Explaining Algorithmic Fairness Through Fairness-Aware Causal Path Decomposition

Weishen Pan¹, Sen Cui¹, Jiang Bian², Changshui Zhang¹, Fei Wang³

¹Institute for Artificial Intelligence, Tsinghua University (THUAI), State Key Lab of Intelligent Technologies and Systems, Beijing National Research Center for Information Science and Technology (BNRist), Department of Automation, Tsinghua University, P.R.China
²Department of Health Outcomes and Biomedical Informatics, College of Medicine, University of Florida, USA
³Department of Population Health Sciences, Weill Cornell Medicine, USA

{pws15,cuis19}@mails.tsinghua.edu.cn, bianjiang@ufl.edu
zcs@mail.tsinghua.edu.cn, few2001@med.cornell.edu

ABSTRACT
Algorithmic fairness has aroused considerable interests in data mining and machine learning communities recently. So far the existing research has been mostly focusing on the development of quantitative metrics to measure algorithm disparities across different protected groups, and approaches for adjusting the algorithm output to reduce such disparities. In this paper, we propose to study the problem of identification of the source of model disparities. Unlike existing interpretation methods which typically learn feature importance, we consider the causal relationships among feature variables and propose a novel framework to decompose the disparity into the sum of contributions from fairness-aware causal paths, which are paths linking the sensitive attribute and the final predictions, on the graph. We also consider the scenario when the directions on certain edges within those paths cannot be determined. Our framework is also model agnostic and applicable to a variety of quantitative disparity measures. Empirical evaluations on both synthetic and real-world data sets are provided to show that our method can provide precise and comprehensive explanations to the model disparities.

CCS CONCEPTS
• Mathematics of computing → Causal networks; • Computing methodologies → Supervised learning.

KEYWORDS
fairness; explanation; causal graph

1 INTRODUCTION
Machine learning algorithms have been widely applied in a variety of real-world applications including high-stakes scenarios such as loan approvals, criminal justice, healthcare, etc. In these real-world applications, fairness is getting increasing attentions as machine learning algorithms may lead to discrimination against certain disadvantaged sub-populations. This triggers the research on algorithmic fairness, which focus on whether members of specific unprivileged groups are more likely to receive unfavorable decisions made from machine learning algorithms.

One important line of research in the computational fairness community is to develop metrics for measuring group fairness, such as demographic parity [11], equalized opportunity [14], accuracy parity [32], etc., so that the discrepancy among the decisions made in different groups (a.k.a. disparity) are precisely quantified, which can further inspire the development of fair machine learning models that aim to minimize such disparities [3, 22, 32, 33].

Despite the great efforts on fairness quantification and fair model development, one critical issue that has not been studied extensively is the diagnostics of model fairness, i.e., what are the reasons that lead to the model disparity? This information is crucial for understanding the intrinsic model mechanism and provides insights on how to improve model fairness. As an example, under the current pandemic, researchers have found that the racial and ethnic minority groups have been disproportionately affected by COVID-19. African Americans and Hispanics or Latinos are found to be more likely to have positive tests [2, 23], COVID-19 associated hospitalizations and deaths [26], compared with non-Hispanic Whites. In this case, it is crucial to figure out whether such disparity is coming from genetic factors or accessibility to adequate healthcare services, which will imply completely different clinical management plans and public health policies for battling with the pandemic.

In view of this need, recently researchers have leveraged Shapley value based methods [21] to attribute the model disparity as the sum of individual contributions from input features [6, 20], so that we can understand which feature contributes more or less to the model disparity. However, in real-world problems, the mechanism that causes model disparity could be much more complex. Considering the COVID-19 example above, it turns out that one main factor contributing to such disparity is disproportionate access to care for patients with different races and ethnicity, which can be impacted by both economic status [13] and food insecurity [30]. In practice, they correspond to two different causal paths leading to outcome disparity and imply different public health intervention policies.

In this paper, we propose FACTS (which stands for Fairness-Aware Causal paTh decompoSition), a novel framework for algorithm fairness explanation. FACTS decomposes the model disparity as the sum over the contributions of a set of Fairness-Aware Causal paThs (FACT) linking the sensitive attributes with the outcome variable. In this way, our approach can quantify the different causality mechanisms that can lead to the overall model disparity. Specifically, FACTS includes two steps:

• Step 1. FACTs identification, where we propose a method to identify all active paths that link the sensitive attributes and final outcome without colliders given a causal graph constructed on feature variables (which are referred to as FACTs). The graph could be given according to domain knowledge or
learned from the training data. One important consideration here is that frequently the causal directions for certain edges on the graph cannot be determined [7, 25], which makes the graph a Partially Directed Acyclic Graph (PDAG). Our proposed algorithm can effectively identify active paths on PDAGs.

- **Step 2. Disparity attribution through Shapley value decomposition**, where we propose a Shapley value [21] based method to attribute the quantified model disparity value (e.g., according to demographic parity [11]) to identified FACTs, so that the contribution of each FACT can be quantified.

In addition, with the derived attributed disparity on FACTs, we further develop a fair learning approach by selectively removing the FACTs based on their effects on disparity and accuracy through data transformation. Our framework is model agnostic and can be applied to a broad set of popular group-fairness criteria including demographic parity, equalized opportunity, equalized odds, and accuracy parity.

With experiments on both synthetic and real-world datasets, we show that FACTs can accurately quantify individual path contributions to the model disparity. With qualitative analysis on real-world datasets, we demonstrate how our approach can appropriately explain the sources of disparity and successfully make fair adjustments.

## 2 Preliminaries and Related Works

### 2.1 Notations

In this paper, we use capitalized/lower-case letters in italics to represent a variable/value of the variable. We use capitalized/lower-case letters in boldface to represent a variable set/values of the variable set. We use [...] to represent a set without ordering relationship and use [ ] to represent a sequence. \(\pi\) is a function indicating the rank of specific elements in an ordered sequence. For example, for an order pair of features \(\{X_1, X_2\}\), we will have \(\pi(1) < \pi(2)\).

Suppose we are given a dataset including samples characterized by a set of variables \(\{A, X, Y\}\), where \(X = \{X_1, \ldots, X_M\}\) is the set of input feature variables. \(A \in \{0, 1\}\) is the sensitive attribute and \(Y \in \{0, 1\}\) is the outcome. We set \(Y = 1\) as the favored outcome. \(f\) is a trained model and \(\hat{Y}\) is the predicted outcome. \(x = (x_1, x_2, \ldots, x_M)^T\) is a concrete data vector with \(X_1 = x_1, X_2 = x_2, \ldots, X_M = x_M\).

### 2.2 Causal Model

We introduce causal model-related concepts that will be used throughout the paper in this subsection. Our descriptions are based on the definitions in [25, 28].

**Nodes and Edges.** A graph \(G\) consists of a set of nodes (variables) \(\{A, Y, \hat{Y}, X_1, \ldots, X_M\}\) and a set of edges (variable relations). The edges can be either directed (\(\rightarrow\)) or undirected (\(-\)). We call two nodes are adjacent to each other if there is an edge linking them. The collection of nodes adjacent to \(X_i\) is denoted as \(\text{Adj}(X_i)\).

**Paths.** A path \(p\) from \(X_i\) to \(X_j\) in \(G\) is a sequence of nodes where every successive nodes are adjacent in \(G\). A path from \(X_i\) to \(X_j\) in

![Figure 1: An example of a causal graph. Here the prediction \(\hat{Y}\) is obtained by a function \(f\) which takes \(X_1, \ldots, X_4\) as input features.

which all edges are directed from \(X_i\) towards \(X_j\) \((X_i \rightarrow \cdots \rightarrow X_j)\) is a causal path from \(X_i\) to \(X_j\).

**Causal Relationships.** \(X_i\) is a parent of \(X_j\) if there is a directed edge from \(X_i\) to \(X_j\), and \(X_j\) is a child of \(X_i\). \(X_i\) is an ancestor of \(X_j\) if there is a causal path from \(X_i\) to \(X_j\), and \(X_i\) is a descendant of \(X_i\). Following prior research [5, 17], \(\hat{Y}\) is the child of all \(X_i\), which means the predictor \(f\) maps the features variables \(X\) to predicted output \(\hat{Y}\).

**DAGs and PDAGs.** A directed graph is a graph where all edges are directed. A directed graph without directed cycles is a directed acyclic graph (DAG), where directed cycle is formed as a causal path from \(X_i\) to \(X_j\) plus a directed edge \(X_j \rightarrow X_i\). A partially directed graph is a graph where edges can be either directed or undirected. Similarly, a partially directed acyclic graph (PDAG) is a partially directed graph without directed cycles.

