Causal Feature Selection for Algorithmic Fairness

Sainyam Galhotra
University of Chicago
sainyam@uchicago.edu

Prasanna Sattigeri
IBM Research AI
psattig@us.ibm.com

Karthykeyan Shanmugam
IBM Research AI
karthykeyan.shanmugam2@ibm.com

Kush R. Varshney
IBM Research AI
krvarshn@us.ibm.com

ABSTRACT
The use of machine learning (ML) in high-stakes societal decisions has encouraged the consideration of fairness throughout the ML lifecycle. Although data integration is one of the primary steps to generate high-quality training data, most of the fairness literature ignores this stage. In this work, we consider fairness in the integration component of data management, aiming to identify features that improve prediction without adding any bias to the dataset. We work under the causal fairness paradigm [46]. Without requiring the underlying structural causal model a priori, we propose an approach to identify a sub-collection of features that ensure fairness of the dataset by performing conditional independence tests between different subsets of features. We use group testing to improve the complexity of the approach. We theoretically prove the correctness of the proposed algorithm and show that sub-linear conditional independence tests are sufficient to identify these variables. A detailed empirical evaluation is performed on real-world datasets to demonstrate the efficacy and efficiency of our technique.

CCS CONCEPTS
- Theory of computation → Machine learning theory.

KEYWORDS
Causal fairness, feature selection, fair machine learning

ACM Reference Format:
Sainyam Galhotra, Karthykeyan Shanmugam, Prasanna Sattigeri, and Kush R. Varshney. 2022. Causal Feature Selection for Algorithmic Fairness. In Proceedings of the 2022 International Conference on Management of Data (SIGMOD '22), June 12–17, 2022, Philadelphia, PA, USA. ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3514221.3517909

1 INTRODUCTION
Algorithmic fairness is of great societal concern when supervised classification models are used to support allocation decisions in high-stake applications. There have been numerous recent advances in statistically and causally defining group fairness between populations delineated by protected attributes and in the development of algorithms to mitigate unwanted bias [5]. Bias mitigation algorithms are often categorized into pre-processing, in-processing, and post-processing approaches. Pre-processing techniques modify the distribution of the training data, in-processing techniques modify the objective function of the training procedure or consider additional constraints in the learning phase, and post-processing techniques modify the output predictions — all in service of improving fairness metrics while upholding classification accuracy [11]. Table 1 summarizes a representative set of prior bias mitigation algorithms. However, this categorization misses an important stage in the lifecycle of machine learning practice: data collection, engineering and management [26, 47]. Holstein et al. [21] report that practitioners “typically look to their training datasets, not their ML models, as the most important place to intervene to improve fairness in their products”. Data integration, one of the first components of data management, aims to join together information from different sources that captures rich context and improves predictive ability. With the phenomenal growth of digital data, ML practitioners may procure features from millions of sources spanning data lakes, knowledge graphs, etc [15, 38]. They typically generate exhaustive sets of features from all sources and then perform subset selection [15, 33, 53]. Feature selection is a promising direction for fairness in ML as it does not require assumptions about data distribution and is robust to distribution shifts [12], assuming distribution shifts do not change the structural aspects of the causal model. Some may argue that data integration is a part of pre-processing but we make this distinction as data integration does not involve modification of the data distribution and is considered as the task of a data engineer as opposed to a data modeler.

Filtering methods for feature selection exploit the correlation of features to identify a subset [19]. However, these techniques are ignorant of sensitive attributes and fairness concerns. For example, consider a dataset with features $F_1$ and $F_2$ such that $F_1$ provides slightly more improvement in accuracy than $F_2$; however, incorporating $F_1$ yields a classifier that reinforces discrimination against protected groups whereas incorporating $F_2$ yields a classifier with similar outcomes for different groups. Feature selection techniques that are not discrimination-aware will prefer $F_1$ to $F_2$, but $F_2$ is a better feature to select from a societal perspective.

Table 1: Different categories of fairness techniques. 

| Pre/Post-processing | Feature Selection |
|---------------------|-------------------|
| [7, 14, 27]         | [9, 25, 46]       |
| [6, 8, 20, 28, 52]  | [40, 44]          |

1 We use the terms sensitive attribute and protected attribute interchangeably.
To overcome the fairness limitations of standard feature selection methods, we study the problem of fair feature selection, specifically in the context of data integration when we are integrating new tables of features with an existing training dataset (PK-FK joins) or source selection or generating new features using transformations [12, 15, 31]. Our goal is to identify a subset of new features that can be integrated with the original dataset without worsening its biases against protected groups. As an additional advantage, the feature selection paradigm is known to be stable against changes in data distribution as compared to prior techniques that modify the output predictions or the data distribution to mitigate bias [48]. Following the framework of prior fair algorithms [9, 10, 46], we assume access to protected/sensitive attributes which are used to identify the feature subset that obey fairness. The identification of features that do not induce additional bias is tricky because of relationships between non-protected attributes and protected ones that allow the reconstruction of information in the protected attributes from one or more non-protected ones. For example, zip code can reconstructed race information [23].

There are two main types of techniques to ensure fairness in data: Associational and Causal (summarized in Table 1). Associational techniques look for associative relationships between sensitive attributes and the prediction outcome to mitigate unwanted biases. However, these techniques are based on correlation between attributes and fail to capture causal relationships. There has been a lot of interest in studying causal frameworks [9, 10, 25, 29, 30, 32, 35, 41, 54, 55] to achieve fairness. Due to their ability to distinguish different discrimination mechanisms, we use causal fairness [37, 46] as our fairness framework. Certain causal approaches assume access to the underlying causal structure, which is unrealistic in practice [9, 10, 25, 30, 35, 51, 55]. Importantly, we do not make the assumption that we are given the causal graph (formally, the structure of the causal bayesian network that generates the data) a priori.

