entropy

MDPI

Article

Interventional Fairness with Indirect Knowledge of Unobserved Protected Attributes

Sainyam Galhotra [1,*], Karthikeyan Shanmugam [2], Prasanna Sattigeri [2] and Kush R. Varshney [2]

[1] Department of Computer Science, University of Chicago, Chicago, IL 60637, USA
[2] IBM Research, Yorktown Heights, NY 10598, USA; karthikeyan.shanmugam2@ibm.com (K.S.); psattig@us.ibm.com (P.S.); krvarshn@us.ibm.com (K.R.V.)
* Correspondence: sainyam@uchicago.edu

Abstract: The deployment of machine learning (ML) systems in applications with societal impact has motivated the study of fairness for marginalized groups. Often, the protected attribute is absent from the training dataset for legal reasons. However, datasets still contain proxy attributes that capture protected information and can inject unfairness in the ML model. Some deployed systems allow auditors, decision makers, or affected users to report issues or seek recourse by flagging individual samples. In this work, we examined such systems and considered a feedback-based framework where the protected attribute is unavailable and the flagged samples are indirect knowledge. The reported samples were used as guidance to identify the proxy attributes that are causally dependent on the (unknown) protected attribute. We worked under the causal interventional fairness paradigm. Without requiring the underlying structural causal model a priori, we propose an approach that performs conditional independence tests on observed data to identify such proxy attributes. We theoretically proved the optimality of our algorithm, bound its complexity, and complemented it with an empirical evaluation demonstrating its efficacy on various real-world and synthetic datasets.

Keywords: causal fairness; responsible data science

check for updates

Citation: Galhotra, S.; Shanmugam, K.; Sattigeri, P.; Varshney K.R. Interventional Fairness with Indirect Knowledge of Unobserved Protected Attributes. *Entropy* **2021**, *23*, 1571. https://doi.org/10.3390/e23121571

Academic Editors: Kateřina Hlaváčková-Schindler and Sotiris Kotsiantis

Received: 22 September 2021
Accepted: 22 November 2021
Published: 25 November 2021

1. Introduction

Due to the societal impact of automated systems, fairness in supervised learning has been a topic of prime importance. There have been numerous advances in defining fairness in terms of associational and causal effects of protected attributes on the prediction attribute [1–4], thereby mitigating unwanted bias. The majority of these algorithms assume that the protected attribute is accurately specified for the training dataset, which is then used to mitigate unwanted biases by processing the input dataset or modifying the training algorithm (in-processing) or post-processing the output of the prediction algorithm. However, the protected attribute is often unavailable or anonymized for legal reasons [5–7].

The absence of protected attributes from the training dataset does not guarantee fairness of the prediction algorithm. One of the primary reasons for this is the presence of proxy attributes that are causally dependent on the protected attributes. In such settings, a key challenge to ensure fairness is to identify these proxy attributes that may percolate bias into the prediction algorithm and then develop ways to mitigate such biases. Even if the dataset lacks any information about these attributes, software testing by legal auditors, recourse analysis of certain samples [8], or complaints from customers often uncover the presence of bias. In this work, we formalize a framework that leverages such indirect knowledge to identify proxy attributes, which can then help to improve fairness. We motivate this setting with the following example.

Example 1. *Imagine that you are a manager examining a machine learning-powered resume screening app that your software company is starting to use internally [9]. You notice that a candidate named Latanya Sweeney—with an S.M. degree in electrical engineering and computer* *science from MIT and professional experience in minimizing privacy risk—has not been prioritized for your requisition for a staff software engineer to work on a HIPAA-compliant cloud infrastructure project. Suspecting algorithmic bias, you flag Latanya's resume as feedback to the resume app.*

In this example of possible unfairness, neither the app nor the manager had access to any protected attributes such as race and gender for legal reasons [5,6]. The missingness of the protected attribute, however, did not prevent the manager from mentally using proxies for race and gender to flag the prediction. In this case, the name Latanya Sweeney is correlated with black women. If the machine learning model behind the app did have unwanted bias providing systematic disadvantage to black people and/or women, the algorithm must have used proxy attributes (like zip code, projects, or writing style) to reconstruct the information in the protected attributes. However, it is difficult to know what those proxy attributes were; it is usually not as simple as just the name of the individual or their zip code.

In this research, we studied fairness in terms of the causal effect of protected attributes on the prediction output/outcome attribute [1–4] and sought to identify the proxy attributes that are causally dependent on the protected attributes (that we do not know and do not have). A variable X is said to be causally dependent on another attribute X' if $X' \rightarrow X$ in the causal graph, i.e., X is functionally dependent on X' and any manipulation of X' would impact X. However, we needed some extra information to help us on this quest. The information we utilized is precisely the indirect knowledge that we can glean from the flagging of possibly unfair decisions that the manager in our example submitted as feedback. We do not assume that the causal graph is known a priori.

We formalized the feedback-based framework to identify proxy attributes that are causally dependent on the unknown protected attribute. In terms of the causal graph, a proxy attribute is defined as the child of a protected attribute. We proposed efficient polynomial time algorithms that identify various connectivity properties of the causal graph that differ in the input dataset and the samples that are flagged by an auditor (indirect knowledge). It then uses these properties to identify constraints over pairs of input attributes, which are then used to formulate a constraint satisfaction problem (CSP). The solution of the CSP returns the set of proxy attributes.

Contributions. Our primary contributions are as follows.

1. We formalized a novel problem of using indirect signals to identify proxy attributes that are causally dependent on the protected attribute.
2. We identified unique connectivity properties of the causal graph, which are leveraged to develop a suite of efficient polynomial time algorithms that do not require the causal graph as an input. Our proposed techniques use off-the-shelf conditional independence tests to identify these attributes.
3. We proved theoretical guarantees that our algorithm accurately identifies the proxy attributes and runs in polynomial time. We showed that the complexity of our algorithm is linear in the number of attributes for sparse graphs.
4. We performed an end-to-end evaluation of our proposed techniques on various real-world and synthetic datasets. In real-world datasets, we showed that the classifier trained using our methods is fair and maintains high accuracy. On synthetic datasets, we validated the correctness of our algorithm by comparing with the ground truth.

2. Problem Setup

We denote random variables (also known as dataset attributes or features) by uppercase letters like X, S, A and their corresponding sample values in lowercase like x, s, a. Table 1 summarizes the notation.

Table 1. Notation Table.

Symbol	Meaning
S	Unobserved protected attribute
$\mathcal{V}$	Set of attributes (also known as variables of the causal graph)
D	Input dataset containing $\mathcal{V}$ attributes
Y	Prediction attribute
Y'	Classifier output
F	Feedback attribute
D'	Feedback set
$\mathcal{V}' \subseteq \mathcal{V}$	Proxy attributes
$\mathcal{V}_F \subseteq \mathcal{V}$	Parents of F in the causal graph

Causal DAG and interventions A *causal* directed acyclic graph (DAG), G over a set of attributes $\mathcal{V}$ is a DAG that models the functional dependence between attributes in $\mathcal{V}$. Each node X represents an attribute in $\mathcal{V}$ that is functionally determined by its parents $Pa(X)$ in the DAG and some unobserved variables. An intervention to a causal graph is where an attribute X is set to some specific value, say x, and its effect on the distribution of the learned target attribute Y is observed. The do-operator allows this effect to be computed on a causal DAG, denoted $P(Y|\texttt{do}(X = x))$. To compute this value, we assumed that X is determined by a constant $X = x$. This assumption is equivalent to a modified graph with all incoming edges into X removed, and the value of X was set to x.

We assumed that the causal graph G on $\mathcal{V}$ is faithful to the observational distribution on $\mathcal{V}$. This means that if two nodes A and B are connected by an edge in the causal graph, the data cannot result in any incorrect conditional independence of the form $A \perp B \mid C$ for any subset $C \subset \mathcal{V} \setminus \{A, B\}$. It is one of the most common assumptions in the causal discovery literature [1,3,10–19]. We use $\perp$ to denote independence. We denote the edges of the causal graph E as a list of pairs (X_1, X_2) such that either X_1 causes X_2 or vice versa.

Unobserved Protected Attribute Consider a dataset D consisting of attributes $\mathcal{V} = \{X_1, \ldots, X_n\}$ along with a target attribute Y. Let S denote the protected attribute that is not available in the dataset D. S is considered as the common confounder for the set of attributes $\mathcal{V}' \subseteq \mathcal{V}$. This is generally the case in settings where the protected attribute is the root node (has no parent) of the causal graph [3].