**Colliders, Active Nodes and Paths.** Even though there are paths linking \(X_i\) and \(X_j\) in \(G\), \(X_i\) and \(X_j\) are not guaranteed to be dependent. In the example shown in Figure 1, there are paths (e.g., \(A \rightarrow X_3 \leftarrow X_4\)) linking \(A\) and \(X_4\), but \(A\) and \(X_4\) are independent due to fact that they are linked by a collider \(X_3\). To characterize the variable relationships, we introduce the following definitions.

If a path \(p\) contains \(X_i \rightarrow X_k \leftarrow X_j\) as a subpath, then \(X_k\) is a collider on \(p\). For a given set of nodes \(C\) (referred to as the conditioning set), a node \(X_j\) is active relative to \(C\) on a path \(p\) if either: 1) \(X_j\) is a not a collider on \(p\) and not in \(C\); 2) \(X_j\) is a collider, and \(X_j\) or any of its descendants is in \(C\). A path is active relative to \(C\) only when every node on the path is active relative to \(C\). When \(C = \emptyset\), the definition of an active path is degenerated to that there is no collider on the path. When we say \(p\) is an active path, we means \(p\) is an active path relative to \(\emptyset\) in this paper.

As in Figure 1, considering the path \(A \rightarrow X_3 \rightarrow \hat{Y}\) when \(C = \emptyset\), \(X_3\) is an active node since \(X_3\) is not a collider. Thus \(A \rightarrow X_3 \rightarrow \hat{Y}\) is an active path. Similarly, \(A \rightarrow X_5 \leftarrow X_4\) is not an active path because \(X_3\) is a collider. But \(A \rightarrow X_3 \leftarrow X_4\) is an active path relative to \(\{X_3\}\) since \(X_3\) is a collider in the conditioning set \(\{X_3\}\).

**Faithfulness.** The distribution of the variables and \(G\) are faithful to each other means for all \(X_i, X_j, C, X_i\) is conditional independent with \(X_j\) on \(C\) if and only if there exists no active path from \(X_i\) to \(X_j\) with relative to \(C\). Faithfulness is an important and common assumption in the research field of causality, which will be also used in this paper.

### 2.3 Fairness and Disparity

We list some popular algorithmic fairness definitions as follows:

**Demographic Parity (DP) [11].** A prediction \(\hat{Y}\) satisfies demographic parity if \(P(\hat{Y} = 1\mid A = 1) = P(\hat{Y} = 1\mid A = 0)\).
Equalized Odds [14]. A prediction $\hat{Y}$ satisfies equalized opportunity if $P(\hat{Y} = 1|X = y, A = 1) = P(\hat{Y} = 1|X = y, A = 0), \forall y \in \{0, 1\}$.

Equalized Opportunity (EO) [14]. A prediction $\hat{Y}$ satisfies equalized opportunity if $P(\hat{Y} = 1|X = y, A = 1) = P(\hat{Y} = 1|X = y, A = 0)$.

Accuracy Parity [32] A prediction $\hat{Y}$ satisfies accuracy parity if $P(\hat{Y} = Y|A = 1) = P(\hat{Y} = Y|A = 0)$.

In practice, we can take the difference between the two sides of the equalities in the above definitions as a quantification measure for disparity. For example,

$$\Delta_{DP} = P(\hat{Y} = 1|A = 1) - P(\hat{Y} = 1|A = 0)$$

$$\Delta_{EO} = P(\hat{Y} = 1|Y = 1, A = 1) - P(\hat{Y} = 1|Y = 1, A = 0)$$

are two popular algorithm disparity measures used in fairness learning algorithms [22, 27]. Here we follow the work of [6] to use signed difference across groups to show which group is privileged. These fairness definitions are based on the (conditional) independence of the $\hat{Y}$ and the $A$, which can be determined using the causal graph [5]. For example, according to the definitions in Section 2.2, $\hat{Y}$ is independent to $A$ if there is no active path between them, which can obtain demographic parity. In other words, any non-zero $\Delta_{DP}$ comes from the active paths linking $A$ and $\hat{Y}$. In the example in Figure 1, active paths between $A$ and $\hat{Y}$ contain $A \rightarrow X_1 \rightarrow \hat{Y}$, $A \rightarrow X_2 \rightarrow \hat{Y}$ and $A \rightarrow X_3 \rightarrow \hat{Y}$.

2.4 Shapley Values

Shapley values is a popular concept that has been used in model interpretation in recent years [1, 12, 15, 21]. These methods typically decompose the prediction $f(x)$ for a given $x$ as follows.

$$f(x) = \phi_f(0) + \sum_{i=1}^{M} \phi_f(x)(i)$$

where $\phi_f(x)(i)$ is the contribution of feature $X_i$ to $f(x)$. $\phi_f(0) = E[f(X)]$ is the averaged prediction with the expectation over the observed data distribution $P(X)$. $\phi_f(x)(i)$ is referred as the Shapley value of feature $X_i$ for $f(x)$.

In order to calculate $\phi_f(x)(i)$, we firstly assume a sequential order $\pi$ for all variables in $X$, such that $\pi(i)$ corresponds to the rank of $X_i$. The Shapley value of $X_i$ for $f(x)$ with respect to $\pi$ is

$$\phi_f(x)(i) = \frac{1}{M!} \sum_{S \subset X \setminus X_i} \frac{1}{|S|} (f_S(x) - f_{S \setminus \{x_i\}}(x))$$

where $\phi_f(x)(S)$ represents the model’s output on a selected coalition of features $X_S$ with a feature subset $S \subset \{X_1, ..., X_M\}$. $\phi_f(x)(S)$ must satisfy that $\phi_f(x)(\emptyset) = f(x)$ when $S = X$ and $\phi_f(x)(\emptyset) = \phi_f(0)$ when $S = \emptyset$.

2.5 Fairness Explanation

There have been a few studies trying to derive explanations for model fairness, which can be categorized as either feature-based or path-specific explanation.

2.5.1 Feature-based Explanation. Lundberg [20] and Begley et al. [6] leveraged Shapley values defined in Eq.(3) to attribute the feature contributions to $\Delta_{DP}$ in Eq.(1). Specifically, they proposed to use the group difference of $\phi_f(x)(i)$ to quantify the contribution of $X_i$ to $\Delta_{DP}$ as follows:

$$\Phi_f(i) = E_{X|A=i}[\phi_f(x)(i)] - E_{X|A=0}[\phi_f(x)(i)]$$

Begley et al. [6] has also extended this formulation to disparity measured on equalized opportunity as in Eq.(2). However, decomposing model disparity into feature contributions ignores the causal structure of the features.

2.5.2 Path-Specific Explanations. There have been existing research studying fairness and the causal effects on the outcome by flipping the sensitive attribute value. They also studied the causal effects from particular paths, which are called path-specific effects [9, 18, 24, 31]. Intuitively, the path-specific effect can be viewed as quantification of path-specific contribution to model disparity. Existing research has only focused on causal paths (paths in which all edges are directed outwards $A$) so far, this may miss other sources of disparity. For example, in Figure 1, $A \rightarrow X_1 \rightarrow \hat{Y}$ is an active path linking $A$ and $\hat{Y}$ and thus may contribute to model disparity, but it is not a causal path. Therefore, the sum of the path-specific effects considering only causal paths will not amount to the entire model disparity (e.g., measured by Eq.(1)), which leads to incomplete explanations.

In this paper, we propose a novel algorithmic fairness explanation method called Fairness-Aware Causal paTh decompoSition (FACTS), which is introduced in the next section.

3 METHODOLOGY

In this section, we will introduce our framework with $\Delta_{DP}$ as the model disparity metric. We provide generalizations of our framework to other model disparity metrics in the appendix.

Our framework is based on a causal graph $G$, which could be obtained based on domain knowledge or learned from the training data with existing causal discovery algorithms [28]. According to Section 2.3, active paths from $A$ to $\hat{Y}$ are sources of $\Delta_{DP}$. If there is no active path between $A$ and $\hat{Y}$, $\Delta_{DP} = 0$. If $G$ is a DAG, we can identify all active paths based on the definition above and analyze their contributions to disparity. However, in reality $G$ could be a PDAG, where the causal directions on some edges cannot be determined [7, 25]. This makes the problem more challenging.

In this section, we first propose a corresponding concept potential active path to represent the correlation relations between $A$ and
The set of features involved in a fairness associated causal path (FACT) in this paper. Then we propose an algorithm to extract all potential active paths from \( A \) to \( \hat{Y} \). Finally, we decompose \( \Delta_{DP} \) as the sum of the contributions of these paths following the Shapley values strategy.