We propose an algorithm SeqSel to identify all new features that when added to the original dataset still ensure causal fairness. Our algorithm takes as input a dataset D comprising an outcome variable, sensitive features, admissible features, and a collection of features that are neither admissible nor sensitive. A feature is considered admissible if the protected variables are allowed to affect the outcome through it. For example, consider a credit card application system that contains gender and race as sensitive attributes, expected monthly usage as an admissible attribute (it may have a sensitive attribute as one of its parent but it is permissible for the sensitive attribute to influence the outcome through this variable), and age and education level as variables which are neither sensitive nor admissible. A set of features X is considered to ensure causal fairness if after adding these features one could increase accuracy of a subsequently trained classifier on this new dataset without worrying about causal fairness metrics, i.e. in effect the subset of features when added does not introduce any tradeoff between fairness and accuracy and they are safe to subsequent attempts at building a purely predictive classifier. Our approach operates in two phases focused towards performing conditional independence tests with respect to the sensitive attributes and the target variable. These tests help identify variables that (1) do not capture information about sensitive attributes, or (2) ensure fairness even if they capture some information about sensitive attributes. We theoretically prove that both types of these variables ensure causal fairness and analyze the conditions to identify all such variables.

The naïve SeqSel algorithm performs a number of conditional independence tests that grows linearly in the number of features in the dataset. One of the major shortcomings of extant conditional independence testing methods is that they generate spurious correlations between variables if too many tests are performed [50]. To overcome this limitation and reduce the chances of getting spurious results, we propose a more efficient algorithm, GrpSel, that uses graphoid axioms to show that group testing can reduce the number of tests to the logarithm of the number of features and additionally improves the overall efficiency of the pipeline.

Our primary contributions are:

- We formalize the problem of fairness in data integration and feature selection setting using causal fairness.
- We provide an algorithm that performs conditional independence tests to identify the variables that do not worsen the fairness of the dataset.
- We prove theoretical guarantees that the variables identified by our algorithm ensure fairness and identify a closed form expression for variables that cannot be added.
- We propose an improved algorithm that leverages ideas of group testing to reduce the chances of getting spurious correlations and has sub-linear complexity.
- We show empirical benefits of our techniques on synthetic and real-world datasets.

The paper represents a principled use to address an important problem that has not been addressed before: fair data integration.

## 2 PRELIMINARIES

In this section, we review the background on algorithmic fairness and models of causality.

We denote variables (also known as dataset attributes or features) by uppercase letters like X, S, A, corresponding values in lower case like x, s, a, and sets of attributes or values in bold (X or x).

### 2.1 Algorithmic Fairness

The area of algorithmic fairness aims to ensure unbiased output for different sub-groups identified by specific set of attributes (also known as protected or sensitive attributes). For example, a loan prediction software should not discriminate against female applicants (gender is the protected attribute). The literature on algorithmic fairness considers a set of protected attributes $S = \{S_1, \ldots, S_k\}$, a target variable Y and a prediction algorithm $f : V \to Y$ where V denotes the set of input attributes and the output of $f$ is called the prediction output or an outcome. Typically, ML tasks train a classifier on a dataset D (comprising of attributes V and target Y) which is assumed to be distributed according to a distribution Pr. In order to measure the fairness of $f$ with respect to S, two different types of metrics have been studied: Associational and Causal.

**Associational fairness** methods capture statistical variabilities in the behavior of the prediction algorithm for different groups of individuals. For example, equalized odds requires that the false positive
and true positive rate of different sub-groups identified by the sensitive attributes is the same. Other associational fairness measures include Demographic parity, conditional statistical parity, and predictive parity [6–8, 20, 28, 52]. Even though associational methods of quantifying fairness are very popular, all these methods fail to distinguish between causal influence and spurious correlations between different input attributes of the prediction algorithm [46]. To this end, recent methods have proposed to capture the causal dependence of the outcome on the protected attribute. Before describing these methods, we present a background on causal graphs.

2.2 Causal DAGs

Probabilistic Causal DAG. A causal DAG over a set of variables $V$ is a directed acyclic graph $G$ that captures functional dependencies between these variables. A variable $X_i$ is considered to cause $X_j$ iff $X_i \rightarrow X_j$ in the causal DAG $G$. Each variable in the causal graph $G$ is functionally determined by its parents and some unobserved exogenous variables. The causal graph is used as a compact representation to denote the dependence between different variables. Two variables $X$ and $Y$ are independent when conditioned on $Z$ if $\Pr(Y = y|X = x, Z = z) = \Pr(Y = y|X = x)$ and is denoted by $X \perp Y | Z$. To test this condition, we consider a conditional independence (CI) test [30] that returns if $X$ and $Y$ are independent conditioned on $Z$. An orthogonal line of work has studied different techniques to efficiently test this condition [50]. The joint probability distribution of a set of variables $V$ can be decomposed similar to that of bayesian networks,

$$\Pr(V) = \prod_{X \in V} \Pr(X|\text{Pa}(X)), \quad (1)$$

where $\text{Pa}(X)$ denotes the set of parents of $X$ in the graph $G$.

$d$-separation and Faithfulness

One of the common questions that are answered using causal DAGs is whether $X \perp Y | Z$, i.e. a set of variables $X$ is independent of $Y$, conditioned on $Z$. $d$-separation between three sets of variables $X, Y, Z$, denoted by $X \perp Y | Z$, is a sufficient graphical criterion that syntactically captures observed conditional independencies. $X$ and $Y$ are said to be $d$-separated given $Z$, if all paths between $X$ and $Y$ are blocked by $Z$ (Please refer to the full version [16] for a formal definition of blocking and $d$-separation). Probability distribution of a dataset $D$ is said to be markov compatible [42] if $d$-separation implies CI with respect to the probability distribution $\Pr$. If the converse also holds ($X \perp Y | D, Z \implies X \perp Y | Z$), the probability distribution $\Pr$ is considered faithful to the causal graph $G$ [43]. We assume throughout this work that $\Pr$ is markov compatible and faithful to $G$. As CI and $d$-separation are equivalent under these assumptions, we ignore the sub-script $\Pr$ or $d$ in subsequent discussions. Faithfulness is a standard assumption in causal inference, which ensures that all CI observed in the dataset correspond to $d$-separations in the corresponding causal graph [36, 42, 43]. Graphoid axioms [36, 42, 45] are the popular set of properties that are used to infer conditional independence. We list two axioms that are relevant for this study.

**Lemma 1 (Theorem 1 [36]).** Consider a dataset $D$ with a causal graph $G$, where the data distribution $\Pr$ is faithful to the graph $G$.