Interventional Fairness In this work, we considered the causal interventional fairness [3] paradigm that does not allow the protected attributes to affect the classifier output Y' through any attribute that is not admissible ($\mathcal{A}$). Intuitively, an admissible attribute is the one that is allowed to percolate bias into the training algorithm. In Example 1, attributes like race and gender are considered protected attributes, and user preferences like type of job and expected salary are admissible.

Definition 1 (Causal Interventional Fairness)**.** *For a given set of admissible attributes $\mathcal{A}$, a classifier is considered fair if for any collection of values a of $\mathcal{A}$ and output Y', the following holds: $Pr(Y' = y|\texttt{do}(\mathcal{S}) = s, \texttt{do}(\mathcal{A} = a)) = Pr(Y' = y|\texttt{do}(\mathcal{S}) = s', \texttt{do}(\mathcal{A} = a))$ for all values of $\mathcal{A}$, $\mathcal{S}$ and Y'.*

Intuitively, this definition means that the probability distribution of the classifier output Y' is independent of the protected attributes when we intervene on the admissible attributes. In terms of the causal graph, this holds when all paths from the protected attribute to Y' are blocked by the admissible attributes. For more details about this definition, please refer to [3]. As discussed in the example, the current classifier output Y' does not satisfy this fairness criterion, and we wanted to identify the proxy attributes in order to train a fair classifier.

Feedback Attribute In this problem setup, we assumed that a biased classifier outputs Y' are available and that an auditor inspects a subset of these records to identify biased outcomes. These flagged records are denoted with an extra attribute F, where $F = 1$ denotes an example that was flagged by the auditor. As discussed in Example 1, the auditor processes a subset of the features, say, $\mathcal{V}' \subseteq \mathcal{V}$, to flag a data point. Therefore, F is a function of a subset $\mathcal{V}' \subseteq \mathcal{V}$ and the learned target Y' such that $F = 1$ refers to a biased prediction. In terms of the causal graph, the attributes that were used as a signal to flag the classifier output are parents of F.

Complaint set. In order to define the complaint set, we assumed a subset of the records from marginalized groups are discriminated, and a small subset of these discriminated records are reported as complaints. Therefore, all individuals in the complaint set are assumed to correspond to a specific subset of the marginalized group. The set of complaints are denoted by D', comprising attributes $\mathcal{V}$ for a small subset where $F = 1$. (Note that the complaints D' does not contain all samples that suffer from biased prediction but only the ones that have been flagged.) Therefore, any conditional independence test of the form $A \perp_{D'} B|C$ on the sample D' is equivalent to conditioning on the attribute F along with C, denoted by $(A \perp_D B|C, F)$. Whenever it is clear from context, we ignore the subscript D from the expressions. Unless specified, we always write the expression in terms of $\perp_D$. The operator $\perp_{D'}$ is equivalent to $\perp_D$ with a conditioning on F. Since the feedback $F = 1$ refers to a sample of biased predictions, we assumed that the majority of the samples with $F = 1$ correspond to the members of marginalized or otherwise unprivileged communities.

Assumption 1. *Considering the set of complaints (dataset D' where $F = 1$), the protected attribute $S = s$ is fixed for some records in the marginalized group $S = s$ that have been flagged.*

This assumption is crucial to ensure that the feedback set D' contains indirect information about the marginalized group of individuals. Without this assumption, the set D' cannot be used to relate the complaints with the marginalized group. Note that the set D' does not contain all datapoints that have $S = s$. Therefore, adding a new column that treats all records in feedback set as $S = s$ and all others as $S = s'$ cannot be used as the protected attribute of individuals. Let $\mathcal{V}_F \subseteq \mathcal{V}$ denote the set of attributes that are used by the auditor to flag the datapoint. In terms of the causal graph, F is functionally dependent on F. Since F is a common descendant of all these attributes, any pair of attributes $X_1, X_2 \in \mathcal{V}_F$ cannot be d-separated over D' i.e., $(X_1 \not\perp_{D'} X_2|A) \equiv (X_1 \not\perp_D X_2|A, F), \forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$.

Proxy variables. We defined the proxy variables as the non-admissible set of attributes that are functionally dependent on the unobserved protected attribute and that, therefore, have the maximum causal impact of the protected attribute. Due to the absence of the protected attribute, considering the proxy attributes as protected while employing any prior fairness-aware learning algorithm would guarantee a causally fair classifier. More formally, we claim the following.

Lemma 1. *Consider a causal graph G over a set of attributes $\mathcal{V}$, with unobserved protected attribute S. Let Children of the protected attribute S be denoted by $Ch(S)$. If*

$$Pr(Y'|\mathtt{do}(Ch(S) \setminus \mathcal{A}) = c, \mathtt{do}(\mathcal{A}) = a) = P(Y'|\mathtt{do}(Ch(S) \setminus \mathcal{A}) = c', \mathtt{do}(\mathcal{A}) = a)$$

then Y' is causally fair, i.e., $P(Y'|\mathtt{do}(\mathcal{A}) = a, \mathtt{do}(S) = s) = P(Y'|\mathtt{do}(\mathcal{A}) = a, \mathtt{do}(S) = s')$

Proof. Let $\mathcal{T}$ denote the children of S in the causal graph. If $Pr(Y'|\mathtt{do}(\mathcal{T}) = c, do(\mathcal{A}) = a) = P(Y'|\mathtt{do}(\mathcal{T}) = c', do(\mathcal{A}) = a)$, then all paths from the attributes $\mathcal{T}$ to Y' are blocked when incoming edges of $\mathcal{T}$ and $\mathcal{A}$ are removed from G. In order to show that a classifier that obeys the condition of causal fairness with respect to S, we need to prove the following. After removing all incoming edges of S and $\mathcal{A}$, there should be no directed paths from S

to Y' without a collider (Y' should not be a descendant of S). Since all incoming edges of S have been removed, all directed paths from S to Y' pass through the children $\mathcal{T}$. These paths $S \rightarrow X \rightarrow \ldots \rightarrow Y'$ where $X \in \mathcal{T}$: these paths that contain outgoing edges from $\mathcal{T}$ are all blocked because $Pr(Y'|\texttt{do}(\mathcal{T}) = c, do(\mathcal{A}) = a) = P(Y'|\texttt{do}(\mathcal{T}) = c', (\mathcal{A}) = a)$. This shows that whenever the proxy variables are considered as protected while training a fair classifier, causal fairness of the outcome is guaranteed. □

Note that any superset of the children of S (multi-hop descendants) is a valid set of proxy variables as they may be causally dependent on S. However, $Children(S)$ is the smallest set of attributes that need to be accounted for fair classification. Considering more variables as proxies could affect the overall classification accuracy.

3. Problem Statement and Solution Approach

In this section, we first define the problem statement and give high-level observations about the connectivity properties of the causal graph. We then use these properties to design a simple algorithm, which is then improved by formulating a constraint satisfaction problem. We then improve the efficiency of the algorithm by leveraging the sparsity properties of causal graphs.

Based on the notation we defined in the previous section, we can state the problem of identifying proxy-protected attributes as follows.

Problem 1. *Given a dataset D comprising attributes $\mathcal{V}$ with a classifier output Y' and a biased feedback set D', identify the smallest subset $\mathcal{V}' \subseteq \mathcal{V}$ such that the hidden protected attribute S is a common confounder for the attributes in $\mathcal{V}'$.*

Now let us work towards a solution. Let us first identify the condition under which proxies for the protected attribute can be identified from observational data and develop efficient techniques for the same. Consider a simple toy causal graph example, shown in Figure 1, where only the protected attribute is unobserved. We made a simplistic assumption that only the protected attribute is unobserved for this example. Our technique and theoretical analysis extends to the general case where many other attributes may be unobserved. Note that we have access to the training dataset D containing $\mathcal{V} = \{X_1, X_2, X_3\}$ and a small feedback dataset D', which is equivalent to conditioning $F = 1$. The subset of the data that has $F = 1$ may not overlap with the training data. In this example, the attributes that impact F are $\mathcal{V}_F = \{X_1, X_3\}$, and the proxy attributes are $\mathcal{V}' = \{X_1, X_2\}$. We can see that identifying proxy attributes is an easy task if the causal graph is known. Now, let us look at some of the properties of D and D' that can help in the absence of the causal graph.