### 3.1 Potential Active Paths from \( A \) to \( \hat{Y} \)

Potential active paths on a PDAG are defined as follows:

**Potential Active Paths.** A path \( p \) in \( G \) is a potential active path if one of the following properties is satisfied:

1. All edges on \( p \) are directed and \( p \) satisfies the definition of active path in Section 2.2.
2. \( p \) contains undirected edges. If we add arbitrary directions to all undirected edges (adjacent node pairs) on \( p \), the resulting path is active.
3. \( p \) contains undirected edges. Considering all possible directions of undirected edges in \( G \), there exists at least one situation to obtain a DAG \( G' \) which satisfies: 1) corresponding path obtained from \( p \) in \( G' \) is active; 2) the conditional independence relationships among variables encoded in \( G' \) are consistent with those inferred from observational data.

We illustrate the above definition with examples in Figure 2. For path \( A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \) in \( G_1 \), since it is an active path by definition, it is a potential active path as well. As for \( A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \) in \( G_2 \), since no matter what direction of \( A \rightarrow X_1 \) is, the resulting path \( A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \) or \( A \leftarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \) is active. Thus \( A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \) in \( G_2 \) is a potential active path. For \( A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \) in \( G_3 \), suppose the conditional independence relation obtained from data is \( A \perp \!\!\!\!\perp X_1|X_2, A \perp \!\!\!\!\perp X_2|X_1, X_1 \perp \!\!\!\!\perp X_2|A \). Consider the directions of all undirected edges to be \( X_1 \rightarrow X_2 \), the corresponding path \( A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \) in \( G_3 \) under this case is active and the relationships among variables is consistent with the conditional independence relation observed from the data. So \( A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \) is a potential active path.

When \( G \) is DAG, the potential active path is equivalent to the active path. The potential active paths satisfy the following property:

**Proposition 3.1.** If there is no potential active path between two variables on \( G \), then the two variables are independent.

This proposition shows the importance of potential active paths between \( A \) and \( \hat{Y} \) when we consider \( \Delta_{DP} \). Since if there is no potential active path between \( A \) and \( \hat{Y} \), \( A \perp \!\!\!\!\perp \hat{Y} \) and \( \Delta_{DP} = 0 \).

In order to search potential active paths more efficiently, we have the following proposition:

**Proposition 3.2.** If \( p \) is a potential active path in \( G \), then every subpath of \( p \) is also a potential active path.

The proofs of propositions are provided in the appendix. Based on this proposition, we propose an algorithm to search all potential active paths from \( A \) to \( \hat{Y} \) as demonstrated in Algorithm 1.

#### Algorithm 1 Search Potential Active Paths from \( A \) to \( \hat{Y} \)

**Input:** A PDAG \( G \), Dataset \( \{A, X\} \)

**Output:** A set of potential active paths from \( A \) to \( \hat{Y} \): \( P \), A set of features involved in \( P \): \( X (P) \)

**Initialization:**

\[ P_{A→} = \{ \text{Path directly connects } A \text{ and } X_i : X_i \in Adj(A) \}, P = \emptyset \]

\( X (P) = Adj(A) \rightarrow Adj(A) \) is the set of nodes adjacent to \( A \) on \( G \).

(Here \( P_{A→} \) is a temporary set to store the potential active paths from \( A \) during searching)

1. while \( P_{A→} \neq \emptyset \) do
2.   Let \( p \in P_{A→} \)
3.   Remove \( p \) from \( P_{A→} \)
4.   for \( X \leftarrow Adj(p[-1]) \) do \( p[-1] \) means the last node of \( p \)
5.   if \( p' \) is a potential active path by definition then
6.     \( X (P) = X (P) \cup \{ X \} \)
7.   else \( X \leftarrow Adj(p[-1]) \)
8.   end if
9. end for
10. end while

Table 1 summarizes the notations that will be needed for the follow up presentations. As an example in \( G_1 \) of Figure 2, \( P = \{ A \rightarrow X_1 \rightarrow \hat{Y}, A \rightarrow X_2 \rightarrow \hat{Y}, A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y} \} \), \( X (P) = \{ X_1, X_2 \} \), \( X (P) = \{ X_3 \} \). We will have \( X (P) \perp \!\!\!\!\perp A \).

#### Table 1: Important Math Notations

| Notation | Meaning |
|----------|---------|
| \( P \)  | The set of potential active paths from \( A \) to \( \hat{Y} \) |
| \( X (P) \) | The set of features involved in \( P \) |
| \( \hat{X} (P) \) | Set of features not involved in \( P \) |
| \( \phi_f(x_i) (p_i) \) | Contribution of path \( p_i \) to \( f (x) \) |
| \( \Phi(p_i) \) | Contribution of path \( p_i \) to \( \Delta_{DP} \) |

**Ordering Relationships with respect to \( A \).** For two nodes \( X_i \) and \( X_j \) in \( X (P) \), \( X_i \) is defined to be prior to \( X_j \) with respect to \( A \) if:

1. There exist at least one potential active paths from \( A \) to \( \hat{Y} \) that both \( X_i \) and \( X_j \) are on these paths.
2. For all potential active paths from \( A \) to \( \hat{Y} \) containing both \( X_i \) and \( X_j \), \( X_i \) precedes \( X_j \) on them.

In the following, we write such relationship as \( X_i \succ_A X_j \). We also define: \( \forall X_i, A \succ_A X_i, X_i \succ_A \hat{Y} \).

If \( X_j \succ_A X_i \) and \( X_j \) is adjacent to \( X_i \), we call \( X_i \) a direct successor of \( X_j \) with respect to \( A \), and \( X_j \) is a direct predecessor of \( X_i \) with respect to \( A \). The set of all direct successors of \( X_i \) w.r.t.
\(A\) is denoted by \(D_{SA}(X_i)\). The set of all direct predecessors of \(X_i\) w.r.t. \(A\) is denoted by \(D_{PA}(X_i)\). For \(G_1\) in Figure 2, we have \(D_{PA}(X_1) = \{A, D_P(X_2) = \{A, X_1\}\}.\) We will write \(D_{PA}(X_i)\) as \(D_P\) and \(D_{SA}(X_i)\) as \(D_S\) for simplicity in the following.

**Completely Ordered with respect to** \(A\). \(X (P) \in G\) is defined to be completely ordered with respect to \(A\) on \(G\) if \(\forall X_i, X_j \in X (P), X_j \in \text{Adj}(X_i), \) one of them must be the direct successor or predecessor of the other. This can also be written as \(\forall X_i, X_j \in X (P), X_j \in \text{Adj}(X_i),\) we have \(X_j \in D_P \cup D_S\). Considering a special case where \(G\) is a DAG, we have the following proposition.

**Proposition 3.3.** If \(G\) is a DAG and no other node is the parent of \(A\), then \(X(P)\) is completely ordered with respect to \(A\) on \(G\).

Because the condition is sufficient but not necessary for completely ordering w.r.t. \(A\). For example, as in \(G_2\) of Figure 2, even though the edge \(X_1 \rightarrow A\) is undirected, we can still have that \(X(P)\) is completely ordered with respect to \(A\).

### 3.2 FACT Decomposition of Model Disparity

In this subsection, we propose an algorithm to quantitatively attribute the model disparity \(D_{DP}\) to individual FACTs. We first present the algorithm for the case when \(X (P)\) is completely ordered with respect to \(A\) on \(G\). The extension of our algorithm to the scenario when the completely-ordered condition does not hold is introduced in Section 3.3.