1. **Decomposition axiom:** If $A \perp B, C | Z$, then all paths from $A$ to any of $B$ or $C$ are blocked given $Z$. Therefore, any path from $A$ to $B \subseteq B \cup C$ is also blocked given $Z$. Therefore, $A \perp B | Z$. Symmetrically, the same argument proves that $A \perp C | Z$. Therefore, $A \perp B, C | Z$.

2. **Composition axiom:** If $A \perp B | Z$ and $A \perp C | Z$, then $A \perp B, C | Z$.

**Proof.** We use the notion of $d$-separation to prove these results.
We make the following assumptions about the causal graph:

Testing causal fairness. Causal fairness is an interventional definition that is represented using do operators. A straightforward way to test this definition is to leverage a fully specified causal graph (graph structure and equations) to estimate the post-intervention conditional independency of the form. For more insights about the definition of causal fairness, we refer the reader to [46]. Recent work has also studied causal fairness in settings where the protected attribute is unobserved [17].

Lemma 2. If conditional-mutual information between the classifier output $Y'$ and protected attributes $S$ is zero when conditioned on the admissible set $A$, i.e., $I(Y', S|A = a) = 0$ then $Y'$ is causally fair.

3 PROBLEM STATEMENT

In this section, we define the problem of feature selection to ensure interventional fairness and provide high level intuition of the involved challenges.

Consider a dataset $D$ comprising of a disjoint set of two types of features (i) Sensitive $S = \{S_1, \ldots, S_{|S|}\}$ and (ii) Admissible $A = \{A_1, \ldots, A_{|A|}\}$ along with a target variable $Y$. Let $X = \{X_1, \ldots, X_n\}$ denote the collection of $n$ features that are neither admissible nor sensitive and can be added to $D$ by performing a joint between the input dataset and different datasets from different sources or by feature transformation over a subset of the features. Let $V = A \cup S \cup X \cup Y$ denote the exhaustive list of available variables and $Y'$ denote the learnt target variable which has been trained over a subset $T \subseteq V$. Now, we present the definition of causally fair features that can be added to the original dataset.

Definition 2 (Causally Fair Features). For a given set of admissible variables, $A$, we say a collection of features $D = A \cup T$ is causally fair if the bayes optimal predictor $Y'$, trained on $D$ satisfies causal fairness with respect to sensitive attributes $S$.

The goal is to identify the largest subset $T \subseteq V$ such that the variable $Y'$, trained using these variables is fair.

Problem 1. Given a dataset $D = \{A, S, Y\}$ and a collection of variables $X$, identify the largest subset $T \subseteq X$ such that the features $D' = A \cup T$ is causally-fair.

The goal of our problem is to identify all features that can be considered for training a classifier without worsening the fairness of the dataset $D$. Note that $D$ contains only features $S \cup A$ to begin with, so there is no fairness violation as sensitive attributes are allowed to influence $Y'$ through $A$ and $S$ are not used for training. We make the following assumptions about the causal graph:

Assumption 1 (Faithfulness assumption). The causal graph $G$ on $V$ is faithful to the observational distribution on $V$.

This assumption implies that if two variables $A$ and $B$ are connected in the causal graph, the data cannot result in any spurious conditional independency of the form $(A \perp B|C)$ for any subset $C \subset V \setminus \{A, B\}$. Faithfulness assumption is one of the most common assumptions in causality and fairness literature [9, 10, 25, 29, 30, 35, 34, 44, 51, 54, 55], which is crucial to model the input dataset.

Classifier Training. A new variable $Y'$ (prediction variable) is generated by learning a predictor over the selected subset of features $(A \cup T)$, and this predictor is the Bayes optimal classifier with $Pr(Y'|A \cup T)$ derived from the observational distribution $P(V)$. It is equivalent to adding $Y'$ as a new node in the causal graph which is a children of all features that impact the classifier output. We make Assumption 2 to ensure that one would apply the same Bayes optimal predictor that has been learnt from observational data to all datasets irrespective of the intervention. This assumption is crucial to decouple fairness of feature selection from the training procedure and to theoretically analyze the quality of bias removal in feature selection. Training the classifier by performing feature engineering over the identified features satisfies this assumption.

Assumption 2. For evaluating the fairness criterion in Definition 2 using hypothetical interventional distributions, we assume that the mechanism generating $Y'$ is the same as $P(Y'|A \cup T)$ where $P(\cdot)$ is the observational distribution.

Problem intuition: According to the definition of causal fairness, the output distribution of the prediction algorithm should not change when the value of sensitive variables is changed whenever we intervene on $A$. According to do-calculus, intervention on $(A)$ is equivalent to removal of its incoming edges and conditioning on $A$. If all paths from the sensitive variables to the learnt target $Y'$ that go through the variables considered by $f$ are blocked after an intervention on the admissible variables, then the features considered by $f$ are causally-fair. We first show that the maximal set of features that ensure causal fairness is unique.

Lemma 3. Consider two different set of attributes $X_1$ and $X_2$ such that $X_1 \neq X_2$. If a classifier trained on $X_1$ and $X_2$ separately is causally fair, then a classifier trained on $X_1 \cup X_2$ is also causally fair.

Proof. Let $Y_1'$ and $Y_2'$ denote the output variable of the classifier trained on $X_1$ and $X_2$. Let $G'$ denote a modified causal graph where incoming edges of $S$ and $A$ are removed. According to the definition of causal fairness, all paths from the sensitive attributes to $Y_1'$ are blocked in $G'$, i.e., $S \perp Y_1'|G'\cdot A$. Since, $Y_1'$ is a child of attributes in $X_1$, all paths from $S$ to the parents of $Y_1'$ are blocked, i.e., $Pa(Y_1') \perp S|G'\cdot A$. We get the same condition for $X_2$. Let $Y' = f(X_1 \cup X_2)$. We first simplify the LHS of causal fairness definition as follows.

Since, $Y'$ is trained over $X_1$ and $X_2$, $Pa(Y') \subseteq X_1 \cup X_2$. Therefore, $Pa(Y') \perp S|G', implying P_{G'}(Pa(Y') = c|S = s, A = a) = P_{G'}(Pa(Y') = c|A = a)$. Following the same simplification on RHS of Definition 2, we get that $X_1 \cup X_2$ are causally fair. □

Using Lemma 3, we prove that problem 1 has a unique solution.