1. Consider the attributes X_1 and X_2, which are confounded by the protected attribute S and $(X_1, X_2) \notin E$. Since S is unobserved in the dataset D, X_1 and X_2 cannot be d-separated, i.e., $X_1 \not\perp_D X_2 | A, \forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$. However, the feedback F is equivalent to considering a smaller sub-population (conditioning on S), which breaks the confounding relation between X_1 and X_2. Therefore, $X_1 \perp_D X_2 | F \equiv X_1 \perp_{D'} X_2$. This equation can be easily tested by performing a CI test on the flagged samples.
2. Consider the attributes X_1 and X_3, which are not confounded by the protected attribute S. For such attributes, there exists a subset $A \subseteq \mathcal{V} \setminus \{X_1, X_3\}$ such that $X_1 \perp X_3 | A$. In Figure 1, $A = \phi$. However, $X_1, X_3 \in \mathcal{V}_F$ means that the collider path $X_1 \rightarrow Y' \leftarrow X_3$ gets unblocked given F, implying $X_1 \not\perp_D X_3 | A, F \equiv X_1 \not\perp_{D'} X_3 | A$, $\forall A \subseteq \mathcal{V} \setminus \{X_1, X_3\}$. Therefore, X_1 and X_3 can never be d-separated in the feedback dataset D'.

These observations show that different attributes in the causal graph satisfy different properties based on their membership. We formalized these intuitions for general graphs and proved the following properties for any pair of attributes. Lemma 2 proves the condition in which X_1 and X_2 can be d-separated with respect to D and D', if X_1, X_2 are proxy attributes.

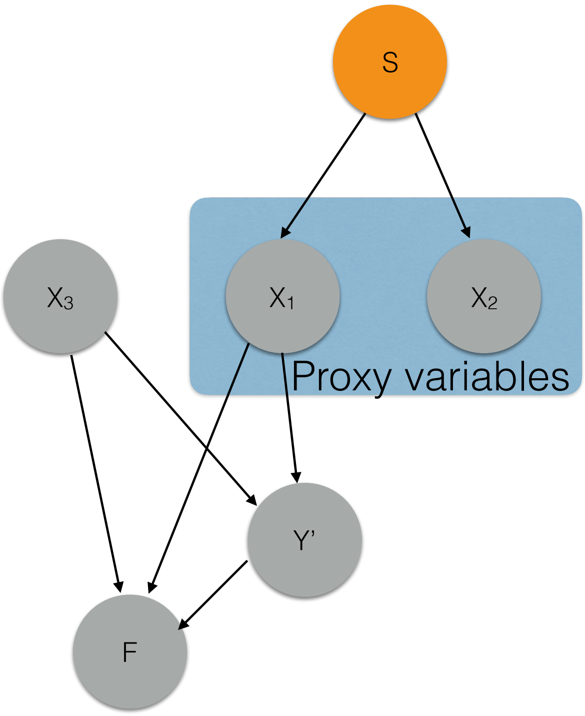


Figure 1. Example dataset where the protected attribute S and the causal graph are unobserved. The attribute Y' denotes the learned target attribute; F is the feedback attribute, which refers to the selection variable for the complaints flagged by an auditor; and X_1 and X_2 are proxy attributes.

Lemma 2. *Consider a pair of attributes X_1 and $X_2 \in \mathcal{V}$ with $(X_1, X_2) \notin E$. $X_1, X_2 \in \mathcal{V}'$, and at least one of X_1 and X_2 does not belong to $\mathcal{V}_F$ iff*

1. $X_1 \not\perp X_2 | A$ *for all* $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ *and*
2. $X_1 \perp X_2 | A, F$ *for some* $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$

Proof. We consider the two sides of the lemma separately. First, let us assume that $(X_1, X_2) \notin E$, $X_1, X_2 \in \mathcal{V}'$ and at least one of X_1 and X_2 do not belong to $\mathcal{V}_F$. This implies the following conditions.

- If $X_1, X_2 \in \mathcal{V}'$, then S is a common confounder for both X_1 and X_2. Therefore, X_1 and X_2 can not be d-separated, implying $(X_1 \not\perp X_2 | A)$ $\forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ because S is not observed.
- If at least one of X_1 and X_2 do not belong to $\mathcal{V}_F$ and $(X_1, X_2) \notin E$, then there exists some A such that X_1 and X_2 are d-separated given A, F. This is because conditioning on the feedback F implies $S = 1$ (conditioning on S), which breaks the confounding relationship between X_1 and X_2.

  For the other direction,

- If $X_1 \perp X_2 | A, F$ for some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$, then both X_1 and X_2 cannot be in V_F and $(X_1, X_2) \notin E$. This is because if $X_1, X_2 \in \mathcal{V}_F$, then $X_1 \not\perp X_2 | A, F$ for any A (by definition of $\mathcal{V}_F$).
- If $X_1 \not\perp X_2 | A$ for all A but $\exists A' \mid X_1 \perp X_2 | A', F$ (we also know that $(X_1, X_2) \notin E$.). Suppose X_1, X_2 are not confounded by S. Conditioning on F and A' blocks all paths from X_1 to X_2. Since conditioning on F does not open any new paths between X_1 and X_2, there will exist A' such that $X_1 \perp X_2 | A'$ if X_1 and X_2 are not confounded by S. This is a contradiction, implying X_1 and X_2 are confounded by S.

□

Lemma 3 proves the properties for X_1 and X_2, whenever both of these attributes are considered by the auditor to flag the datapoint.

Lemma 3. *For a pair of attributes X_1 and $X_2 \in \mathcal{V}$ with $(X_1, X_2) \notin E$, $X_1, X_2 \in \mathcal{V}_F$, and at least one of X_1 and X_2 does not belong to $\mathcal{V}'$ iff*

1. $X_1 \perp X_2 | A$ *for some* $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$
2. $X_1 \not\perp X_2 | A, F$ *for all* $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$

Proof. First, let us assume that $(X_1, X_2) \notin E$, $X_1, X_2 \in \mathcal{V}_F$, and at least one of X_1 and X_2 do not belong to $\mathcal{V}'$.

- If at least one of X_1 and X_2 do not belong to $\mathcal{V}'$ and $(X_1, X_2) \notin E$, then there exists some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ such that X_1 and X_2 are d-separated given A.
- If $X_1, X_2 \in \mathcal{V}_F$, then $X_1 \rightarrow F \leftarrow X_2$ forms a collider path, which is unblocked given F. Therefore, $(X_1 \not\perp X_2 | A, F)\ \forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$

  For the other direction,
- If $X_1 \perp X_2 | A$ for some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$, then both X_1 and X_2 cannot be in $\mathcal{V}'$ and $(X_1, X_2) \notin E$. This is because if $X_1, X_2 \in \mathcal{V}'$, then $X_1 \not\perp X_2 | A$, $\forall A \subseteq \mathcal{V}$ because of an unblocked path $X_1 \leftarrow S \rightarrow X_2$
- If $X_1 \not\perp X_2 | A, F$ for all A but $\exists A$ such that $X_1 \perp X_2 | A$. We also know that $(X_1, X_2) \notin E$. Consider the A for which $X_1 \perp X_2 | A$. In this causal graph, all paths from X_1 to X_2 are blocked but on conditioning F along with A, some path gets unblocked. Since X_1 and X_2 cannot be d-separated when we condition on F, $X_1, X_2 \in \mathcal{V}_F$.

□

For simplicity, we proved these properties for two cases. These properties can be extended for any combination of attributes based on their occurrence in $\mathcal{V}'$ and $\mathcal{V}_F$. Table 2 lists these conditional independence/dependence behavior of all possible combination of attributes X_1 and X_2. For example, the first row shows that if X_1 and $X_2 \in \mathcal{V}_F \cap \mathcal{V}'$, then $X_1 \not\perp X_2 | A$ for all $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$.

3.1. Simple Algorithm

Using the properties listed in Table 2, Algorithm 1 presents the pseudocode of a simple algorithm that identifies proxy-protected attributes. It iterates over all pair of attributes and performs two types of conditional independence tests (one with conditioning on $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ and the other with conditioning on A and F, i.e., with respect to D'). Following Lemma 2, if $\exists A$ such that $X_1 \perp X_2 | F, A$ and $X_1 \not\perp X_2 | A, \forall A$, then X_1 and X_2 are both added to the set $\mathcal{V}'$. Lemma 4 analyzes the conditions when an attribute $X_1 \in \mathcal{V}'$ is correctly identified by Algorithm 1.