Specifically, our algorithm is based on the Shapley values strategy in Section 2.4, and our goal is to decompose \(D_{DP}\) as the sum of contributions from the paths in \(P\) as \(\Delta_{DP} = \sum_{p_i \in P} \Phi_f(p_i)\), and

\[
\Phi_f(p_i) = \mathbb{E}_{X|A=1}[\phi_f(x)\{p_i\}] - \mathbb{E}_{X|A=0}[\phi_f(x)\{p_i\}]
\]

(8)

where \(\phi_f(x)\{p_i\}\) is the Shapley value of FACT \(p_i\). Following Eq.(4), we can define \(\phi_f(x)\{p_i\}\) as

\[
\phi_f(x)\{p_i\} = \frac{\sum_{\pi \in \Pi} v_f(x)\{p_j : \pi(p_j) \leq \pi(p_i)\} - v_f(x)\{p_j : \pi(p_j) < \pi(p_i)\}}{\Pi}
\]

(9)

where \(\pi\) is a permutation function for all FACTs, and \(\Pi\) is the collection of all permutations. \(v_f(x)\{T\}\) is a value function defined on a set of FACTs \(T \subset P\). In order to appropriately define \(v_f(x)\{T\}\), we need to leverage the causality structure encoded in \(P\). In particular, \(P\) can be seen as a system which transfers the information of \(A\) downstreams and finally affects the value of \(Y\). So an intuitive idea is to formulate this system as a calculation process started from \(A\), passing through FACTs and finally get the model prediction \(f(x)\). During the inference process, each \(X_i \in X(P)\) can be estimated as

\[
X_i = g_{X_i}(D_P, \hat{X}(P), E_i)
\]

(10)

where \(g_{X_i}\) is the regression link function, \(D_P\) is the set of predecessors of \(X_i\) in \(P\), \(\hat{X}(P)\) is the set of feature variables that are not in \(X(P)\), \(E_i\) is the random regression error. We assume \(\{E_i\}\) are mutually independent and each \(E_i\) is independent of \(D_P\) and \(X(P)\). Hyvarinen et al. [16] proved that we can always construct such \(\{g_{X_i}\}\) and \(\{E_i\}\). In the following we present the calculation process with a concrete example.

Considering \(G_1\) in Figure 2, we have \(X(P) = \{X_1, X_2\}, X(P) = \{X_3\}, P = \{A \rightarrow X_1 \rightarrow \hat{Y}, A \rightarrow X_2 \rightarrow \hat{Y}, A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y}\}.

According to Eq.(10), we also have \(X_1 = g_{X_1}(A, X_3, E_1)\), \(D_{p_2} = X_1\) and \(X_2 = g_{X_2}(A, X_1, X_3, E_2)\). For calculating \(v_f(x)\{T\}\), we first consider two extreme cases.

- **T = P.** In this case, the actual value of \(A\) is visible to all FACTs, which makes \(v_f(x)\{T\} = f(x)\)
- **T = 0.** In this case, the actual value of \(A\) is visible to none of the FACTs. We can sample \(A\) from its marginal distribution \(a' \sim P(A)\) and calculate \(X_1\) under \(A = a'\), denoted as \(x_i(a')\), then we calculate \(v_f(x)\{T\}\) as

\[
\begin{align*}
    x_1(a') &= g_{X_1}(a', x_3, e_1) \\
    x_2(a') &= g_{X_2}(a', x_1(a'), x_3, e_2) \\
    v_f(x)\{T\} &= \mathbb{E}_{a' \sim P(A)} f(x_1(a'), x_2(a'), x_3)
\end{align*}
\]

(11)

The case of \(T \subset P\) is more complicated. We denote \(x_i(a'|T)\) as the value of \(X_i\) with \(T\). In such process, the values of \(A\) which pass through \(T\) will be set to \(a\), while those passing through \(P \setminus T\) will be set to \(a' \sim P(A)\). Considering the situation \(T = \{A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y}\}\) in the example, we need to transfer the information of \(A = a\) along the path \(A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y}\) but block this information and use a random sample \(a'\) along other paths \(A \rightarrow X_1 \rightarrow \hat{Y}\) and \(A \rightarrow X_2 \rightarrow \hat{Y}\):

\[
\begin{align*}
    x_1(a'|T) &= g_{X_1}(a', x_3, e_1) \\
    x_2(a'|T) &= g_{X_2}(a', x_1(a'), x_3, e_2) \\
    v_f(x)\{T\} &= \mathbb{E}_{a' \sim P(A)} f(x_1(a'|T), x_2(a'|T), x_3)
\end{align*}
\]

(12)

In this way, we can calculate \(\phi_f(x)\{p_i\}\) as in Eq.(9).

In practice, we can choose \(f(x)\) as the probability to predict \(x\) to be positive or the binary decision with a threshold on the probability. In the latter case, the decomposed Shapley values of \(D_{DP}\) on FACTs satisfy the following properties (detailed proofs are provided in the appendix).

1. (Efficiency) \(\sum_{p_i \in P} \Phi_f(p_i) = D_{DP}\)
2. (Linearity) \(\Phi_{a_f + b_f}(i) = a \Phi_{f,P}(i) + b \Phi_f(i)\)
3. (Nullity) \(\Phi_f(p_i) = 0\) when \(v_f(x)\{T \cup \{p_i\}\} = v_f(x)\{T\}\) for all \(x, T \subset P\)

Our path explanations can also be aggregated to generate feature-level explanations. To obtain the contribution of feature \(X_j\) to \(D_{DP}\), we can sum the path contributions for all paths ended with \(\cdots X_j \rightarrow \hat{Y}\).

### 3.3 Algorithm Implementation

When \(G\) is not completely ordered, some of FACTs could be contradictory to each other. Considering \(G_3\) in Figure 2, we have \(P = \{A \rightarrow X_1 \rightarrow \hat{Y}, A \rightarrow X_2 \rightarrow \hat{Y}, A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y}, A \rightarrow X_2 \rightarrow X_1 \rightarrow \hat{Y}\}\). In this case, \(A \rightarrow X_2 \rightarrow X_1 \rightarrow \hat{Y}\) and \(A \rightarrow X_1 \rightarrow X_2 \rightarrow \hat{Y}\) cannot be active simultaneously.

One solution is to consider all orientation possibilities of undirected edges. For example, the direction of \(X_1 \sim X_2\) in \(G_2\) could be either \(X_1 \leftarrow X_2\) or \(X_1 \rightarrow X_2\). We can study each situation respectively and then summarize them. However, this makes the exploration space potentially huge (suppose we have \(N_u\) undirected edges, then we can have \(2^{N_u}\) orientation possibilities to explore).

We propose to solve this problem by grouping the adjacent feature variables which cause the inconsistency problem. In the example of \(G_2\) in Figure 2, if we group \(X_1\) and \(X_2\) as \(\chi_1 = \{X_1, X_2\}\) and
We created a dataset with \( \mathcal{G}_3 \) satisfies the completely-ordered condition. Therefore, we propose to first obtain a partition of \( X(P) \) as \( X(P) = \{X_1, \ldots, X_K\} \) so that \( X(P) \) is completely ordered with respect to \( A \) with these grouped variables \( \{X_k\}_{k=1}^K \). Then we can calculate the contributions of the paths with group-level variables. The concrete algorithm implementation steps are as follows.

\textbf{Step 1: Identify Potential Active Paths from} \( A \) \textbf{t} \( \hat{Y} \). With \( \mathcal{G}_3 \), we find all potential active paths from \( A \) to \( \hat{Y} \) as FACTs with Algorithm 1.

\textbf{Step 2: Generate Groups Completely Ordered With Respect To} \( A \). We group the feature variables so that the grouped variables are completely ordered with respect to \( A \) with Algorithm 2.

\textbf{Step 3: Calculate Path Contributions to} \( \Delta_{DP} \). For each group level variable \( \chi_i \), we learn a prediction link function \( g_{\chi_i} \) and obtain the error term \( E_{\chi_i} \) (both \( g_{\chi_i} \) and \( E_{\chi_i} \) are multi-dimensional, with each dimension corresponding to an individual feature variable in \( \chi_i \)). Finally we calculate the path contribution on the group-level with the procedure in Section 3.2.

\begin{algorithm}[H]
\caption{Generate Groups Completely Ordered w.r.t \( A \)}
\begin{algorithmic}
\State \textbf{Input:} A PDAG \( \mathcal{G}_3 \), \( P, X(P) \)
\State \textbf{Output:} A division of \( X(P) \) into groups \( \chi_1, \ldots, \chi_K \). A set of active paths on group-level \( P' \)
\State \textbf{Initialization:}
\State Create groups \( \{\chi_i\} \), each \( \chi_i \) containing \( X_i \in X(P) \) and path set \( P' \) with all paths in \( P \)
\While {\{\chi_i\} is not completely ordered w.r.t \( A \)}
\State Let \( \chi_i : X_j \in \text{Adj}(\chi_i), X_j \notin Dp(\chi_i) \cup Ds(\chi_i) \)
\State Merge \( \chi_i \) and all \( \chi_j \in \text{Adj}(\chi_i) \setminus Dp(\chi_i) \cup Ds(\chi_i) \), get \( \chi^* \)
\State \text{UPDATE\_RELATION}(\chi^*, P')
\EndWhile
\Procedure{UPDATE\_RELATION}{\chi^*, P'}
\For {\( p \leftarrow P' \)}
\State If \( p \) contains node \( \chi_l : \chi_l \subset \chi^* \) then
\State Replace all \( \chi_l (\chi_l \subset \chi^*) \) by \( \chi^* \) on \( p \)
\State Merge repeated \( \chi^* \) on \( p \)
\EndIf
\EndFor
\State Remove repeated paths in \( P' \)
\EndProcedure
\end{algorithmic}
\end{algorithm}

\subsection{3.4 Fair Learning Through FACT Selection}

With the FACTs based model disparity decomposition approach, we can obtain the qualitative contribution of each FACT to the model disparity. At the same time, if we also consider the model utility, then we can select the paths with high model utility and low model disparity contributions when building the model.