Lemma 4. Problem 1 has a unique solution $T^*$. 

4 SOLUTION APPROACH

In this section, we first present key properties using an example and generalize them to discuss our algorithm, SeqSel. Section 4.2 analyzes the different steps of Algorithm 1 to guarantee causal fairness of identified features and Theorem 1 presents a close-form expression to identify maximal set of causally-fair features.

4.1 Algorithm

One naïve solution to ensure fairness is to consider only the admissible variables A for prediction and not add any other feature to the dataset D. This would satisfy the fairness condition but achieve poor prediction performance as there may be a variable X ∈ X that is highly correlated with the target variable Y. Another extreme solution is to consider all the variables of X for prediction. This approach would yield high predictive performance but can have arbitrarily poor fairness. We propose SeqSel (Algorithm 1) which considers the collection of variables A, S and X to identify the largest subset of X which when considered along with A ensure causal fairness of the learnt variable Y’. SeqSel algorithm performs CI tests over the observed data without explicit knowledge of the underlying causal graph. We use causal graphs only to illustrate the intuition behind the different components of our algorithm.

Figure 1 presents different example causal graphs, to understand the solution approach and identify CI tests that can be performed without inferring the complete causal graph. These graphs contain sensitive variables S, admissible variables A, target variable Y along with other subsidiary variables X’s.

1. In all three figures, variables like X₁ have unblocked paths from S to X₁ but all these paths are blocked by the admissible set. Therefore, these variables do not capture any new information about the protected variables. In general, such variables can be identified by checking if X₁ is conditionally independent of S given A, i.e. (X₁ ⊥ S|A).

2. Variables like X₃ in Figure 1(b) are independent of the sensitive attributes and can be identified easily by performing CI test between variable X and S.

3. Variable like X₃ in Figure 1(c) is not independent of S₁ but is independent of S₁ given A₂. X₃ ensures causal fairness and can be identified by testing X₃ ⊥ S|A₂.

4. X₂ in Figure 1(b) and 1(c) is not independent of S₁ even with an intervention on A and captures sensitive information. However, X₂ is independent of Y given A.

Algorithm 1 SeqSel

1. Input: Variables A, S, X, Y
2. C₁ ← ∅
3. for X ∈ X do
4. if ∃A ⊆ A such that (X ⊥ S|A) then
5. C₁ ← C₁ ∪ {X}
6. C₂ ← ∅
7. X ← X \ C₁
8. for X ∈ X do
9. if (X ⊥ Y|A∪C₁) then
10. C₂ ← C₂ ∪ {X}
11. return C₁∪C₂

The different types of variables considered in points 1-3 above do not capture any sensitive information after intervening on A or any subset of A. We denote these variables by C₁, identified by testing CI of X with S given any subset of A. Therefore, all paths from S → X → Y are blocked for all these variables. The variables that capture sensitive information but are independent of Y given all the selected features C₁ ∪ A also do not impact the bayes-optimal classifier. This shows that all the variables discussed above ensure causal fairness. Any variable that is not independent of S and Y even after intervening on A is biased and is not safe to be added. X₂ in Figure 1(a) is one such example. Consider a variation in Figure 1(b) by adding an edge X₃ → X₁. Even then X₃ is a valid feature to ensure causal fairness. However, X₁ ∉ S₁|A₁ and therefore, the above mentioned CI conditions do not capture such variables. Specifically, if a variable X has a blocked path from S which forms a collider at the admissible attribute A, then above mentioned CI tests do not capture X in the set of fair features. We discuss this condition more formally in Theorem 1.

Remark 1. In Figure 1(a), X₂ ⊆ X₁ because there does not exist any path from S to X₁ which is unblocked given A.

Remark 2. If C₂ is conditionally independent of Y given A, C₁, it may not contribute towards the predictive power of the Bayes optimal classifier trained on these variables. However, for most practical purposes the classifier trained can leverage C₂ for better prediction.
Algorithm 1 captures these intuitions to perform CI tests in two phases. The first phase (lines 3–5) identifies all variables that do not get affected by sensitive attributes, in the presence of admissible attributes A or any subset of A. All these variables do not capture any extra information about sensitive attributes and are safe to be added to the dataset D. The rest of the variables, X \ C1, capture information about sensitive attributes which can worsen fairness of the dataset. The second phase (lines 6–10) identifies the subset such that the target variable is not affected by their sensitive information in the presence of admissible attributes. We call this algorithm SeqSel as it sequentially performs CI tests to select features.

4.2 Theoretical Analysis
In this section, we show that the variables identified by SeqSel ensure causal fairness. We consider the original causal graph G along with a new variable Y′ that refers to the prediction variable trained using the variables A along with the variables returned by Algorithm 1. We first show that the variables C1 and C2 identified by Algorithm 1 maintain causal fairness. For this analysis, we assume that the target variable Y does not have a child.

Lemma 5. Consider a dataset D with admissible variables A and sensitive S and a collection of variables C1. If ∃A ⊆ A such that (C1 ⊥ S|A) then A ∪ C1 is causally fair.

Proof. Given (C1 ⊥ S|A) for some A ⊆ A, the variable X does not capture any information about the sensitive variables. Hence all paths from S to the target Y that pass through X are blocked. Mathematically, we consider a causal graph along with Y′ and evaluate the distribution under the intervention of A and S as follows.

\[
\Pr[Y' \mid do(S), do(A)] = \sum_{C_1} \Pr[Y' \mid C_1, do(S), do(A)] \Pr[C_1 \mid do(S), do(A)]
\]

Using Lemma 9 from the full version [16]

\[
= \sum_{C_1} \Pr[Y' \mid C_1, do(S), do(A)] \Pr[C_1 \mid do(A)]
\]

Using Lemma 10 from the full version

\[
= \sum_{C_1} \Pr[Y' \mid C_1, do(A)] \Pr[C_1 \mid do(A)] = \Pr[Y' \mid do(A)]
\]

This shows that any intervention on S does not affect the variable Y′, thereby ensuring causal fairness of the considered features.

The following lemma justifies the addition of C2 to the dataset D without affecting its causal fairness.