Table 2. Conditional independence properties for a pair of attributes $X_1, X_2 \in \mathcal{V}$ such that $(X_1, X_2) \notin E$ where the output of conditional independence tests varies based on the set that X_1, X_2 belong to and vice versa. For example, $X_1, X_2 \in \mathcal{V}' \cap \mathcal{V}_F$ iff $X_1 \not\perp X_2 | A$ and $X_1 \not\perp X_2 | A, F$ for all $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$.

Conditions on X_1, X_2	**Conditioning on D**	**Conditioning on D'**
$X_1, X_2 \in \mathcal{V}' \cap \mathcal{V}_F$	$X_1 \not\perp X_2 \| A$ for all $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$	$X_1 \not\perp X_2 \| A, F$ for all $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$
$X_1, X_2 \in \mathcal{V}_F$ and ($X_1 \notin \mathcal{V}'$ and/or $X_2 \notin \mathcal{V}'$) (Lemma 3) and	$X_1 \perp X_2 \| A$ for some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$	$X_1 \not\perp X_2 \| A, F$ for all $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$
$X_1, X_2 \in \mathcal{V}'$ and ($X_1 \notin \mathcal{V}_F$ and/or $X_2 \notin \mathcal{V}_F$) (Lemma 2)	$X_1 \not\perp X_2 \| A$ for all $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$	$X_1 \perp X_2 \| A, F$ for some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$
$X_1 \in \mathcal{V}' \setminus \mathcal{V}_F$ and $X_2 \in \mathcal{V}_F \setminus \mathcal{V}'$	$X_1 \perp X_2 \| A$ for some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$	$(X_1 \perp X_2 \| A, F)$ for some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$
$X_1 \notin \mathcal{V}' \cup \mathcal{V}_F$	$X_1 \perp X_2 \| A$ for some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$	$(X_1 \perp X_2 \| A, F)$ for some $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$

Algorithm 1 Proxy identification.

```
Input: attributes $\mathcal{V}, F$
$\mathcal{V}' \leftarrow \phi$
for $X_1 \in \mathcal{V} \setminus \mathcal{V}'$ do
    for $X_2 \in \mathcal{V}$ do
        if $\exists A \subseteq \mathcal{V} \setminus \{X_1, X_2\} \mid (X_1 \perp X_2 | F, A)$ then
            if $\forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\} \mid (X_1 \not\perp X_2 | A)$ then
                $\mathcal{V}' \leftarrow \mathcal{V}' \cup \{X_1, X_2\}$
return $\mathcal{V}'$
```

Lemma 4. *An attribute $X \in \mathcal{V}'$ is correctly identified to belong to $\mathcal{V}'$ if $\exists X' \in \mathcal{V}'$ such that $(X, X') \notin E$ and $|\mathcal{V}_F \cap \{X, X'\}| \leq 1$.*

Proof. Consider an attribute $X \in \mathcal{V}'$, and let $X' \in \mathcal{V}'$ such that $\mathcal{V}_F \cap \{X, X'\} \leq 1$. Therefore, one of X and $X' \notin \mathcal{V}_F$. Using Lemma 2, $X \not\perp X' | A$, $\forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$, and $\exists A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ such that $X \perp X' | A, F$ holds. Therefore, Algorithm 1 correctly identifies X and $X' \in \mathcal{V}'$. □

However, Algorithm 1 has two main drawbacks:

1. In dense graphs, there may exist an attribute $X \in \mathcal{V}'$ such that $\nexists X' \in \mathcal{V}'$ where $(X, X') \notin E$. Such attributes may not be identified by Algorithm 1.
2. The conditional independence test of the form $X_1 \not\perp X_2 | A, \forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ requires us to test the conditional dependence for every subset $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$. This condition requires an exponential number of conditional independence tests.

We now present a constraint satisfaction problem-based formulation that overcomes the first limitation (Section 3.2) and an efficient mechanism to optimize the total number of required conditional independence tests (Section 3.3).

3.2. Constraint Satisfaction Formulation

In this section, we leverage the properties of Table 2 to formulate a constraint satisfaction problem (CSP), which is then solved to identify the membership of the attributes. Let us first define the set of variables for this CSP. For each attribute $X \in \mathcal{V}$, define two binary variables X^F and $X^S \in \{0, 1\}$ such that $X^F = 1$ if $X \in \mathcal{V}_F$ and 0 otherwise. Similarly, $X^S = 1$ if $X \in \mathcal{V}'$ and 0 otherwise. Given a pair of attributes X_1 and X_2, we can perform conditional independence tests as described in Table 2 and introduce one of the following constraints based on their output.

- If $X_1 \not\perp X_2 | A, \forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ and $\exists A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ such that $X_1 \perp X_2 | A, F$, then both X_1 and $X_2 \in \mathcal{V}'$ and at least one of the two attributes does not belong to $\mathcal{V}_F$ (Using Lemma 2). Therefore, $X_1^S = X_2^S = 1$ and $X_1^F + X_2^F \leq 1$.
- If $\exists A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ such that $X_1 \perp X_2 | A$ and $X_1 \not\perp X_2 | A, F$ $\forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$, then both attributes X_1 and X_2 belong to $\mathcal{V}_F$, and at least one of the attributes does not belong to $\mathcal{V}'$ (Using Lemma 3). Therefore, $X_1^F = X_2^F = 1$ and $X_1^S + X_2^S \leq 1$
- If $\exists A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ such that $X_1 \perp X_2 | A$ and $\exists A' \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ such that $X_1 \perp X_2 | A', F$, then X_1 and $X_2 \notin \mathcal{V}' \cap \mathcal{V}_F$. Therefore, $X_1^F + X_1^S + X_2^F + X_2^S \leq 2$.

Using this strategy, we introduced constraints for every pair of attributes $X_1, X_2 \in \mathcal{V}$. The membership of all attributes can be identified by solving this constraint satisfaction problem. To solve this constraint satisfaction problem (containing at most $O(\binom{n}{2})$ constraints), we can use any standard CSP solver [20]. Note that most of the presented constraints are binary, and we can easily implement a polynomial time solver to calculate their membership. An efficient implementation of this instance would be to construct a complete graph over the attributes $\mathcal{V}$ with constraints on nodes and edges. For example, the constraint of the form $X_1^S + X_1^F \leq 1$ is a constraint on the node (as these constraints involve a single attribute), and the ones of the form $X_1^F + X_2^F \leq 1$ refer to edge constraints.

To identify a feasible solution, we iteratively removed the constraints by processing node constraints that fix the values of variables and then propagating their effect on the edge constraints. In this constraint satisfaction formulation, membership of all variables that have a unique value are correctly identified. All other variables that do not have a unique value cannot be classified correctly and are considered as proxy attributes. However, we next show that membership of all attributes are correctly identified for realistic settings (sparse graphs). The membership may not be identified in case a number of attributes have a very high degree (see Lemma 4). As an extreme case, membership of an attribute that is functionally dependent on all other attributes would not be identified by the CSP. However, it is impossible to identify its membership as all attributes are dependent on this high-degree attribute.

The main advantage of this algorithm over Algorithm 1 is that we leveraged properties from Table 2 to identify the membership of an attribute X. If an attribute X is attached to every other attribute $X' \in \mathcal{V}$, then our techniques would not be able to pin-point whether X is a proxy attribute or not. In such cases, it returns three sets of attributes (a) proxy attributes having $X^S = 1$, (b) non-proxy attributes ($X^S = 0$), and (c) undecided attributes (high-degree nodes for which X^S is not uniquely determined). If all the proxy and undecided attributes are not used, the trained classifier is guaranteed to be fair.

3.3. Efficient Implementation

Algorithm 1 and the constraint satisfaction problem rely on conditional independence tests that consider all possible subsets $A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$. Therefore, a naive implementation of Algorithm 1 requires $O(2^{|\mathcal{V}|})$ tests. This may not be feasible for large values of $|\mathcal{V}|$, especially when it has to be performed for all pairs of attributes.