Without the loss of generality, we assume the outcome variable \( Y \in \{0, 1\} \) and the prediction model \( f(x) \) can return the prediction of \( x \) belonging to 1, then the utility of \( f \) can be estimated as

\[ U(f) = \mathbb{E}_{X,Y}[yf(x) + (1 - y)(1 - f(x))] \]  \hspace{1cm} (13)

For a given data sample \( \{x, y\} \), we can calculate the specific model utility of this sample as

\[ U(f(x)) = yf(x) + (1 - y)(1 - f(x)) \]  \hspace{1cm} (14)

We can decompose \( U(f(x)) \) as the contributions of FACTs in \( P \). Similar to Eq.(9), we define \( \Psi_f(x, y) \) as

\[ \Psi_f(x, y)(p_l) = \sum_{\chi_l \in \chi^*} \psi_{\chi_l}(\{p_j : \pi(p_j) \leq \pi(p_l)\}) - \psi_{\chi_l}(\{p_j : \pi(p_j) < \pi(p_l)\}) \]  \hspace{1cm} (15)

where \( f_{\chi_l}(x) = f(x) \) if \( y = 1 \) and \( f_{\chi_l}(x) = 1 - f(x) \) otherwise. In this way, the contribution of \( p_l \) to \( U(f) \) is

\[ \Psi_f(p_l) = \mathbb{E}_{X,Y}[\psi_f(x, y)(p_l)] \]  \hspace{1cm} (16)

Thus \( U(f) \) can be decomposed as

\[ \Psi_f(\emptyset) + \sum_{p_l \in P} \Psi_f(p_l) \]  \hspace{1cm} (17)

With the decomposition of \( U(f) \) and \( \Delta_{DP} \), we can construct an interpretable fair learning algorithm to achieve trade-off between accuracy and fairness through FACT selection. Specifically, our goal is to select a set of paths \( T^* \subset P \) by minimizing the objective:

\[ L(T) = -\sum_{p_l \in T} \Psi_f(p_l) + \lambda \sum_{p_l \in \Phi} \Phi_f(p_l) \]  \hspace{1cm} (18)

We propose a greedy algorithm shown in Algorithm 3 to solve this problem by iteratively removing the edges in \( P \) to get \( T^* \). After we obtain \( T^* \), \( \tilde{U}_f(x)(T^*) \) is used as the new prediction result.

\begin{algorithm}[H]
\caption{Select Paths to Achieve Better Trade-off Between Accuracy and \( \Delta_{DP} \)}
\begin{algorithmic}
\State \textbf{Input:} \( P, \{\Psi_f(p_l) : p_l \in P\}, \{\Phi_f(p_l) : p_l \in P\}, \lambda \)
\State \textbf{Output:} \( T^* \)
\State \textbf{Initialization:} \( T = P \)
\While {\( T \neq \emptyset \)}
\State Let \( p^* = \arg \min_{p \in T} L(T \setminus \{p\}) \)
\State Remove \( p^* \) from \( T \)
\If {\( L(T) < L(T^*) \)}
\State \( T^* = T \)
\EndIf
\EndWhile
\end{algorithmic}
\end{algorithm}

\section{Experiments}

\subsection{4.1 Datasets}

\textbf{Synthetic Data:} We created a dataset with 10 features under a DAG \( \mathcal{G} \). The feature variables and the sensitive attribute \( A \) are randomly connected by directed edges. We generated the data in two different settings: 1) \( S_1 \): the relation between features and outcome is linear; 2) \( S_2 \): the relation between features and outcome is non-linear. The detailed data generation process is described in the appendix.

\textbf{Adult [19]:} The Adult dataset consists of 48,842 samples. The task is to predict whether one’s annual income is greater than 50K. We consider gender (male, female) as sensitive attribute and age, nationality, marital status, level of education, working class and hours per week as feature variables similar to [34]. We set \( Y = 1 \) for Income \( \geq 50K \) and \( A = 1 \) for male.

\textbf{COMPAS [4]:} The dataset contains 6,172 samples and the goal is to predict whether a defendant will recidivate within two years.
Table 2: Normalized root-mean-square error of estimated path contributions.

|        | MLP  | Xgboost |
|--------|------|---------|
| PSE    | Ours |         |
| S1     | 0.09 (± 0.03) | 0.06 (± 0.02) |
| S2     | 0.14 (± 0.03) | 0.10 (± 0.03) |
|         | 0.07 (± 0.02) | 0.07 (± 0.01) |

Table 3: Normalized absolute difference of the summation of path contributions and \( \Delta_{DP} \). If the difference equals 0, the \( \Delta_{DP} \) can be completely explained by the contributions.

|        | MLP  | Xgboost |
|--------|------|---------|
| PSE    | Ours |         |
| S1     | 0.05 (± 0.03) | 0.01 (± 0.01) |
| S2     | 0.09 (± 0.04) | 0.01 (± 0.01) |
| Adult  | 0.23 (± 0.04) | 0.07 (± 0.01) |
| COMPAS | 0.59 (± 0.04) | 0.06 (± 0.03) |
|        | 0.04 (± 0.03) | 0.01 (± 0.01) |
|        | 0.29 (± 0.02) | 0.10 (± 0.01) |
|        | 0.66 (± 0.10) | 0.15 (± 0.07) |

Figure 3: (a): PDAG to explain \( \Delta_{DP} \) on Nutrition, showing top 7 paths contributing to \( \Delta_{DP} \). The meanings of nodes are in Table 4. (b): The set of selected FACTs (\( T' \)) under different values of \( \lambda \) (larger \( \lambda \) means stronger fairness constraints) on Nutrition dataset. Red paths indicates the paths in \( T' \).

or no (\( Y = 1 \) for non-recidivism). Race is the sensitive attribute (\( A = 1 \) for white people) and we choose 7 other attributes including age, gender, number of prior crimes, triple of numbers of juvenile felonies/juvenile misdemeanors/other juvenile convictions, original charge degree.

Nutrition (National Health and Nutrition Examination Survey) [10]: This dataset consists of 14,704 individuals with 20 demographic features and laboratory measurements. The target is to predict 15-year survival. We follow the data preprocessing procedures in [29]. Race (white, non-white) is selected as the sensitive attribute. We set \( Y = 1 \) for 15-year survival and \( A = 1 \) for white people.

4.2 Experimental Setting

We evaluate FACTS from two aspects: 1) path explanations for \( \Delta_{DP} \); 2) fair learning through FACT selection. Here are some general settings to train model \( f \) and learn the FACTs.

Model Training: We train \( f \) using a 70%/30% train/test split. We randomly split data with this ratio and run experiments 5 times. The average result is reported. The hyper-parameters of the model are tuned by cross-validation. When we calculate path explanations for \( \Delta_{DP} \), we implement \( f \) as MultiLayer Perceptron (MLP) or Xgboost [8] and report the results respectively.

Causal Discovery: For the synthetic dataset, we directly use the ground-truth causal graph. For real-world datasets, the ground-truth causal graphs are not available. For Adult and Nutrition datasets, we use the causal graphs built in previous works [9, 24, 29]. For COMPAS dataset, we construct the causal graph with PC algorithm [28], with directions on certain edges restricted and corrected according to domain knowledge. The PDAGs and rules we use to determine the causal directions are shown in the appendix.

4.3 Path Explanations for \( \Delta_{DP} \)

4.3.1 Baselines. Since there is no existing method specifically designed to explain disparity by causal paths. We adopt the following explanation methods as baselines:

- Feature-based Explanation by Shapley Values: we use different Shapley values (Independent Shapley Values(ISV) in [21] and Asymmetric Shapley Values(ASV) in [12]) to calculate the feature-based contribution to disparity with Eq.7.
- Path-specific Explanations (PSE): we calculate the path-specific effect of each potential active path as the estimation of its contribution to disparity, following the calculation in [9].

Since the calculation of path-specific effect requires that the causal relations among \( X(P) \) must be identified in \( \mathcal{G} \). We only report the quantitative results on synthetic, Adult and COMPAS datasets, where this condition is satisfied.