Lemma 6. Consider a dataset D with admissible variables A and sensitive S, a set of variables C1 satisfying (C1 ⊥ S|A) and a collection of variables C2 with (C2 ⊥ Y|A, C1), if (C2 ⊥ Y|A, C1) then A ∪ C2 ∪ C1 is causally fair.

Proof. We simplify the causal fairness condition as follows:

\[
\Pr[Y' \mid do(S), do(A)] = \sum_{C_1, C_2} \left( \Pr[Y' \mid C_1, C_2, do(S), do(A)] \times \Pr[C_1, C_2 \mid do(S), do(A)] \right)
\]

Using Lemma 10 from the full version [16]

\[
= \sum_{C_1, C_2} \left( \Pr[Y' \mid C_1, C_2, do(A)] \times \Pr[C_1, C_2 \mid do(S), do(A)] \Pr[C_1 \mid do(S), do(A)] \right)
\]

Since Y′ is independent of C2 given A and C1

\[
= \sum_{C_1, C_2} \left( \Pr[Y' \mid C_1, do(A)] \Pr[C_1 \mid do(S), do(A)] \times \Pr[C_1 \mid do(S), do(A)] \right)
\]

This condition shows that A ∪ C1 ∪ C2 ensure causal-fairness.

This shows that the features C1 and C2 ensure causal fairness of the dataset. Using these results, we identify a closed form expression to identify all variables that ensure causal fairness. Note that whenever the trained classifier is not bayes optimal, C1 still ensure causal fairness but the effectiveness of C2 crucially relies on the optimality of the trained classifier.

Theorem 1. Consider a dataset D with admissible variables A, sensitive S, a set of variables X with a target Y. A variable X ∈ X is safe to be added along with T ∪ A, where T ⊆ C1 ∪ C2 ∪ A without violating causal fairness iff (i) (X ⊥ S|A) for some A ⊆ A or (ii) (X ⊥ Y|C′, A), where (C′ ⊥ S|A) or (iii) X is not a descendant of S in G_A, where G_A is same as G with incoming edges of A removed.

Proof. Using Lemma 5 and 6, we can observe that all the variables C1 ∪ C2 such that (C1 ⊥ S|A), where A ⊆ A and (C2 ⊥ Y|C1, A) are safe to be added without worsening the fairness of the dataset. Now consider a variable X, which is not a descendant of S in G_A. All paths from S to X are blocked when we intervene on A as all incoming edges of A are removed. Therefore it is safe to add X without affecting causal fairness of the dataset.

To show the converse, when X ⊥ S|A, ∀A ⊆ A and X ⊥ Y|C′, A and X is a descendant of S in G_A, then we show that X can worsen the fairness. We can observe the following properties about X:

• (S ⊥ X|A) implies there exists a path from S to X that is unblocked given A.
• (X ⊥ Y|A, C′) implies that X is predictive of Y given the features T ⊆ C1 ∪ C2. Therefore, there will be a direct edge from X to the learned variable Y′.

If the paths from S to X are unblocked in G_A then S to X is unblocked when we intervene on A. In this case, the path from S → X → Y′ is unblocked and therefore X is a biased variable that violates causal fairness of the dataset.

SeqSel captures variables that can be identified by performing CI tests. However, the last condition of Theorem 1 requires intervention to identify other variables. Devising a set of CI tests to identify these variables is an interesting question for future work.

Remark 3. PC-algorithm [49], one of the most popular causal discovery techniques learn the causal graph structure from the data. However, these techniques are known to work under specific modelling assumptions of the data and are highly inefficient. The number of CI tests required by such techniques is generally exponential in the number of input attributes.

Complexity: Algorithm 1 tests conditional independence (CI) of each variable with S and Y. In the worst case, it requires O(2^|A| n) CI tests to identify all the variables that do not worsen the fairness of D. In most realistic scenarios, |A| is a small constant, yielding overall complexity of O(n), where n is the number of features. Existing CI testing techniques can generate spurious correlations between independent variables for large values of n. In the next section, we
propose a group testing formulation that reduces this complexity to \(O(\log n)\) tests, thereby improving its accuracy.

### 4.3 Group Testing

Group testing is an old technique that efficiently performs tests on a logarithmic number of groups of items rather than testing each item separately. It has not been used in causal inference to identify independent variables. We use graphoid axioms to show the following two results for any collection of variables \(X\) and \(Z\) justifying the correctness of group testing in our framework.

### Algorithm 2 GrpSel

```plaintext
1: Input: Variables A, S, X, Y
2: \(C_1 \leftarrow \text{first\_phase}\((A, S, X, Y)\)
3: \(C_2 \leftarrow \text{final\_candidates}\((A, S, X, Y, C_1)\)
4: return \(C_1 \cup C_2\)
```

### Algorithm 3 first_phase

```plaintext
1: Input: Variables A, S, X, Y
2: \(C_1 \leftarrow \phi\)
3: if \(\exists A \subseteq A\) such that \((X \perp Z|A)\) then
4: \(C_1 \leftarrow X\)
5: else
6: \(X_1, X_2 \leftarrow \text{random\_partition}(X)\)
7: \(C_1 \leftarrow \text{first\_phase}\((A, S, X_1, Y)\)
8: \(C_1 \leftarrow C_1 \cup \text{first\_phase}\((A, S, X_2, Y)\)
9: return \(C_1\)
```

### Algorithm 4 final_candidates

```plaintext
1: Input: Variables A, S, X, Y, \(C_1\)
2: \(C_2 \leftarrow \phi\)
3: if \((X \perp Y|A, C_1)\) then
4: \(C_2 \leftarrow X\)
5: else
6: \(X_1, X_2 \leftarrow \text{random\_partition}(X)\)
7: \(C_2 \leftarrow \text{final\_candidates}\((A, S, X_1, Y, C_1)\)
8: \(C_2 \leftarrow C_2 \cup \text{final\_candidates}\((A, S, X_2, Y, C_1)\)
9: return \(C_2\)
```

**Lemma 7.** If \(\exists X_i \in X\) such that \(X_i \perp X|Z\) then \((X_i \perp X \setminus \{X_i\}|Z)\) for some variables \(X_1\) and \(Z\).