In order to improve the overall complexity, we made the following observation for sparse causal graphs. If there exist two attributes X_1 and $X_2 \notin \mathcal{V}'$ where $(X_1, X_2) \notin E$, then they are not connected to any length-2 collider path (paths of the form $X_1 \rightarrow X' \leftarrow X_2$ for some $X' \in \mathcal{V}$) iff $X_1 \perp X_2 | \mathcal{V} \setminus \{X_1, X_2\}$. This holds because when we condition on all attributes except X_1 and X_2, all paths from X_1 and X_2 are blocked except length-2 collider paths of the form $X_1 \rightarrow X_3 \leftarrow X_2$. Since there are no such paths, it means that the test $X_1 \not\perp X_2 | A, \forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ is equivalent to testing for $X_1 \not\perp X_2 | \mathcal{V} \setminus \{X_1, X_2\}$ for such pairs of attributes. Lemma 5 extends this observation to general scenarios where the number of such length-2 collider paths between a pair of attributes is bounded.

Lemma 5. *Consider a pair X_1 and X_2 such that $(X_1, X_2) \notin E$ and at least one of the two attributes does not belong to $\mathcal{V}'$. The following conditions hold:*

1. *X_1 and X_2 are independent when conditioned on all other attributes ($X_1 \perp X_2 | \mathcal{V} \setminus \{X_1, X_2\}$) iff there does not exist $X' \in \mathcal{V}$ such that $X_1 \rightarrow X' \leftarrow X_2$ form a collider path.*
2. *$\exists \mathcal{V}_1$ such that $X_1 \perp X_2 | \mathcal{V}_1$ where $|\mathcal{V}_1| \geq n - t$ iff the number of attributes in set $\mathcal{V}'$ is less than t, where $\mathcal{V}'$ contains all attributes $X \in \mathcal{V}$ that form a length-2 collider path $X_1 \rightarrow X \leftarrow X_2$ or X is a descendant of some attribute $X' \in \mathcal{V}'$, where X' forms a length-2 collider path.*

Proof of Lemma 5. Consider a pair of attributes X_1 and X_2 such that $(X_1, X_2) \notin E$ and at least one of $X_1, X_2 \notin \mathcal{V}'$. If X_1 and X_2 do not have any length-2 collider path, conditioning on all attributes d-separates X_1 and X_2. This holds because for any collider path of length more than 2 (say $X_1 \rightarrow X_i \ldots \leftarrow X_j \leftarrow X_2$), then both X_i or X_j are conditioned. Similarly for any path with incoming edges into X_1 or X_2 (backdoor paths), the parents of both attributes are also conditioned on. Therefore, $X_1 \perp X_2 | \mathcal{V} \setminus \{X_1, X_2\}$.

If a set of attributes $\mathcal{X}', |\mathcal{X}'| \leq t$ where $\mathcal{X}'$ contains all X such that attributes forming length-2 collider of the form $X_1 \rightarrow X \leftarrow X_2$ or X is a descendant of an attribute $X' \in \mathcal{X}'$. In this case, X_1 and X_2 can be d-separated by conditioning on all attributes except $\mathcal{X}'$ because conditioning on any ancestor of X_1 and X_2 does not open new paths. Similarly, if the collider path has a length greater than 2, then the path is blocked by conditioning on all attributes that are not in $\mathcal{X}'$. For example, if the collider path is length 3, $X_1 \rightarrow X_3 \rightarrow X_4 \leftarrow X_2$, then conditioning on X_3 and X_4 does not open this collider path.

More formally, consider any collider path of length greater than 2, say $X_1 \rightarrow X_i \ldots X_j \leftarrow X_2$. If $X_i, X_j \in \mathcal{X}'$, then all descendants of X_i and X_j also belong to $\mathcal{X}'$. Therefore, this path is blocked. If $X_i \notin \mathcal{X}'$, this path is blocked by conditioning on X_i, and conditioning on X_i does not open any length-2 collider paths because $X_i \notin \mathcal{X}'$. Any > 2 length collider path that is unblocked by conditioning on X_i get blocked by another $X_{j'}$, which is a child of X_1 or X_2 in that path. Therefore, conditioning on $\mathcal{V} \setminus \mathcal{X}'$ does not open any path from X_1 to X_2. □

Algorithm 2 uses this property to optimize the number of conditional independence tests required to calculate the membership of each attribute. It initializes with $t = |\mathcal{V}|$ (line 3) and iteratively decreases t to consider attributes with at most $|\mathcal{V}| - t$ length-2 collider paths. For an iteration t, it considers all subsets of $\mathcal{V}$ of size $n - t$ (denoted by $\mathcal{T}$) as the conditioning set (line 6). Using this conditioning set, it evaluates conditional independence constraints for every pair of attributes $X_1, X_2 \in \mathcal{V}$ (Algorithm 3). These constraints are the same as the ones discussed in Section 3.2. The `SolveCSP` subroutine then solves the CSP with new constraints and removes the attributes from U for which X^S has been uniquely determined (line 9). The procedure stops as soon as the X^S values of all attributes $X \in \mathcal{V}$ have been uniquely identified ($U = \phi$) and returns the subset for which $X^S = \{1\}$.

Algorithm 2 Proxy identification.

```
1:  Input: attributes $\mathcal{V}, F$
2:  $U \leftarrow \mathcal{V}, C \leftarrow \phi$
3:  $X^S, X^F \leftarrow \{0,1\}, \forall X \in \mathcal{V}$
4:  $t \leftarrow |\mathcal{V}|$
5:  while $t \geq 0$ and $U \neq \phi$ do
6:      $\mathcal{T} \leftarrow$ IdentifySubset$(\mathcal{V}, t)$
7:      $C \leftarrow C \cup$ PairwiseConstraints$(\mathcal{V}, \mathcal{T})$
8:      SolveCSP$(\mathcal{V}, C)$
9:      $U \leftarrow \{X : 0, 1 \in X^S, X \in \mathcal{V}\}$
10:     $t \leftarrow t - 1$
11: $\mathcal{V}' \leftarrow \{X : X^S = \{1\}\}$
12: return $\mathcal{V}'$
```

Algorithm 3 `Pairwise constraints`.

```
Input: Attributes $\mathcal{V}, F, \mathcal{T}$
$C \leftarrow \phi$
for $(X_1, X_2) \in \mathcal{V} \times \mathcal{V}$ do
    if $\exists T \in \mathcal{T} \mid X_1 \perp X_2 | T \setminus \{X_1, X_2\}$ and $X_1 \not\perp X_2 | T \setminus \{X_1, X_2\}, F \; \forall T \in \mathcal{T}$ then
        $C \leftarrow C \cup \{X_1^F, X_2^F \leftarrow 1\}$
        $C \leftarrow C \cup \{X_1^S + X_2^S \leq 1\}$
    if $X_1 \not\perp X_2 \mid T \setminus \{X_1, X_2\}$ and $X_1 \perp X_2 \mid T \setminus \{X_1, X_2\}, F$ then
        $C \leftarrow C \cup \{X_1^S, X_2^S \leftarrow 1\}$
        $C \leftarrow C \cup \{X_1^F + X_2^F \leq 1\}$
    if $X_1 \perp X_2 \mid T \setminus \{X_1, X_2\}$ and $X_1 \perp X_2 \mid T \setminus \{X_1, X_2\}, F$ then
        $C \leftarrow C \cup \{X_1^S + X_1^F + X_2^S + X_2^F \leq 2\}$
return $C$
```

`PairwiseConstraints`**.** Algorithm 3 presents the pseudocode for this subroutine. It iterates over pairs of attributes and performs CI tests to identify the corresponding constraint, guided by Table 2.

In order to prove the correctness of Algorithm 2, we argue that it does not introduce any spurious constraints in the CSP optimization. Lemma 6 shows that if a pair X_1 and X_2 have more than α length-2 collider paths, then X_1 and X_2 cannot be d-separated by conditioning on any subset of size more than $n - \alpha$. Since each new constraint introduced by Algorithm 3 requires conditional independence of X_1 and X_2 with respect to some subset on D or D', it does not identify incorrect constraints. We now prove Lemma 6.

