Situation Testing-Based Discrimination Discovery: A Causal Inference Approach

Lu Zhang, Yongkai Wu, and Xintao Wu
University of Arkansas
{lz006,yw009,xintaowu}@uark.edu

Abstract

Discrimination discovery is to unveil discrimination against a specific individual by analyzing the historical dataset. In this paper, we develop a general technique to capture discrimination based on the legally grounded situation testing methodology. For any individual, we find pairs of tuples from the dataset with similar characteristics apart from belonging or not to the protected-by-law group and assign them in two groups. The individual is considered as discriminated if significant difference is observed between the decisions from the two groups. To find similar tuples, we make use of the Causal Bayesian Networks and the associated causal inference as a guideline. The causal structure of the dataset and the causal effect of each attribute on the decision are used to facilitate the similarity measurement. Through empirical assessments on a real dataset, our approach shows good efficacy both in accuracy and efficiency.

1 Introduction

Discrimination refers to the unjustified distinction against an individual based on the group, class or category to which that individual belongs or is perceived to belong. Discrimination is widespread in subjective decision making processes. This situation even deteriorates today due to the increased use of predictive models that are built around the collection of individual data to make important decisions like employment, credit, insurance, etc. A large number of laws and regulations (e.g., Council Directive 2000/78/EC or Title VII of the Civil Rights Act of 1964) have been established to forbid discrimination on several grounds, such as gender, age, pregnancy, sex orientation, race, national origin, religion, genetic information, and disability or illness, which are referred to as the *protected attributes*. Although the laws and regulations for discrimination are clear, proving discrimination in practice is difficult, because discrimination is usually hidden behind various pretexts and it is hard to find clear and sufficient evidence of discrimination in many cases [Rorive, 2009].

Discrimination discovery has been an active research area in data mining recently [Hajian and Domingo-Ferrer, 2013; Kamiran and Calders, 2012; Ruggieri *et al.*, 2010; Romei and Ruggieri, 2014]. To detect discrimination against specific individuals, some researchers proposed to simulate situation testing using data mining algorithms over a historical decision dataset [Luong *et al.*, 2011]. Situation testing is a legally grounded technique for analyzing the discriminatory treatment on an individual. It has been widely adopted both in the United States [Bendick, 2007] and the European Union [Rorive, 2009]. Situation testing is carried out in responding to a complaint about discrimination from an individual. Pairs of testers who are similar to the individual are then sent out to participate in the same decision process (e.g., applying for the same job). For each pair, the two testers possess the same characteristics except the membership to the protected group. For example, in the case of employment, the resumes of a pair of testers with different gender can be made equivalent in the education background, work experience, expertise and skills, and only vary in details and formats to avoid being considered as duplicates. The objective is to measure the treatments or decisions given to the members from the same pair. If one of the pair receives a different decision, the distinction implies discriminatory behavior. To ensure representativeness and "average out" random circumstances, a certain number of tester pairs are usually required by the court.

Employing the situation testing methodology, discrimination can be detected by finding a representative group of tuples from the historical dataset for a target individual. The representative group contains pairs of tuples with similar characteristics apart from belonging to the protected group and non-protected group. The target individual is considered as discriminated if significant difference is observed between the decisions from the two parts of tuples. The key issue in the implementation of situation testing is how to define and determine the representative group. Luong et al. proposed a K-nearest neighbor (K-NN) classification based approach to find the similar tuples [Luong *et al.*, 2011]. In their method, the similarity between two tuples is modeled via a distance function that takes all attributes as the input. The normalized Manhattan distance and overlap measurement are adopted to compute the distance between two tuples. Their approach shows a successful implementation of the situation testing methodology, however, there are several limitations: 1) They use all attributes for computing the distance. However, not all attributes are relevant to the decision, even if they are legally admissible. 2) The distance function does not distinguish the

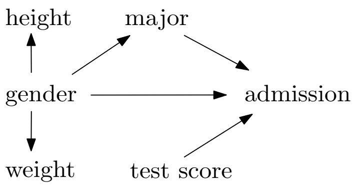


Figure 1: CBN of the illustrative example.

Table 1: Part of the dataset of the illustrating example.