**Lemma 8.** If \((X_1 \perp X \setminus \{X_1\}|Z)\) then \(\exists X_i \in X \setminus \{X_1\}\) such that \((X_1 \perp X_1|Z)\) for some \(X_i\).

These results yield the following two properties that make Algorithm 1 more efficient:

- If \((X_1 \perp X_2, X_3|Z)\) then \(X_1 \perp X_2|Z\) or \(X_1 \perp X_3|Z\)
- If \((X_1 \perp X_2, X_3|Z)\) then \(X_1 \perp X_2|Z\) and \(X_2 \perp X_3|Z\)

Algorithm 2 presents an improved version of SeqSel1 that uses group testing to remove all the variables that do not satisfy the CI statements shown in Theorem 1. We call this approach GrpSel. GrpSel operates in two phases, aiming to capture variables \(C_1\) and \(C_2\), respectively. The first phase (Algorithm 3) identifies the variables which do not capture any new information about sensitive variables given \(A \subseteq A\). It tests the CI between \(S\) and \(X\) given \(A \subseteq A\). If the variables are conditionally independent, then all the variables \(X\) are identified to maintain causal fairness. On the other hand, if the variables are conditionally dependent, the set \(X\) is partitioned into two equal partitions and first_phase algorithm is called recursively for both the partitions. Algorithm 4 performs the second phase to identify the variables which are independent of the target variable \(Y\) given \(A\) and \(C_1\). This algorithm operates similarly to first_phase with a different CI test.

**Complexity.** Algorithm 3 requires a total of \(2^{|A|}k\log n\) tests to identify all variables \(X\) that satisfy \((S \perp X|A)\), where \(k\) is the number of variables that do not satisfy the condition. The second phase requires \(k'\log k\) tests to identify the variables that satisfy CI with \(Y\) where \(k'\) is the number of variables that do not satisfy the condition. Therefore, GrpSel has better complexity when the total number of biased variables \(k\) is \(o(n/\log n)\).

### 5 EXPERIMENTS

In this section, we empirically evaluate our technique along with baselines on real-world and synthetic datasets. We answer the following research questions. **Q1** Are SeqSel1 and GrpSel1 able to ensure causal fairness of the trained classifier? **Q2** How does the quality of classifier trained using different feature selection algorithms compare in terms of fairness and accuracy? **Q3** Is GrpSel effective in reducing the number of required CI tests?

#### 5.1 Setup

**Datasets.** We consider the following datasets.

- **Medical Expenditure (MEPS)** [2]: predict total number of hospital visits from patient medical information (Healthcare utilization is sometimes used as a proxy for allocating home care). We consider two variations denoted by MEPS(1) and MEPS(2). MEPS(1) considers ‘Arthritis diagnosis’ as admissible and MEPS(2) considers ‘Arthritis diagnosis’ and ‘Mental health’ as admissible. Race is considered sensitive. Contains 7915 training and 3100 test records.
- **German Credit** [1] applications. The account status is considered admissible and person’s age is used as a sensitive attribute. Contains 800 training and 200 test records.
- **Compas** [24]: predict criminal recidivism from features such as the severity of the original crime. The number of prior convictions, age and severity of charge degree are taken as admissible and race as sensitive. Contains 7200 samples.
- **Adult** [3]: predict income of individuals. Gender is considered sensitive and hours per week, occupation, age, education are considered admissible. Contains 48k individuals.
- **Synthetic:** a synthetically constructed dataset where a feature is constructed to be highly correlated to a sensitive feature with probability \(p\). This dataset is used for understanding the effect of number of features and the fraction of noisy features on the complexity of our techniques.

**Baselines.** We consider the following baselines to identify a subset of features for the training task.

1. **A:** uses the variables in the admissible set.
2. **ALL:** uses all features present in the dataset.
3. **Hamlet** [34]: uses heuristics to identify features which do not add value to the data set and can be ignored.
(4) SPred: learn a classifier using an exhaustive set of features to predict the sensitive attribute. Based on feature importance, we remove the highly predictive features.

(5) Capuchin [46]: state-of-the-art in-processing technique that ensures causal fairness by adding or removing tuples.

(6) Fair-PC: learns the causal graph using PC algorithm [49] and uses it to infer features that ensure causal fairness.

**Experiment Setup.** We evaluate accuracy and fairness of the trained classifier on the test set. To evaluate fairness, we measure conditional mutual information (CMI) and absolute odds difference calculated as the difference in false positive rate and true positive rate between the privileged and unprivileged groups. We consider the CMI and group fairness metric as a proxy because zero CMI implies causal fairness which further implies group fairness and can be easily evaluated from observed data [46]. We use RCIT [50] package in R for CI tests and logistic regression as the classifier.

**Figure 2: Classifier fairness and accuracy on MEPS, German, and Compas datasets.**

**Figure 3: (a) Accuracy vs. Abs. odds difference (b) Running time comparison for varying conditioning set size.**

5.2 **Solution Quality**

Figure 2 compares the accuracy of the classifier trained with the features identified by our baselines along with its fairness. All learns the most accurate classifier as compared to all other techniques. However, it achieves the highest odds difference and hence worst fairness with respect to the sensitive attribute of the dataset. A maintains high fairness but achieves quite low accuracy as compared to SeqSel and GrpSel. Hamlet does not identify features that are highly correlated with sensitive attributes and does not improve its fairness. SPred identifies a few features that capture sensitive information but is unable to identify all such features. Hence, it does not improve the fairness of the classifier as compared to GrpSel. Capuchin and FairPC are able to improve fairness as compared to ALL but performs worse than GrpSel and SeqSel. However, accuracy of the learnt classifier is lower for FairPC than Capuchin. SeqSel and GrpSel maintain high fairness with respect to various metrics of fairness without much loss in accuracy. We calculated feature importance of identified attributes and identified that a number of attributes identified in the second phase of our algorithm have non-zero feature importance and contribute towards classifier prediction.

For MEPS and German datasets, GrpSel and SeqSel are able to identify features that mitigate the bias and do not lose much in classifier accuracy. However, all other techniques have higher bias against the protected attribute on Compas. In this case, we observe that the admissible feature is correlated to the sensitive attribute, affecting the fairness of the trained classifier. We empirically swept the p-value threshold from 0.01 to 0.05, and results are stable and do not impact its performance. As an example, the accuracy of the trained classifier was 0.83-0.84 on MEPS and within 0.73-0.76 on German on varying the thresholds. We observed similar behavior on changing the classifier from logistic regression to random forest.