Lemma 6. *Consider a pair of attributes X_1 and X_2 such that the total number of length-2 collider paths ($X_1 \rightarrow X \leftarrow X_2$ where $X \in \mathcal{V}'$) is at least α. Any CI test between X_1 and X_2 conditioning on A where $|A| > n - \alpha$ returns $X_1 \not\perp X_2 | A$.*

Proof. If a pair of attributes X_1 and X_2 have more than α length-2 collider paths, then conditioning on any subset of size more than $n - \alpha$ implies conditioning on at least one of the collider nodes. Therefore, $X_1 \not\perp X_2 | A$ whenever $|A| > n - \alpha$. □

3.4. Time Complexity

We now analyze the running time of Algorithm 2 for commonly studied causal graph models. Theorem 1 bounds the total number of CI tests required for a degree-bounded graph, and then we extend our analysis to Erdős-Renyi graphs.

Theorem 1. *For a causal graph where each node $X \in \mathcal{V}$ has a degree less than α and $|\mathcal{V}' \setminus \mathcal{V}_F| > \alpha^2$, Algorithm 2 requires $O(n^2)$ CI tests to identify all proxy attributes.*

Proof. For a node X with degree $< \alpha$, the maximum number of 2-hop neighbors of X is $\leq (\alpha - 1)^2$. This analysis considers all edges as undirected and can be tightened by considering directions and splitting α into incoming and outgoing degrees of each node. Therefore, X can have at most $(\alpha - 1)^2$ length-2 collider paths. This means that if $\mathcal{V}' \setminus \mathcal{V}_F$ contains more than $(\alpha - 1)^2$ two-hop and $\alpha - 1$ one-hop attributes, then $\exists X' \in \mathcal{V}'$ such that X' is at least 2-hops away from X. Since $\alpha^2 > (\alpha - 1)^2 + (\alpha - 1)$, $\exists X' \in \mathcal{V}'$ that satisfies this condition. Such attributes are identified in the CI test $X \perp X' | F, \mathcal{V} \setminus \{X, X'\}$. Therefore, all attributes are correctly identified in 1 test for every pair of attributes. □

Erdős-Renyi Graphs. We consider a randomized generative model for the causal graph construction where each pair of attributes are causally related independently with a probability p. We show that whenever $p < 1/\sqrt{n}$, Algorithm 2 identifies all proxy attributes in $O(n^2)$ running time. Such connectivity models for causal graphs have been widely studied [21]. Lemma 7 bounds the expected number of length-2 collider paths between a pair of attributes X_1 and X_2.

Lemma 7. *Consider a pair of attributes X_1 and X_2 such that $(X_1, X_2) \notin E$. The probability that X_1 and X_2 have a length-2 collider path between them is less than $p^2(n-2)$.*

Proof. Let X_v denote a binary random variable such that $X_v = 1$ if $X_1 \rightarrow X \leftarrow X_2$ forms a collider path for $X \in \mathcal{V}$. The probability that $(X_1, X) \in E$ and $(X, X_2) \in E$ is $p \times p = p^2$. Therefore, $Pr[X_v = 1] = p^2$. □

Using this result, we prove the following complexity of our algorithm.

Theorem 2. *Algorithm 2 identifies the proxy attributes in less than $O(n^2)$ CI tests if $p = o(\sqrt{1/n})$*

Proof. Given a pair of attributes X_1 and X_2, the probability that X_1 and X_2 are within 2-hops from each other is $p^2(n-2) = o(1)$ if $p = o(\sqrt{1/n})$. Therefore, $\forall X \in \mathcal{V}'$, there will exist $X' \in \mathcal{V}'$ such that $(X, X') \notin E$ and the two attributes are more than 2-hops away. Therefore, $X \not\perp X' | A \forall A \subseteq \mathcal{V} \setminus \{X, X'\}$ and $X \perp X' | A, F$ for some $A \subseteq \mathcal{V} \setminus \{X, X'\}$.

This means that all attributes in $\mathcal{V}'$ have been recovered in the first iteration of Algorithm 2. □

3.5. Graphical Lasso-Based Algorithm

In this section, we study a specific class of causal graphs where the structural equations are Gaussian. In this setting, we show that Algorithm 2 can be implemented efficiently using the graphical lasso algorithm.

Graphical lasso [22] is one of the widely studied methods to infer the precision matrix of the underlying causal model in settings where the structural equations are Gaussian. (The precision matrix is the inverse of the covariance matrix; its non-zero values encode the edges in the graph.) Following the properties of Lemma 2, we know that $X_1 \not\perp X_2 | A$, $\forall A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ if $X_1, X_2 \in \mathcal{V}'$. Therefore the precision matrix identified over D would contain (X_1, X_2) as an edge. Similarly, Lemma 2 also shows that $\exists A \subseteq \mathcal{V} \setminus \{X_1, X_2\}$ such that $X_1 \perp X_2 | A, F$ iff $X_1, X_2 \in \mathcal{V}'$. This means that the entry corresponding (X_1, X_2) in the precision matrix will be 0. Using this property, a simple algorithm to identify the proxy attributes is as follows. (a) Step 1: Run graphical lasso on the original dataset D. Let P denote the returned precision matrix. (b) Step 2: Run graphical lass on the dataset D'. Let P' denote the returned precision matrix. (c) Step 3: Calculate the set difference $P \setminus P'$. All attributes with degree more than 0 in $P \setminus P'$ are the proxy attributes. One of the advantages of this technique is that the graphical lasso algorithm is highly efficient, but it is restricted to multivariate Gaussian causal models and does not generalize to general datasets.

4. Experiments

In this section, we evaluate the effectiveness of our techniques to identify proxy attributes that capture protected information such that removing these attributes improves classifier fairness. The protected attributes are hidden from the dataset and are used only to evaluate the fairness of the learned classifier.

4.1. Setup

4.1.1. Datasets

We consider the following real-world datasets.

- *Medical Expenditure* (MEPS) [23]: This dataset is used to predict the total number of hospital visits from patient medical information. Healthcare utilization is sometimes used as a proxy for allocating preventative care management. We consider "arthritis diagnosis" as admissible. Race is considered protected and is hidden for experimentation. The dataset contains 7915 training and 3100 test records.
- *German Credit* [24] dataset contains attributes of various applicants, and the goal was to classify them based on credit risk. The account status is taken as admissible, and whether the person is below the mean age is considered protected. The dataset contains 800 training and 200 test records.
- *Adult* dataset [25] contains demographic information of individuals along with their information on their level of education, occupation, working hours, etc. The task was to predict whether or not the annual income of an individual exceeds 50K. Race was treated as the protected attribute, and education was treated as admissible. The dataset contains around 32K training and 16K test records.

4.1.2. Baselines

Our experimental setup is similar to that of [3], where the input dataset contains admissible attributes (denoted by $\mathcal{A}$), referring to the set of attributes that are allowed to inject bias into the trained classifier. In the implementation of our algorithm, we identified all proxy attributes and trained a new classifier after removing them from the dataset. Due to the small size of $\mathcal{A}$, classifiers trained on $\mathcal{A}$ tend to predict a single class if the training data are not balanced. Therefore, we compared the performance of the trained classifier on both original and balanced data. All algorithms were implemented in Python, and we use Scikit-Learn's logistic regression classifier with default parameters.

Since causal fairness cannot be tested on real datasets, we evaluated the fairness of the classifier in terms of absolute odds difference (AOD) as a proxy. AOD is calculated as the difference in the false-positive rate and the true-positive rate between the privileged and unprivileged/marginalized groups. The set of privileged and unprivileged/marginalized groups are identified according to the sensitive attribute. For example, white individuals are considered privileged in MEPS dataset. The feedback sample is constructed randomly by considering a small sample of unprivileged records that received negative outcomes (less than 100 data points). We used the RCIT package [26] for CI testing, and the Glass package [27] for graphical lasso. These packages are in R. Unless specified, we used Algorithm 2 for our experiments. We considered the following baselines. (i) `A` uses the attributes in the admissible set. (ii) `ALL` uses all attributes present in the dataset.

4.2. Solution Quality

Table 3 compares the accuracy and average precision of the trained classifier along with absolute odds difference to measure fairness. Among all datasets, the accuracy of our approach is similar to `All`, and the fairness is similar to that of `A`. This experiment validates that the removal of proxy attributes from the dataset does not worsen the overall accuracy but helps to improve fairness of the trained classifier. Low average precision (less than 0.60) for `A` shows that it does not learn the target attributes Y and predicts the same label for each datapoint. On the other hand, `All` has high accuracy but is highly unfair. As an example, it has an odds difference of 0.38 on the `Adult` and 0.27 on the `MEPS` dataset.