No.	gender	major	score	height	weight	admission
1	F	CS	B	low	low	reject
2	M	CS	B	high	high	admit
3	F	CS	A	low	low	reject
4	M	CS	A	median	median	admit
5	F	CS	C	low	median	reject
6	M	CS	C	high	median	reject
7	M	EE	B	low	low	reject
......						

potentially different effects on discrimination due to different values taken by an attribute. As a result, the identified representative group may not be accurate and the following situation testing could have incorrect results, as demonstrated in our empirical evaluations.

We propose the use of the Causal Bayesian Networks (CBNs) as a guideline in the implementation of situation testing for discrimination discovery. A CBN is a probabilistic graph model which is widely used for causality representation, reasoning and inference [Pearl, 2009]. Our ideas are as follows. First, only the attributes that are the direct causes of the decision should be used in the distance computation. Second, the causal effect of an attribute on the decision can be used to facilitate the measurement of distance between different values taken by the attribute. Consider a toy model for a university admission system shown in Figure 1 and Table 1, which we use as the illustrative example throughout the paper. Intuitively, `height` and `weight` should not be involved in the distance computation as they are irrelevant to the decision. On the other hand, suppose that score grades A, B, C stand for good, median, and failure. It would be more reasonable if the A-to-B distance is measured smaller than the B-to-C distance.

In this paper, we develop a situation testing-based technique to capture discrimination with the support of the CBN. To find similar tuples, we define a distance function on the set of attributes that are the direct causes of the decision. For each attribute, we measure the distance between two domain values taking into consideration the value difference as well as the causal effect on the decision of changing the attribute from one value to the other. Through empirical assessments on a real dataset, we show that both the identification accuracy and efficiency can be significantly improved.

2 Preliminary Concepts

2.1 Data Representation

We consider a historical dataset that consists of a set $\mathcal{T}$ of tuples, each of which describes the profile of an individual. Each tuple $t \in \mathcal{T}$ is explicitly specified by a set of attributes, which contains some protected attributes, the decision attribute, and the non-protected attributes. In the illustrative example, `gender` is the protected attribute, `admission` is the decision attribute, and all the other attributes are the non-protected attributes. Throughout this paper, we use an uppercase alphabet, e.g., X, to represent an attribute, e.g., `major`; a bold uppercase alphabet, e.g., $\mathbf{X}$, to represent a subset of attributes, e.g., {`major`, `score`}. We use a lowercase alphabet, e.g., x, to represent a domain value of attribute X; a bold lowercase alphabet, e.g., $\mathbf{x}$, to represent a value assignment of $\mathbf{X}$. For ease of representation, we assume that there is only one protected attribute. We denote the protected attribute by C, associated with domain values of the protected group c^- (e.g., female) and the non-protected group c^+ (e.g., male); and denote the decision by E, associated with domain values of positive decision e^+ and negative decision e^-. The set of all the other non-protected attributes is denoted by $\mathbf{R}$. Thus, a tuple t can be specified by the triple $(c, \mathbf{r}, e)$. Our approach can extend to handle multiple domain values of C/E and even multiple Cs.

2.2 Causal Bayesian Network

A CBN is a probabilistic graph model which is specified by a directed acyclic graph (DAG) $\mathcal{G} = (\mathbf{V}, \mathbf{A})$ and a group of conditional probability distributions. $\mathbf{V}$ is a set of nodes and $\mathbf{A}$ is a set of arcs. In the DAG, each node represents an attribute. Each arc, denoted by an arrow → pointing from the cause to the effect, represents the direct causality between the two attributes. The absence of an arc between two nodes represents a claim of zero direct effect between the two attributes in all distributions. In this paper, we assume that arc $C \to E$ is present in the CBN. This indicates that there exits some kind of causal effect of C on E in the whole dataset. However, whether a tuple is discriminated or not cannot be identified via the presence of arc $C \to E$. The CBN satisfies the Markov assumption, i.e., each node X is independent of all its non-descendants conditional on its parents $\mathrm{Par}(X)$. We also use $\mathrm{Par}(X)$ to represent a value assignment of the parents of X if no unambiguity occurs in the context. Each node is associated with a conditional probability distribution, i.e., $\mathrm{P}(X|\mathrm{Par}(X))$. The joint probability distribution over all attributes can be computed using the factorization formula [Koller and Friedman, 2009].