Table 2 compares the conditional mutual information between the learnt variable $Y'$ (according to GrpSel) and target $Y$ with $S$ given $A$.

| Dataset | CMI($S, Y$ | CMI($S, Y'$ | Number of tests |
|---------|-------------|-------------|----------------|
| MEPS(1) | 0.0015 | 0.0015 | 247 |
| MEPS(2) | 0.0014 | 0.0014 | 390 |
| German  | 0.002 | 0.002 | 81 |
| Compas  | 0.0015 | 0.0015 | 257 |
| Adult   | 0.0003 | 0.0003 | 23 |

**Table 2: Conditional Mutual Information [39] and number of CI tests required for each dataset.**

**Model Selection.** We tested these pipelines by training other ML algorithms like random forest and Adaboost classifier. Across all datasets, we observe that SeqSel and GrpSel maintain fairness of the trained classifier while maintaining high accuracy.

5.3 **Synthetic Data**

In this experiment, we tested the causal fairness metric by simulating interventions presented in Definition 2 and compared with ground truth. We evaluate GrpSel and SeqSel on multiple synthetic datasets generated using causal graphs of varied sizes (1000, 3000 and 5000). Across all datasets, we observed that SeqSel and GrpSel identified majority of the variables that ensure causal fairness. However, other baselines were not able to identify all the biased features, thereby leading to biased datasets.

Complexity. The total number of CI tests required by SeqSel and GrpSel are shown in Table 2. GrpSel requires fewer tests than SeqSel across all datasets. Since all these datasets contain fewer

---

3 Some mutual information values were slightly negative and were truncated to 0 as suggested by Mukherjee et al. [39].
than 1000 features, the improvement is not very significant. To understand the difference in complexity of the two techniques, we perform an extensive simulation study by varying the total number of features and the fraction of biased variables.

Figure 5 compares the total number of CI tests required to identify variables that ensure causal fairness. With the increase in total number of features (n), the number of tests required by SeqSel grows linearly. However, the growth of GrpSel is sub-linear and requires fewer tests than SeqSel for larger n. This result is coherent with our theoretical analysis of $O(n)$ tests for SeqSel and $O(k \log n)$ for GrpSel, where $k$ is the number of biased variables and $n$ is the total number of features in the dataset.

Effect of $p$. Figure 4 compares GrpSel and SeqSel as a function of the total fraction of biased variables in the dataset. SeqSel’s complexity is driven by the total number of features irrespective of the number of biased features. However, the tests required by GrpSel are dependent linearly on $p$. This experiment confirms the benefit of using group testing when the total number of biased variables are fewer than $(\log n)/n$.

Advantages of Group-testing We now evaluate the benefits of using group-testing based technique for feature selection. We generated a synthetic dataset containing 1000 records and increased the number of features (denoted by $t$) from 100 to 1000 in increments of 100. We tested the correctness of GrpSel’s output with the ground-truth calculated from the causal graph. We observed that around 5 attributes that are independent of $S$ are dropped by SeqSel when $t = 500$. The spuriousness increases to $\approx 47$ features when $t = 1000$. On the other hand, GrpSel did not return any spurious correlation for $t \leq 900$ features and returned less than 5 spurious features when $t = 1000$. This experiment demonstrates that group-testing can reduce the chances of getting a spurious output.

5.4 Robustness

In this experiment, we changed the test data by modifying the effect of sensitive attribute on the target variable through specific attributes (by changing edge weights of the causal graph). This data distribution shift did not affect the performance of GrpSel or SeqSel and both techniques achieved 0 absolute odds difference. In contrast, prior pre-processing techniques led to an increase in absolute odds difference of upto 15%. The evaluation demonstrated the weakness of pre-processing techniques to generalize to settings beyond the data distribution of the repaired training dataset. Prior work has referred to it as over-fitting with respect to fairness [12].

Running time. Figure 3(b) compares the running time of CI test run using RCIT package for varying size of the conditioning set. This experiment shows that the running time increases linearly with increasing set size but the gradient is very slow. For example, the running time for the adult dataset increases from 8 sec to less than 10 sec when the conditioning set size increases from 1 to 256. Therefore, performing a CI test with groups of features is effective.

Among all techniques, we observe that GrpSel and SeqSel execute within 10 minutes for all real-world datasets, and it takes around 1 minute to train a classifier. Therefore, our techniques learn a fair classifier in less than 11 minutes across all datasets.

6 RELATED WORK

To the best of our knowledge, there is very little related work on discrimination-aware or fair feature selection. One of the recent papers on feature construction and exploration [12] has studied the problem of constructing new features that can help improve prediction without affecting fairness. Gršič-Hlača et al. [18] use human moral judgements of different properties of features (volitionality, reliability, privacy, and relevance) as the starting point for feature selection. Although they cite causal fairness definitions as the basis for feature relevance, they do not use the data to quantify this relevance. Salimi et al. [46] consider causal fairness to change the input data distribution as opposed to identification of a small set of features that ensure causal fairness. Dutta et al. [13] start with the causal fairness perspective as well and also use tools from information theory, but use partial information decomposition to partition the information contained in the features into exempt and non-exempt portions; the goal is not feature subset selection, but gaining insight into different types of discrimination. Nabi and Shpitser [40] considered causal pathways to identify discrimination and then train a fair classifier assuming full knowledge of the underlying causal graph. Zhang et al. [56] consider causal definitions of fairness and devise algorithms that repair the dataset to ensure fairness. Noriega Campero et al. [41] and its followup [4] examine an active feature acquisition paradigm from the perspective of fairness but do not study the causal notion of fairness.

7 CONCLUSION

In this paper, we have tackled the problem of data integration — joining additional features to an initially given dataset — while not introducing additional unwanted bias against protected groups. We have utilized the formalism of causal fairness and do-calculus to develop an algorithm for adding variables that is theoretically-guaranteed not to make fairness worse. We have enhanced this algorithm using group testing to make it more efficient (the first use of group testing in such a setting) and shown its efficacy on several datasets. The extension of our techniques for active learning or online setting are interesting questions for future work.
8 PROOFS
First, we show the following property of do-calculus.