Table 3. Comparison of accuracy (Acc), average precision (AvgP), and absolute odds difference (AOD).

Dataset	OurApproach			All			$\mathcal{A}$		
	Acc	**AvgP**	**OD**	**Acc**	**AvgP**	**OD**	**Acc**	**AvgP**	**OD**
Adult	0.79	0.78	0.025	0.80	0.75	0.06	0.75	0.47	0.03
Adult-balanced	0.78	0.71	0.068	0.65	0.59	0.38	0.63	0.59	0.40
MEPS	0.85	0.75	0.09	0.86	0.77	0.15	0.83	0.41	0
MEPS-balanced	0.77	0.67	0.25	0.77	0.67	0.27	0.76	0.59	0.05
German	0.74	0.7	0.075	0.79	0.71	0.12	0.72	0.44	0.003
German-balanced	0.70	0.66	0.06	0.72	0.67	0.13	0.6	0.53	0.05

On training a balanced classifier for the `Adult` dataset, our algorithm achieved higher accuracy than `All` and almost a 0 odds difference. On investigating this dataset, we noticed that the identified proxy attributes did not help with prediction, and ignoring those attributes helped with both accuracy and fairness. Some of the attributes used by our technique for classifier training after removing the proxy attributes were education and capital in `Adult` and purpose and age in `German`. In `MEPS`, our approach used diagnostic features like cancer diagnosis and blood pressure for prediction. We observed similar results on changing the training algorithm to random forest and AdaBoost classifier.

In addition to comparing the odds difference, we considered the causal graph for `Adult` and `German` from the prior literature [2] and used it as a ground truth to test the correctness of our algorithm. Overall, Algorithm 2 identified 95% of the proxy attributes for these datasets. In terms of running time, our presented technique was completed in less than 10 min on all datasets.

4.3. Synthetic Dataset

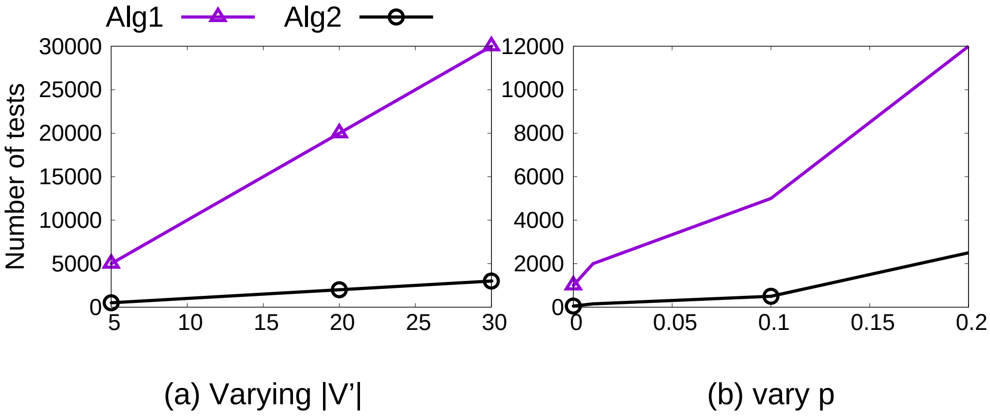
In this experiment, we considered different synthetic datasets and calculated the fraction of proxy attributes identified by Algorithm 2. Since the causal graph was used to generate data, we can verify the correctness of identified proxy attributes for these datasets.

The first experiment considered causal graphs corresponding to `Adult` and `German` where the structural equations of the causal graph followed a multivariate Gaussian distribution. We used the graphical lasso variant of our algorithm for these datasets. Our algorithm identified all proxy attributes on both datasets, and none of the non-proxy attributes were labeled incorrectly.

The second experiment considered random causal graphs containing 20, 40, 60, 80, and 100 attributes consisting of 5 proxy-protected attributes, generated according to the Erdős-Renyi model where every pair of attributes was connected with probability $p = 0.2$. In this case, Algorithm 2 achieved 100% accuracy to identify proxy attributes. To further study the effect of probability p, we considered higher values of $p = 0.5$ and 0.75. In such cases, Algorithm 2 identified 83% of the proxy attributes correctly where the high degree nodes were not identified. These attributes were neither labeled as proxy nor non-proxy.

Complexity Figure 2a shows the effect of an increase in the number of proxy attributes $\mathcal{V}'$ on the number of required conditional independence tests by Algorithms 1 and 2. In this experiment, we considered a causal graph of 50 attributes and varied the number of proxy attributes from 5 to 30. The complexity of both techniques increased linearly with an increase in $|\mathcal{V}'|$, and Algorithm 2 is orders of magnitude better than Algorithm 1. In Figure 2b, we varied the edge formation probability p of the generative model while keeping the size of $\mathcal{V}'$ constant. In this experiment, the total number of tests required increased with increasing p, but Algorithm 1 required much more tests as compared to Algorithm 2. This experiment validated the effectiveness of Algorithm 2 to reduce the number of CI tests required to identify proxy attributes.

In terms of running time, Algorithm 2 ran within 10 minutes for all real-world datasets. In Figure 2, its running time increased proportionally to the increase in the number of CI tests.



(a) Varying |V'|

(b) vary p

Figure 2. Complexity comparison of our techniques for varying dataset sizes.

Effect of feedback set size As an additional experiment, we varied the feedback set size and evaluated the difference in results for real datasets. We observed that our approach ensures fairness whenever the feedback set contains more than 25 samples. An increase in feedback ensures that our technique is stable and ensures fairness across different runs. Whenever the number of samples is small, the behavior of our approach varies. This varied behavior is because our algorithm uses RCIT as a black-box algorithm to test conditional independence, and it returns spurious answers for small sizes of the feedback set.

Overall, this experiment validates that our technique is effective in identifying proxy attributes and mitigating unwanted biases.

5. Related Work

There has been very little work to consider fairness in the absence of protected attributes. Refs. [28,29] consider adversarial reweighting and empirical risk minimization techniques to learn a fair classifier in the absence of demographic information. These techniques do not assume knowledge of protected attributes, but the do not study the causal impact of the unobserved features on the target attribute. Ref. [7] tackles the absence of protected attributes using transfer learning from a different dataset that does have protected attributes. Ref. [30] studies fair class balancing techniques in the absence of protected attributes. There has been some recent interest in studying the effect of noisy attributes on the fairness of classification. Ref. [31] studied the problem of training a fair classifier in the presence of noisy protected attributes. This work does not consider the causal fairness paradigm and does not directly extend to settings where the protected attribute is unobserved. Ref. [32] considered fairness in the presence of noise in the target attribute. These techniques are not directly applicable to our problem setting.

The literature on mitigating unwanted biases considers two types of fairness measures: associational and causal. Associational methods [33–38] have been shown to fail in distinguishing spurious correlations and causal dependence between attributes [3]. Identifying proxy attributes for these techniques is outside the scope of this work. There has been much recent interest in studying causal fairness frameworks [1,10–15,17–19,39] to achieve fairness. Ref. [2] studies the effect of different causal paths from the protected attributes on the target attribute assuming knowledge of the protected attribute and the underlying causal graph. Ref. [3] studies the problem of changing input data distribution in order to ensure interventional fairness. All these techniques require accurate characterization of the protected attribute for all data points. Extending these techniques [2,3] to leverage the information about proxy attributes in the absence of protected attributes is orthogonal to this work and an interesting question for future work.

6. Conclusions

In this work, we formalized a feedback based framework for interventional fairness in settings where the protected attribute is unobserved. Specifically, we examined systems where the auditors, decision makers, or affected individuals report issues in the deployed classifier. These flagged samples that suffered from biased prediction are considered indirect knowledge about the unobserved protected attributes. In this setting, we developed efficient techniques that use conditional independence (CI) testing over the observational data to formulate a constraint satisfaction problem, which identifies the proxy variables. Our techniques partition the variables into different categories based on the output of the performed CI tests. We theoretically proved the correctness of our algorithm, bound its complexity for popular causal graph models, and demonstrated its efficacy on real-world and synthetic datasets.