The basis for causal inference in the CBN is to measure the impact of interventions. An intervention is an action that forces some subset $\mathbf{X}$ of attributes to take certain values $\mathbf{x}$. The intervention is supposed to be *effective* in the sense that the value assignment of $\mathbf{X}$ is completely determined by the intervention, and *local* in the sense that of all other attributes $\mathbf{Y}$ ($\mathbf{Y} = \mathbf{V} \backslash \mathbf{X}$) the conditional distributions $\mathrm{P}(\mathbf{Y}|\mathrm{Par}(\mathbf{Y}))$ are not affected by the intervention. The CBN permits us to estimate post-intervention distributions from the pre-intervention distributions using the *do*-calculus [Pearl, 2009]. Formally, the intervention that sets the value of $\mathbf{X}$ to $\mathbf{x}$ is represented by $do(\mathbf{X} = \mathbf{x})$ or simply $do(\mathbf{x})$. The post-intervention distribution on $\mathbf{Y}$, i.e., $\mathrm{P}(\mathbf{y}|do(\mathbf{x}))$, is readily expressed as a truncated factorization formula

$$\mathrm{P}(\mathbf{y}|do(\mathbf{x})) = \prod_{Y \in \mathbf{Y}} \mathrm{P}(y|\mathrm{Par}(Y))|_{\mathbf{X}=\mathbf{x}}, \tag{1}$$

where $\mathrm{P}(y|\mathrm{Par}(Y))|_{\mathbf{X}=\mathbf{x}}$ means substituting any attributes in $\mathbf{X}$ involved in the conditional probability with the corresponding values in $\mathbf{x}$. There is also a set of inference rules to facilitate the expression of the post-intervention distributions. For a DAG $\mathcal{G}$ and a subset of nodes $\mathbf{X}$ in $\mathcal{G}$, let $\mathcal{G}_{\overline{\mathbf{X}}}$ denote the graph obtained by deleting all arcs pointing to nodes in $\mathbf{X}$, and let $\mathcal{G}_{\underline{\mathbf{X}}}$ denote the graph obtained by deleting all arcs emerging from nodes in $\mathbf{X}$. The following proposition [Pearl, 2009] states the inferences rules.

Proposition 1 (Rules of *do*-Calculus)**.** *Let $\mathcal{G}$ be the DAG of a CBN. For any disjoint subsets of nodes $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$, $\mathbf{W}$, there are following rules.*

1. $\mathrm{P}(\mathbf{y}|do(\mathbf{x}), \mathbf{z}, \mathbf{w}) = \mathrm{P}(\mathbf{y}|do(\mathbf{x}), \mathbf{w})$*, if* $\mathbf{Y}$ *and* $\mathbf{Z}$ *are d-separated by* $\mathbf{X} \cup \mathbf{W}$ *in* $\mathcal{G}_{\overline{\mathbf{X}}}$*.*
2. $\mathrm{P}(\mathbf{y}|do(\mathbf{x}), do(\mathbf{z}), \mathbf{w}) = \mathrm{P}(\mathbf{y}|do(\mathbf{x}), \mathbf{z}, \mathbf{w})$*, if* $\mathbf{Y}$ *and* $\mathbf{Z}$ *are d-separated by* $\mathbf{X} \cup \mathbf{W}$ *in* $\mathcal{G}_{\overline{\mathbf{X}}\underline{\mathbf{Z}}}$*.*
3. $\mathrm{P}(\mathbf{y}|do(\mathbf{x}), do(\mathbf{z}), \mathbf{w}) = \mathrm{P}(\mathbf{y}|do(\mathbf{x}), \mathbf{w})$*, if* $\mathbf{Y}$ *and* $\mathbf{Z}$ *are d-separated by* $\mathbf{X} \cup \mathbf{W}$ *in* $\mathcal{G}_{\overline{\mathbf{X}}\underline{\mathbf{Z}'}}$*, where* $\mathbf{Z}'$ *is the nodes in* $\mathbf{Z}$ *that are not ancestors of any nodes in* $\mathbf{W}$ *in* $\mathcal{G}_{\overline{\mathbf{X}}}$*.*

The above rules make use of the well-known *d*-separation criterion.