4.3.2 Evaluation Metric. As path-specific effect (PSE) can also provide estimations of path contributions to disparity, we directly compare PSE and FACTS with the following evaluation metrics.

(1) Accuracy: For synthetic dataset where the ground-truth path contributions is available, we evaluate the methods with the

Table 4: Feature explanations on Nutrition. Each column shows the contributions of features to \( \Delta_{DP} \) with ISV/ASV.

| Features                  | ISV  | ASV   |
|---------------------------|------|-------|
| Poverty Index, Food Program (\( \chi_1 \)) | 0.0318 | 0.0553 |
| Blood pressure (\( \chi_2 \))            | 0.0141 | 0.0129 |
| Serum magnesium (\( \chi_3 \))           | 0.0076 | 0.0079 |
| Blood protein (\( \chi_4 \))             | 0.0064 | 0.0075 |
| Sedimentation rate (\( \chi_5 \))        | 0.0099 | 0.0061 |
| White blood cells, Red blood cells (\( \chi_6 \)) | -0.0077 | -0.0082 |

Table 5: Path explanations on Nutrition dataset. The first column shows the contributions of paths to \( \Delta_{DP} \) (\( \Phi_f(p_i) \)), the second column shows the contributions to utility (\( \Psi_f(p_i) \)).

| Paths | \( \Phi_f(p_i) \) | \( \Psi_f(p_i) \) |
|-------|------------------|------------------|
| \( A \to \chi_1 \to \hat{Y} \) | 0.0034 | 0.0039 |
| \( A \to \chi_2 \to \hat{Y} \) | 0.0126 | 0.0009 |
| \( A \to \chi_3 \to \hat{Y} \) | 0.0081 | 0.0006 |
| \( A \to \chi_4 \to \hat{Y} \) | 0.0077 | 0.0032 |
| \( A \to \chi_5 \to \hat{Y} \) | 0.0060 | 0.0006 |
| \( A \to \chi_1 \to \chi_3 \to \hat{Y} \) | 0.0031 | 0.0009 |
| \( A \to \chi_5 \to \hat{Y} \) | -0.0082 | -0.0001 |
normalized root-mean-square error: \[
\sqrt{\frac{\sum_{p_i \in P} (\theta_f(p_i) - \phi_f(p_i))^2}{\sum_{p_i \in P} (\theta_f(p_i))^2}}
\]
where \(\theta_f(p_i)\) is the ground-truth contribution of \(p_i\).

(2) Efficiency: Efficiency is the property that the sum of
the contribution values from all FACTs exactly equals to the
total disparity (Property 1): \(\sum_{p_i \in P} \phi_f(p_i) = \Delta_{DP}\). To show
how close methods come to achieving efficiency, we compute
the absolute difference between summation of path
contributions and disparity:
\[
\frac{|\sum_{p_i \in P} \phi_f(p_i) - \Delta_{DP}|}{\Delta_{DP}}.
\]

4.3.3 Quantitative Results on Estimation of Path Contributions.
The result on accuracy is shown in Table 2. Our FACTs outperforms
PSE on different datasets and prediction models, especially when \(f\)
learns non-linear relation. The result on efficiency is shown in Table
3. We can find that the summation of path explanations obtained
by FACT is closer to the original value of disparity than baseline.

4.4 Fairness Learning through FACTs Selection

4.4.1 Baselines. We choose fair learning baselines as in [6].
- UNFAIR: A baseline model without any fairness constraint.
- AFL [33]: AFL is a fair learning algorithm which trains the
model by maximizing the predictors ability to predict \(Y\) while
minimizing the adversary’s ability to predict \(A\).
- RFL [3]: Agarwal proposes an approach to fair classification,
which optimize a constrained objective to achieve fairness.

For fair comparison, we use the same structure of MLP for all
methods. For our method, we will first train an UNFAIR model \(f\)
and apply our algorithm on \(f\) to get \(T^*\). Then we finetune \(f\) and
use \(\phi_f(x)\) as the new prediction result.

4.4.2 Evaluation. We follow the work of [3] to evaluate the fair
learning algorithms. For each method, we run experiments in a
range of the weights of fairness constraints (such as \(\lambda\) in Eq.(18)).
Then we plot the accuracy-disparity curves for each method.

4.4.3 Results. We firstly compare the path contributions to utility
and disparity from Table 5 and use Figure 3(b) to illustrate how
our algorithm works. \(A \rightarrow \chi_1 \rightarrow \hat{Y}\) is the largest disparity contrib-
utor and makes a major contribution to utility. Other paths also
contribute to \(\Delta_{DP}\), but their contributions to utility are relatively
low. Algorithm 3 can determine the optimal sets of selected paths
under different \(\lambda\) in Eq.(18). In Figure 3(b), we can see that when
\(\lambda = 0\), \(T^*\) contains all FACTs to maintain maximal utility. If \(\lambda\) is
large enough (\(\lambda = 1.0\), all FACTs will be removed to meet this strict
fairness constraint. As \(\lambda = 0.1\), \(T^*\) only contains \(A \rightarrow \chi_1 \rightarrow \hat{Y}\) and
\(A \rightarrow \chi_6 \rightarrow \hat{Y}\). The ratio \(\frac{\hat{Y}(p_i)}{\hat{Y}(p_i)}\) of \(A \rightarrow \chi_1 \rightarrow \hat{Y}\) is greater than
\(\lambda = 0.1\) so it is selected. \(A \rightarrow \chi_6 \rightarrow \hat{Y}\) is also included because
the direction of its \(\phi_f(p_i)\) is opposite to that of \(A \rightarrow \chi_1 \rightarrow \hat{Y}\). It
may be odd to keep two paths with different directions of \(\phi_f(p_i)\)
to obtain a low absolute value of disparity. This reflects the limitation
of treating disparity as a single metric of fairness.

We show the accuracy-fairness trade-off curves for demographic
parity in Figure 5. Our fair learning algorithm with FACTs achieves
comparable results as other fair learning algorithms. It outper-
forms baselines on COMPAS and Nutrition datasets. While on Adult
dataset, most of \(\Delta_{DP}\) comes from a single path (See appendix).
After this path is removed, both \(\Delta_{DP}\) and accuracy decrease obviously.
Our algorithm can be completely explained by which paths are
removed, while the baselines can only obtain black-box models.

5 CONCLUSION

In this work, we propose to a novel framework to explain algorithm-
ic fairness with the causal graph. In particular, we decompose
the disparity into contributions from fairness-aware causal paths
that link the sensitive attribute and model outcome on a causal
graph. We propose an algorithm to identify those paths and calcu-
late their contributions to the disparity. With the path explanations,
we can gain more insight into the inequality of a machine learning
algorithm and propose an interpretable method to achieve better
trade-offs between utility and fairness.

ACKNOWLEDGMENTS

Weishen Pan, Sen Cui, and Changshui Zhang would like to acknowl-
dge the funding by the National Key Research and Development
Program of China (No. 2018AAA0100701) and Beijing Academy of
Artificial Intelligence (BAAI). Fei Wang would like to acknowledge
the support from Amazon Web Service (AWS) Machine Learning
for Research Award and Google Faculty Research Award.
Figure 5: The accuracy-fairness trade-off curves for $\Delta DP$ on various datasets. The upper-left corner (high accuracy, low disparity) is preferred. Curves are generated by searching on a range of fair coefficients (for example, $\lambda$ in Eq.(18)).

REFERENCES

[1] Kjersti Aas, Martin Jullum, and Anders Løland. 2019. Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. arXiv preprint arXiv:1903.10464 (2019).

[2] Ayodeji Adegunsoye, Jazmin Bauer Ventura, and Vladimir M Liarski. 2020. Association of black race with outcomes in COVID–19 disease: a retrospective cohort study. Annals of the American Thoracic Society 17, 10 (2020), 1336–1339.

[3] Álaks Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. 2018. A reduction approach to fair classification. In International Conference on Machine Learning. PMLR, 60–69.

[4] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine bias: There’s software used across the country to predict future criminals. And it’s biased against blacks. ProPublica (2016).

[5] Benjamin R Baer, Daniel E Gilbert, and Martin T Wells. 2019. Fairness criteria through the lens of directed acyclical graphical models. arXiv preprint arXiv:1906.11333 (2019).

[6] Tom Begley, Tobias Schwedes, Christopher Frye, and Ilya Feige. 2020. Explainability for fair machine learning. arXiv preprint arXiv:2010.07789 (2020).