Author Contributions: Conceptualization, S.G., K.S., P.S. and K.R.V.; methodology, S.G., K.S., P.S. and K.R.V.; experiments, S.G..; validation, S.G., K.S., P.S. and K.R.V.; formal analysis, S.G., K.S., P.S. and K.R.V.; investigation, S.G., K.S., P.S. and K.R.V.; writing—original draft preparation, S.G., K.S., P.S. and K.R.V.; writing—review and editing, S.G., K.S., P.S. and K.R.V.; visualization, S.G. All authors have read and agreed to this version of the manuscript.

Funding: This research did not receive any external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used in the study are openly available, and we provide the relevant citations.

Conflicts of Interest: Galhotra has worked at the University of Massachusetts, Amherst, and the University of Chicago in the last two years. Shanmugam, Sattigeri, and Varshney have worked at IBM Research in the last two years.

References

1. Chiappa, S. Path-specific counterfactual fairness. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, Honolulu, AI, USA, 27 January–2 February 2019; Volume 33, pp. 7801–7808.
2. Nabi, R.; Shpitser, I. Fair inference on outcomes. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
3. Salimi, B.; Rodriguez, L.; Howe, B.; Suciu, D. Interventional fairness: Causal database repair for algorithmic fairness. In Proceedings of the 2019 International Conference on Management of Data, Amsterdam, The Netherlands, 30 June–5 July 2019; pp. 793–810.
4. Galhotra, S.; Shanmugam, K.; Sattigeri, P.; Varshney, K.R. Fair Data Integration. *arXiv* **2020**, arXiv:2006.06053.
5. Ploeg, M.V.; Perrin, E. (Eds.) *Eliminating Health Disparities: Measurement and Data Needs*; National Academies Press: Washington, DC, USA, 2004.
6. Chen, J.; Kallus, N.; Mao, X.; Svacha, G.; Udell, M. Fairness under unawareness: Assessing disparity when protected class is unobserved. In Proceedings of the Conference on Fairness, Accountability, and Transparency, Atlanta, GA, USA, 29–31 January 2019; pp. 339–348.
7. Coston, A.; Ramamurthy, K.N.; Wei, D.; Varshney, K.R.; Speakman, S.; Mustahsan, Z.; Chakraborty, S. Fair Transfer Learning with Missing Protected Attributes. In Proceedings of the AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society, Honolulu, HI, USA, 27–28 January 2019; pp. 91–98.
8. Ustun, B.; Spangher, A.; Liu, Y. Actionable recourse in linear classification. In Proceedings of the Conference on Fairness, Accountability, and Transparency, Atlanta, GA, USA, 29–31 January 2019; pp. 10–19.
9. Hsu, J. Can AI hiring systems be made antiracist? *IEEE Spectr.* **2020**, *57*, 9–11.
10. Xu, D.; Wu, Y.; Yuan, S.; Zhang, L.; Wu, X. Achieving causal fairness through generative adversarial networks. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19), Macao, China, 10–16 August 2019.
11. Kusner, M.; Loftus, J.; Russell, C.; Silva, R. Counterfactual Fairness. In Proceedings of the 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017 ; pp. 4066–4076.
12. Jiang, R.; Pacchiano, A.; Stepleton, T.; Jiang, H.; Chiappa, S. Wasserstein fair classification. *arXiv* **2019**, arXiv:1907.12059.
13. Chiappa, S.; Isaac, W.S. A causal Bayesian networks viewpoint on fairness. In *IFIP International Summer School on Privacy and Identity Management*; Springer: Cham, Switzerland, 2018; pp. 3–20.
14. Kilbertus, N.; Rojas Carulla, M.; Parascandolo, G.; Hardt, M.; Janzing, D.; Schölkopf, B. Avoiding Discrimination through Causal Reasoning. In Proceedings of the 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 656–666.
15. Zhang, J.; Bareinboim, E. Fairness in Decision-Making—The Causal Explanation Formula. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 2037–2045.
16. Zhang, J.; Bareinboim, E. Equality of Opportunity in Classification: A Causal Approach. In Proceedings of the 32nd Conference on Neural Information Processing Systems, Montréal, Canada; 2–8 December 2018; pp. 3671–3681.
17. Khademi, A.; Honavar, V. Algorithmic Bias in Recidivism Prediction: A Causal Perspective. *arXiv* **2019**, arXiv:1911.10640.
18. Khademi, A.; Lee, S.; Foley, D.; Honavar, V. Fairness in algorithmic decision making: An excursion through the lens of causality. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 2907–2914.
19. Russell, C.; Kusner, M.J.; Loftus, J.; Silva, R. When worlds collide: integrating different counterfactual assumptions in fairness. In Proceedings of the 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 6414–6423.
20. Prosser, P. Hybrid algorithms for the constraint satisfaction problem. *Comput. Intell.* **1993**, *9*, 268–299.
21. Tandon, R.; Shanmugam, K.; Ravikumar, P.K.; Dimakis, A.G. On the information theoretic limits of learning Ising models. In Proceedings of the 32nd Conference on Neural Information Processing Systems, Montréal, Canada, 8–13 December 2014; pp. 2303–2311.
22. Friedman, J.; Hastie, T.; Tibshirani, R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **2008**, *9*, 432–441.
23. Medical Expenditure Panel Survey. 2016. Available online: https://meps.ahrq.gov/mepsweb/ (accessed on 1 May 2021).
24. German Data: Uci Machine Learning Repository. 2013. Available online: https://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29 (accessed on 1 May 2021).
25. Adult Data: Uci Machine Learning Repository. 2013. Available online: https://archive.ics.uci.edu/ml/datasets/adult (accessed on 1 May 2021).
26. Strobl, E.V.; Zhang, K.; Visweswaran, S. Approximate kernel-based conditional independence tests for fast non-parametric causal discovery. *J. Causal Inference* **2019**, *7*, 20180017.
27. Friedman, J.; Hastie, T.; Tibshirani, R.; Tibshirani, M.R. Package 'glasso' Available online: https://cran.r-project.org/web/packages/glasso/index.html (accessed on 1 May 2021).
28. Lahoti, P.; Beutel, A.; Chen, J.; Lee, K.; Prost, F.; Thain, N.; Wang, X.; Chi, E.H. Fairness without demographics through adversarially reweighted learning. *arXiv* **2020**, arXiv:2006.13114.
29. Hashimoto, T.; Srivastava, M.; Namkoong, H.; Liang, P. Fairness without demographics in repeated loss minimization. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 1929–1938.

30. Yan, S.; Kao, H.t.; Ferrara, E. Fair Class Balancing: Enhancing Model Fairness without Observing Sensitive Attributes. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual, 19–23 October 2020; pp. 1715–1724.
31. Celis, L.E.; Huang, L.; Vishnoi, N.K. Fair Classification with Noisy Protected Attributes. *arXiv* **2020**, arXiv:2006.04778.
32. Blum, A.; Stangl, K. Recovering from biased data: Can fairness constraints improve accuracy? *arXiv* **2019**, arXiv:1912.01094.
33. Kamishima, T.; Akaho, S.; Asoh, H.; Sakuma, J. Fairness-aware classifier with prejudice remover regularizer. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Bristol, UK, 23–27 September 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 35–50.
34. Zafar, M.B.; Valera, I.; Rogriguez, M.G.; Gummadi, K.P. Fairness constraints: Mechanisms for fair classification. In Proceedings of the Artificial Intelligence and Statistics, Lauderdale, FL, USA, 20–22 April 2017; pp. 962–970.
35. Calders, T.; Verwer, S. Three naive bayes approaches for discrimination-free classification. *Data Min. Knowl. Discov.* **2010**, *21*, 277–292.
36. Celis, L.E.; Huang, L.; Keswani, V.; Vishnoi, N.K. Classification with fairness constraints: A meta-algorithm with provable guarantees. In Proceedings of the Conference on Fairness, Accountability, and Transparency, Atlanta, GA, USA, 29–31 January 2019; pp. 319–328.
37. Hardt, M.; Price, E.; Srebro, N. Equality of opportunity in supervised learning. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 3315–3323.
38. Calmon, F.P.; Wei, D.; Vinzamuri, B.; Ramamurthy, K.N.; Varshney, K.R. Optimized pre-processing for discrimination prevention. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 3995–4004.
39. Xu, Z.; Liu, J.; Cheng, D.; Li, J.; Liu, L.; Wang, K. Assessing the Fairness of Classifiers with Collider Bias. *arXiv*, **2020**, arXiv:2010.03933.