Definition 1 (*d*-Separation)**.** *A path p is said to be blocked by a set of nodes* $\mathbf{Z}$ *if and only if*

1. *p contains a chain $i \rightarrow j$ or a fork $i \leftarrow m \rightarrow j$ such that the middle node m is in* $\mathbf{Z}$*, or*
2. *p contains an collider $i \rightarrow m \leftarrow j$ such that the middle node m is not in* $\mathbf{Z}$ *and no descendant of m is in* $\mathbf{Z}$*.*

A set $\mathbf{Z}$ *is said to d-separate* $\mathbf{X}$ *from* $\mathbf{Y}$ *if and only if* $\mathbf{Z}$ *blocks all paths from* $\mathbf{X}$ *to* $\mathbf{Y}$*.*

3 Strength and Weakness of Modeling Discrimination as Direct Causal Effect

Typically, a discrimination claim means demonstrating a direct causal connection between protected attribute C and decision attribute E [Foster, 2004]. Thus, the CBN and the associated causal inference have an immediate application in discrimination analysis since the discriminatory effect can be straightly represented by the direct causal effect of C on E. Using Pearl's method [Pearl, 2001], we define the direct causal effect as follows:

Definition 2. *Given a tuple $t = (c, \mathbf{r}, e)$, the direct causal effect of protected attribute C on decision attribute E is defined as*

$$\mathrm{P}(e^+|do(c^+, \mathbf{r})) - \mathrm{P}(e^+|do(c^-, \mathbf{r})), \tag{2}$$

where $\mathbf{r}$ *is a value assignment of* $\mathbf{R}$ *and* $\mathbf{R}$ *is the set of all the other attributes.*

Equation (2) indicates that discrimination can be identified by the response of the decision E to the change of the protected attribute C while fixing the values of all the other attributes $\mathbf{R}$. Note that fixing $\mathbf{r}$ is different from conditioning on $\mathbf{r}$. The simply use of $\mathrm{P}(e^+|do(c^+), \mathbf{r}) - \mathrm{P}(e^+|do(c^-), \mathbf{r})$ may create spurious associations between C and E even when there is no direct causal effect of C on E.

To calculate Equation (2), we have the following lemma.

Lemma 1. *Equation (2) is equivalent to*

$$\mathrm{P}(e^+|c^+, \mathbf{q}) - \mathrm{P}(e^+|c^-, \mathbf{q}), \tag{3}$$

where $\mathbf{q}$ *is the value assignment of* $\mathbf{Q}$ *and* $\mathbf{Q}$ *is E's parent set except C, i.e.,* $\mathbf{Q} = \mathrm{Par}(E) \backslash \{C\}$*.*

The proof of this lemma is straightforward using Equation (1). Equation (3) implies that the response of E can be estimated from the dataset by the difference in the decision of tuples with the same characteristics $\mathbf{Q}$ apart from belonging or not in the protected group.

However, there exists a practical limitation with the above measurement in situation testing. In litigation the courts usually require a certain number of tester pairs for situation testing to ensure representativeness [Rorive, 2009]. Researchers in situation testing suggest that there should be at least 50 tester pairs in a single testing [Bendick, 2007]. However, the tuples used for computing the probabilities in Equation (3) must come from the same subgroup specified by $\mathbf{q}$. Thus, there is no guarantee that the number of these tuples can satisfy any given requirement.

Inspired by [Luong *et al.*, 2011], we tackle this limitation by finding $2K$ tuples that are closest to the target tuple, where K can be designated by the court. The CBN can provide a guidance of finding tuples. Intuitively, we can first select the tuples from the subgroup specified by $\mathbf{q}$, and then select the tuples from the following appropriate subgroups. The details of our method will be discussed in the next section.