[7] Giorgos Borboudakis and Ioannis Tsamardinos. 2012. Incorporating causal prior knowledge as path-conditions in bayesian networks and maximal ancestral graphs. In Proceedings of the 29th International Conference on International Conference on Machine Learning. 427–434.

[8] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 785–794.

[9] Silvia Chiappa. 2019. Path-specific counterfactual fairness. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33. 7801–7808.

[10] Christine S Cox. 1998. Plan and operation of the NSANES I Epidemiologic Followup Study. 1992. Number 35. National Ctr for Health Statistics.

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

[12] Christopher Frye, Colin Rowat, and Ilya Feige. 2020. Asymmetric Shapley values: incorporating causal knowledge into model-agnostic explainability. Advances in Neural Information Processing Systems 33 (2020).

[13] Elise Gould and Valerie Wilson. 2020. Black workers face two of the most lethal preexisting conditions for coronavirus—Racism and economic inequality. Economic Policy Institute 1 (2020).

[14] Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of Opportunity in Supervised Learning. In Advances in Neural Information Processing Systems. 3315–3323.

[15] Tom Heskes, Evi Sijben, Ioan Gabriel Bucur, and Tom Claassen. 2020. Causal Shapley Values: Exploiting Causal Knowledge to Explain Individual Predictions of Complex Models. Advances in Neural Information Processing Systems 33 (2020).

[16] Aapo Hyvärinen and Petteri Pajunen. 1999. Nonlinear independent component analysis: Existence and uniqueness results. Neural networks 12, 3 (1999), 429–439.

[17] Niki Kilbertus, Mateo Rojas-Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. 2017. Avoiding discrimination through causal reasoning. In Proceedings of the 33rd International Conference on Neural Information Processing Systems. 656–666.

[18] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. In Advances in neural information processing systems. 4066–4076.

[19] Moshe Lithman et al. 2013. UCI machine learning repository.

[20] Scott M Lundberg. 2020. Explaining Quantitative Measures of Fairness. Fair & Responsible AI Workshop @ CHI2020 (2020).

[21] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In Advances in neural information processing systems. 4765–4774.

[22] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. 2018. Learning Adversarially Fair and Transferable Representations. In International Conference on Machine Learning. 3384–3393.

[23] Diego A Martinez, Jeremiah S Hinson, Eili Y Klein, Nathan A Irvin, Mustapha Saheed, Kathleen R Page, and Scott R Levin. 2020. SARS-CoV-2 positivity rate for Latinos in the Baltimore–Washington, DC Region. Jama 324, 4 (2020), 392–395.

[24] Razieh Nabi and Ilya Shpitser. 2018. Fair inference on outcomes. In The 22nd International Conference on Artificial Intelligence and Statistics. PMLR, 2164–2173.

[25] Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. 2000. Causation, prediction, and search. MIT press.

[26] Euohni G Price-Haywood, Jeffrey Burton, Daniel Fort, and Leonardo Seoane. 2020. Hospitalization and mortality among black patients and white patients with Covid-19. New England Journal of Medicine (2020).

[27] Jiaming Song, Pratyusha Kalluri, Aditya Grover, Shengjia Zhao, and Stefano Ermon. 2019. Learning controllable fair representations. In The 22nd International Conference on Artificial Intelligence and Statistics. PMLR, 2164–2173.

[28] Saheed, Kathleen R Page, and Scott R Levin. 2020. SARS-CoV-2 positivity rate for Latinos in the Baltimore–Washington, DC Region. Jama 324, 4 (2020), 392–395.

[29] Simon M Tranter, Juliana Korn-Saavedra, and Scott R Levin. 2020. SARS-CoV-2 positivity rate for Latinos in the Baltimore–Washington, DC Region. Jama 324, 4 (2020), 392–395.

[30] About the American Thoracic Society. 2020.-American Thoracic Society website (2020).

[31] Yongkai Wu, Lu Zhang, Xintao Wu, and Hanghang Tong. 2019. Pc-fairness: A unified framework for measuring causality-based fairness. In Advances in Neural Information Processing Systems. 3404–3414.

[32] Mohammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. 2017. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In Proceedings of the 26th international conference on world wide web. 1171–1180.

[33] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. 2018. Mitigating unwanted biases with adversarial learning. In Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. 335–340.

[34] Lu Zhang, Yongkai Wu, and Xintao Wu. 2018. Causal modeling-based discrimination discovery and removal: Criteria, bounds, and algorithms. IEEE Transactions on Knowledge and Data Engineering 31, 11 (2018), 2035–2050.

A PROOFS

A.1 Proof of Proposition 3.1

Proof. Suppose there is no potential active path between two variables $X_i$ and $X_j$ on $G$, but they are dependent. According to the faithful assumption, consider all directions of undirected edges on $G$, there must be a DAG $G'$ that the conditional independence relationships among variables encoded in $G'$ are consistent with those inferred from observational data. That is, there are active paths from certain $X_i$ to $X_j$ on $G$. Suppose $p'$ is one of these active paths and $p$ is its corresponding path on $G$. According to the definition, $p$ must be a potential active path, which violates the assumption. □
A.2 Proofs of Proposition 3.2 and 3.3

To prove Proposition 3.2 and 3.3, we first prove the following lemma:

**Lemma A.1.** In a DAG, if \( p \) is an active path, then every subpath of \( p \) is an active path.

**Proof.** Suppose \( p' \) is a subpath of \( p \). For each \( X \) that is a node on \( p' \), then \( X \) is also on \( p \). Since \( p \) is an active path, then \( X \) is a non-collider. According to the generality of \( X \), each node on \( p' \) is a non-collider. So \( p' \) is also an active path. \( \Box \)

Then we will prove Proposition 3.2 with Lemma A.1:

**Proof.** (Proposition 3.2) Suppose \( p \) is a potential active path and \( p' \) is any subpath of \( p \). We consider the three conditions in the definition of potential active path. If all edges on \( p \) are directed and \( p \) satisfies the definition of active path. Then all edges on \( p' \) are also directed and \( p' \) is active according to Lemma A.1, and thus a potential active path. Similarly, for the rest two cases, if \( p \) satisfies the condition, we can get \( p' \) also satisfies the same condition by Lemma A.1. So we can have \( p' \) is a potential active path. \( \Box \)

Proof. (Proposition 3.3) Since \( G \) is a DAG, a potential active path is equivalent to an active path according to the definition. Assume \( \exists X_i, X_j \in X(P), X_i \in Adj(X_j) \), but neither of them is prior to the other with respect to \( A \). According to the definition, at least one of the following conditions are satisfied:

- There is no active path from \( A \) to \( \hat{Y} \) through both \( X_i \) and \( X_j \). Without loss of generality, we let \( X_i \rightarrow X_j \). Since \( X_j \in X(P) \), there exists an active path according to Lemma A.1. Considering a new path obtained by appending \( X_j \rightarrow \hat{Y} \) to \( p \), it will be obviously an active path since \( X_j \) and \( \hat{Y} \) must be non-colliders. This new path is an active path from \( A \) to \( \hat{Y} \) and both \( X_i \) and \( X_j \) are on it. So the beginning condition does not hold.

- There exist at least two active paths from \( A \) to \( \hat{Y} \), one is as \( *A \rightarrow \cdots \rightarrow X_i \ldots X_j \cdots \rightarrow \hat{Y} * \) and the other is as \( *A \rightarrow \cdots \rightarrow X_j \ldots X_i \cdots \rightarrow \hat{Y} * \). The paths are pointed out from \( A \) because \( A \) is a source node in \( G \). Without loss of generality, we let \( X_i \rightarrow X_j \). We denote the latter path as \( p \). And we denote the subpath of \( p \) from \( X_i \) to \( X_j \) as \( p' \). \( p' \) must be an active path according to Lemma A.1. There are three possibilities: \( X_j \rightarrow \cdots \rightarrow X_i \), if it is satisfied, \( p' \) and the edge \( X_i \rightarrow X_j \) will form a directed cycle, which violates that \( G \) is a DAG; \( X_j \leftarrow \cdots \leftarrow X_i \) or \( X_j \leftarrow \cdots \leftarrow X_k \rightarrow \cdots \rightarrow X_i \), in both cases, \( X_j \) will be a collider on \( p \) if \( p \) is not active. Since all the situation cannot be satisfied, the beginning condition does not hold.