**Lemma 9.** Given a disjoint collection of variables $X$, $Y$ and $Z$ in a causal graph $G$, such that $(X \perp Y | Z')$, where $Z' \subseteq Z$, then $Pr[X\text{do}(Y), do(Z)] = Pr[X\text{do}(Z)]$

**Proof.** Using the third rule of do-calculus (Equation 10, [22]), $Pr[X\text{do}(Y), do(Z)] = Pr[X\text{do}(Z)]$ when $X$ is independent of $Z$ given $Y$ in the graph where incoming edges of $Z$ have been removed. Since, $X \perp Y | Z'$ in $G$ where $Z' \subseteq Z$, removing additional incoming edges will ensure that none of the variables in $Z$ are a collider and conditioning on $Z \setminus Z'$ additionally will still maintain conditional independence. □

**Lemma 10.** Given a dataset $D$ comprising of variables $A \cup S \cup X$, target variable $Y$ and let $Y'$ be the variable learnt using the feature subset $T \cup A$, then $Pr(Y'|do(A), do(S), T) = Pr(Y'|do(A), T)$, where $T \subseteq X$

**Proof.** Based on the assumption about the construction of $Y'$ (Assumption 2), the variable $Y'$ is only dependent on the variables in $A \cup T$ in all environments. Given $A \cup T$, the variable $Y'$ is independent of $S$. The same condition holds even when incoming edges of $A$ are removed. Also, $S$ nodes do not have any incoming edges. Therefore, on applying the third rule of do-calculus, since $Y'$ is independent of $S$ in the modified graph where incoming edges of $A$ and $S$ nodes that are ancestors of $T$ are removed. Therefore, $Pr(Y'|do(A), do(S), T) = Pr(Y'|do(A), T)$ □

8.1 Proof of Lemma 7
We denote conditional mutual information between two variables $X$ and $Y$ given $Z$ as $I(X, Y | Z)$.

**Proof.** Using chain rule, $I(X_1, X_2 | Z) = I(X_1, X_2 | Z) + I(X_1, Z | X_2) ≥ I(X_1, Z | X_2) > 0$ □

8.2 Proof of Lemma 8
**Proof.** $X_1 \perp X_2 \setminus X_1 | Z$ means that the path from $X_1$ to $X_2$ is not blocked. Using assumption 1, that the path to at least one of $X_1 \in X \setminus X_1$ is not blocked. Hence, $\exists i$ such that $X_1 \perp X_2 | Z$. □

8.3 Dataset description
- **Medical Expenditure (MEPS)** 4: This dataset comprises of health assessment features (both physical and mental) along with demographic features. The dataset is used to predict the number of hospital visits.
- **German Credit** 3 dataset from UCI repository contains attributes of various applicants and the goal is to classify them based on credit risk.
- **Compas** 3 was a risk assessment tool used by courts to determine if a defendant should be released or retained. This dataset contains features like age, race, prior conviction, etc.

4https://meps.ahrq.gov/mepsweb/
3https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
3https://github.com/propublica/compas-analysis

Figure 6: Example graph where $X_2$ is not identified as causally fair by GrpSel. We omit other nodes for the sake of clarity.

In addition to the default set of features, we use techniques from [31] to generate new features, constructed by composition of already present features.

**Setup.** We considered the default threshold of $p$-value to be 0.01 and default settings of sklearn’s logistic regression classifier. GrpSel and SeqSel were implemented in R and the classifier training and testing in Python. The code was run on a laptop with 16GB RAM running MAC OS.

9 ADDITIONAL EXPERIMENTS
Our experiments on real-world datasets that compare group fairness metric (absolute odds difference) and conditional mutual information (CMI) correspond two ends of the spectrum. Since causal fairness implies group fairness, Figure 2 provides some evidence that our algorithms can potentially ensure fairness. On the other hand, since GrpSel has low CMI with the target variable given $A$ (Table 2), the CMI of $S$ and $Y'$ will be low even after intervening on $A$. This experiment guarantees the effectiveness of our techniques to ensure causal fairness.

To further analyze the ability of our algorithms to ensure causal fairness, we evaluate GrpSel and SeqSel on multiple synthetic datasets generated using causal graphs of varied sizes (1000, 3000 and 5000) along with the examples shown in Figure 1 a-c.

In this experiment, we validated the effectiveness of SeqSel and GrpSel to identify the variables that ensure causal fairness. Across all datasets, we observed that SeqSel and GrpSel identified all the variables that ensure causal fairness. One of the variables in 1000 node dataset was not detected by our algorithm. We show a small subgraph of this dataset in Figure 6. In this dataset, variable $X_2$ is not identified by GrpSel and SeqSel because $X_2 \perp S_1$ and $X_2 \perp S_1 | A_1$. This is an example scenario where interventional data is required to identify such variables.

We ran an additional experiment to test the robustness of our techniques with respect to distribution shift. In this experiment, we
varied the effect of sensitive attribute on the target variable through specific attributes. This shift in data distribution did not affect the performance of GrpSel or SeqSel but pre-processing techniques like reweighting\(^7\) fail to ensure fairness under the modified distribution.

9.1 d-separation
Two nodes X and Y are d-separated if every path between them (should any exist) is blocked. If even one path between X and Y is unblocked, X and Y are d-connected. More formally,

**Definition 3 (d-separation).** A path \( p \) is blocked by a set of nodes \( Z \) if and only if

1. \( p \) contains a chain of nodes \( A \rightarrow B \rightarrow C \) or a fork \( A \leftarrow B \rightarrow C \) such that the middle node \( B \) is in \( Z \) (i.e., \( B \) is conditioned on), or
2. \( p \) contains a collider \( A \rightarrow B \leftarrow C \) such that the collision node \( B \) is not in \( Z \), and no descendant of \( B \) is in \( Z \).

If \( Z \) blocks every path between two nodes \( X \) and \( Y \), then \( X \) and \( Y \) are d-separated, conditional on \( Z \), and thus are independent conditional on \( Z \).

\(^7\)https://aif360.mybluemix.net/