machine unlearning via algorithmic stability. In *Conference on Learning Theory* (2021), PMLR, pp. 4126–4142.

[67] Union, T. E. Regulation (eu) 2016/679: General data protection regulation (GDPR). lhttps://gdpr-info.eu/, 2016. [Online; accessed 17-Feb-2019].

[68] Verma, S., and Rubin, J. Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)* (2018), pp. 1–7.

[69] Wachter, S., Mittelstadt, B., and Russell, C. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech. 31* (2017), 841.

[70] Wang, X., Dong, X. L., and Meliou, A. Data x-ray: A diagnostic tool for data errors. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 2015), SIGMOD '15, Association for Computing Machinery, p. 1231–1245.

[71] Wu, W., Flokas, L., Wu, E., and Wang, J. Complaint-driven training data debugging for query 2.0. In *Proceedings of the ACM International Conference on Managment of Data (SIGMOD)* (2020).

[72] Xu, H., Zhu, T., Zhang, L., Zhou, W., and Yu, P. S. Machine unlearning: A survey. *ACM Computing Surveys 56*, 1 (aug 2023).

[73] Zhang, B. H., Lemoine, B., and Mitchell, M. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society* (New York, NY, USA, 2018), AIES '18, Association for Computing Machinery, p. 335–340.

[74] Zhang, D., Pan, S., Hoang, T., Xing, Z., Staples, M., Xu, X., Yao, L., Lu, Q., and Zhu, L. To be forgotten or to be fair: Unveiling fairness implications of machine unlearning methods, 2023.