In conclusion, neither of the conditions above can be satisfied. So the original assumption can not be satisfied, and \( X \) (\( P \)) is completely ordered with respect to \( A \) on \( G \). \( \Box \)

A.3 Proofs of Properties

**Proof.** Efficiency: from Eq. (9), we will have \( \sum_{p_i \in P} \phi_f(p_i) = \phi_f(x(P)) = 0 \). Since \( \phi_f(x(P)) = f(x) \), we have:

\[
\sum_{p_i \in P} \Phi_f(p_i) = (\mathbb{E}|_{\mathcal{A}=1}[\phi_f(x(P))] - \mathbb{E}|_{\mathcal{A}=0}[\phi_f(x(P))])
\]

\[
- (\mathbb{E}|_{\mathcal{A}=1}[\phi_f(x(0))] - \mathbb{E}|_{\mathcal{A}=0}[\phi_f(x(0))])
\]

where \( \mathbb{E}|_{\mathcal{A}=1}[\phi_f(x(P))] - \mathbb{E}|_{\mathcal{A}=0}[\phi_f(x(P))] = \Delta_D(P) \). And when we calculate \( \phi_f(x(0)) \), we only use \( E_i \) and \( \hat{X}(P_Y) \), which are independent to \( A \). So \( \phi_f(x(0)) \) is also independent to \( A \), \( \mathbb{E}|_{\mathcal{A}=1}[\phi_f(x(0))] - \mathbb{E}|_{\mathcal{A}=0}[\phi_f(x(0))] = 0 \).

**Linearity:** In the calculation of \( \phi_f(x(T \cup \{p_i\})) = \phi_f(x(T), \forall x, T \subset P(P_i) \), obviously all the item in the summation of Eq. (9) will be 0. Then \( \phi_f(x(P_i)) \) are both 0. \( \Box \)

B EXTENSION TO OTHER DISPARITY

As discussed in previous work by Baer et al. [5], equalized odds and equalized opportunity satisfy when \( \hat{Y} \perp \!\!\!\!\perp A|Y \). In fact, \( \hat{Y} \perp \!\!\!\!\perp A|Y \) can also induce accuracy parity. To study the conditional dependence between \( X \) and \( \hat{Y} \) on \( Y \), we first introduce a new causality concept:

**Spouse.** If both \( X_i \) and \( X_j \) are parents of \( X_k \), then \( X_j \) is of \( X_i \) and \( X_j \) is a spouse of \( X_i \) with child \( X_k \).

**Potential Active Paths Relative to \( Y \).** The definition of potential active path relative to \( Y \) can be obtained by replace the notions "active path relative to \( Y \)" in the definition of potential active path. To save space, we only write the first condition as an example: if a path \( p \) is a directed path and satisfies the definition of active path relative to \( Y \) in Section 2.2, then it is a potential active path relative to \( Y \).

The potential active path and potential active path relative to \( Y \) have the following relation:

**Proposition B.1.** For any path on \( G \) which does not pass through \( Y \), if \( Y \) is a potential active path, then it is a potential active path relative to \( Y \).

**Proposition B.2.** If there is no potential active path relative to \( Y \) between two variables (neither of them is \( Y \)) on \( G \), then the two variables are conditional independent on \( Y \).

We denote the set of potential active paths from \( A \) to \( \hat{Y} \) relative to \( Y \) as \( P_Y \). We use an example in Figure 6(a) to further illustrate the definition. We can get: \( P_Y = \{A \rightarrow X_1 \rightarrow \hat{Y} \rightarrow X_2 \rightarrow X_3 \rightarrow \hat{Y} \} \). But if we consider the conditioning set \( \{Y\} \), we can see \( Y \) is a collider on this path and belongs to the conditional set. While other nodes \( \{X_1, X_3\} \) are non-colliders and not in \( \{Y\} \), according to the definition, \( A \rightarrow X_1 \rightarrow Y \leftarrow X_2 \rightarrow \hat{Y} \) is an active path relative to \( Y \).

We can propose a similar definition of ordering with respect to \( A \) conditioned on \( Y \). \( X_i \not\perp\!\!\!\!\perp_{A|Y} X_j \) means \( X_i \) is prior to \( X_j \) with respect to \( A \) conditioned on \( Y \). When we further consider the information of \( A \) passes to \( P_Y \). It passes through spouses with child
Y (X₁ → Y ← X₂) as well as from direct predecessor to direct successor (A → X₁). So we propose new concepts other than direct predecessor/successor.

If the two following conditions satisfy, then we call Xᵢ is an informative predecessor of Xⱼ with respect to A relative to Y: 1) Xᵢ > Xⱼ|Y Xⱼ; 2) Xᵢ is adjacent to Xⱼ or Xᵢ, Xⱼ are spouses with child Y. And Xᵢ is called an informative predecessor of Xⱼ with respect to A relative to Y. The set of all informative predecessors of Xᵢ is denoted as Is⁰|Y (Xᵢ). The set of all informative predecessors of Xⱼ is denoted as Ip⁰|Y (Xᵢ).

For Figure 6(a), we have Ip⁰|Y (X₁) = {A}, Ip⁰|Y (X₂) = {A, X₁}. We will write Ip⁰|Y (Xᵢ) as Ip and Is⁰|Y (Xᵢ) as Is for simplicity in the following.

Completely Ordered With Respect To A Relative to Y. X (pᵧ) is defined to be completely ordered with respect to A relative to Y on g if: ∀Xᵢ, Xⱼ ∈ X (pᵧ), we have if Xᵢ and Xⱼ are adjacent or spouses with child Y, one of them must be the informative successor or predecessor of the other.

After defining these concepts, we can directly replace the concept of DP with Ip during the step of grouping features. When we try to decompose f(x) into the path contributions conditioned on Y. We need to learn two sets of functions {gₓᵢ|Y=1} and {gₓᵢ|Y=0}, corresponding to different values of Y. And we need to infer Xᵢ as Xᵢ = gₓᵢ|Y (Ipᵢ, X (pᵧ)). Thus we have ̂Φf(x,y)(pᵢ) obtain by replacing the function gₓᵢ with gₓᵢ|Y in the calculation of ̂Φf(x,y)(pᵢ).

For equalized opportunity, we compute the path contribution to ΔOP as:

\[ \Phi_f(pᵢ) = \mathbb{E}_{X|Y=1,A=1}[\hat{\Phi}_f(x,y)(pᵢ)] - \mathbb{E}_{X|Y=1,A=0}[\hat{\Phi}_f(x,y)(pᵢ)] \]

For accuracy parity, we have:

\[ \Phi_f(pᵢ) = \mathbb{E}_{X|A=1}[\hat{\Phi}_f(x,y)(pᵢ)] - \mathbb{E}_{X|A=0}[\hat{\Phi}_f(x,y)(pᵢ)] \]

For equalized odds, we need to consider the disparity conditioned on Y = 1 and Y = 0 respectively.

C IMPLEMENTATION DETAILS

Generation of Synthetic Data. In experiments on efficiency, we generate the data as follows: a feature Xᵢ is randomly connected to Xⱼ (with Xⱼ pointing to Xᵢ) with 0.2 probability if i > j, otherwise 0. A feature Xᵢ is randomly connected to A (with A pointing to Xᵢ) with 0.4 probability. The generative function of each feature is linear.

In experiments on accuracy, we constrain symmetric structure and parameters to obtain the ground-truths. In S₁, the probability of Y = 1 is calculated by the a linear function of features (normalized to be the in range of [0, 1]). While in S₂, the probability is calculated with additional non-linear transformation.

Model Parameters. When we implement f as MLP, we use the network with one hidden layer and the dimensions of hidden layer are 32/16/8/16 for Synthetic/Adult/COMPAS/Nutrition respectively. The models are implemented and trained with the Python package scikit-learn. We choose ADAM to optimize the model.

Causal Discovery Results on Real-world datasets. Since there is no widely-used causal graph for COMPAS dataset, we generate a PDAG by the following procedure: first we run the PC algorithm on the data, then we determine and correct the direction following the rules: {race, age, gender} → {numbers of prior crimes, numbers of juvenile felonies/juvenile misdemeanors/other juvenile conviction} → {prior crimes} → {original charge degree}.

Estimation of Φf. Since FACTS and all baselines are expensive to compute exactly, we use a Monte Carlo approximation of Eq. 9. In particular, we conduct breadth first search and sample 100 orderings from Π and average across those orderings. For mixed-type data, we binarize all categorical features and follow the recent work of Wang et al. [29] to formulate gₓᵢ and Eᵢ.

D ADDITIONAL EXPERIMENT RESULTS

The results on Adult and COMPAS dataset are shown in Figure 6(b) and 6(c). Due to limited space, we only display the top paths of P on the graphs. And the corresponding path contributions are shown in Figure 7. The edge A → O in 6(c) is undirected. In Adult dataset, marital status acts as the essential feature contributing to disparity. While in COMPAS dataset, age is the largest contributor.