4 Discrimination Discovery

In this section, we describe our causal inference approach for discrimination discovery. We formalize situation testing in our context. Given a target tuple t with $c = c^-$ and $e = e^-$, we rank all the tuples according to their distances to t using a distance function defined on an appropriate subset of non-protected attributes. Then, we select the tuples that are within top-$2K$ highest in the ranking for one-to-one pairing. The selected tuples with c^+ are added into set $\mathbf{S}^+$, and the selected tuples with c^- are added into set $\mathbf{S}^-$. We compare the proportion p_1 of tuples with e^+ among the tuples in $\mathbf{S}^+$, and the proportion p_2 of tuples with e^+ among the tuples in $\mathbf{S}^-$. The difference between p_1 and p_2, i.e., $p_1 - p_2$ reflects the bias due to the membership to the protected group. If $p_1 - p_2$ is greater than a threshold τ, then t is considered as being discriminated. The value of threshold τ used for discrimination depends on the law. For instance, the 1975 British legislation for sex discrimination sets $\tau = 0.05$, namely a 5% difference.

We assume that we have a CBN that is compatible with the dataset. We also make a reasonable assumption that there is no arc pointing to C in the CBN, as the protected attribute is always an inherent nature of an individual. A CBN can be learned from data and expert knowledge. In the past decades, many algorithms have been proposed to learn CBNs, and they are proved to be quite successful [Spirtes *et al.*, 2000; Neapolitan and others, 2004; Colombo and Maathuis, 2014].

How a CBN is constructed by a learning algorithm or domain experts and how to measure the quality of a CBN are beyond the scope of this paper.

In the following, we first present the distance function and then describe the discovery algorithm.

4.1 Distance Function

Consider two tuples t and t', the distance function $\mathrm{d}(t, t')$ measures the dissimilarity between the two tuples. To define the distance function, researchers first establish a distance metric for measuring the per-attribute distance, and then compute the joint effect by summing up all the per-attribute distances. In [Luong *et al.*, 2011], the distance between two domain values r_k and r'_k of attribute $R_k \in \mathbf{R}$ is defined as the value difference, where the normalized Manhattan distance is employed for ordinal/interval attributes, and the overlap measurement is employed for categorical attributes. The normalized Manhattan distance is defined as:

$$\mathrm{md}(r_k, r'_k) = \frac{|r_k - r'_k|}{range},$$

where *range* is used to normalize the result, defined as the difference between the maximum and the minimum of the attribute. The overlap measurement is defined as:

$$\mathrm{ol}(r_k, r'_k) = \begin{cases} 0 & \text{if } r_k = r'_k \\ 1 & \text{otherwise} \end{cases}$$

Overall, the difference between t and t' is given by

$$\mathrm{d}(t, t') = \sum_{k=1}^{|\mathbf{R}|} \mathrm{VD}(r_k, r'_k),$$

where

$$\mathrm{VD}(r_k, r'_k) = \begin{cases} \mathrm{md}(r_k, r'_k) & \text{if } R_k \text{ is ordinal/interval} \\ \mathrm{ol}(r_k, r'_k) & \text{if } R_k \text{ is categorical} \end{cases}$$

As discussed in Section 1, the above distance function has two limitations. First, Equation (3) implies that the distance function should be defined on $\mathbf{Q}$, since they are the direct causes of the decision. Other attributes are either not causally related to E or have the causal effects on E that are transmitted by the direct causes. Including these attributes in the distance computation may lead to incorrect results in the similarity measurement. Consider the illustrative example in Section 1. Suppose that tuple 1 is the target for testing, and we want to find the closest tuple from the tuples listed in the table. If we use all non-protected attributes to compute the distance, tuples 3 and 7 are the closest ones as both of them have only one mismatch. However, from the CBN we can see that, `height` and `weight` are not causally related to `admission` and should not be involved in the computation. In fact, tuple 2 is closest to the target since their `major`s and `score`s are exactly the same.

Second, when measuring the per-attribute distance, the causal effect of each attribute on the decision can reveal important information relating to similarity. The response of the decision to change of the attribute reflects the difference in how the two domain values affect the decision. Thus, two values can be considered to be closer if changing the attribute from one value to the other produces smaller influence on the decision. Consider the same above example and we want to measure the distance between difference values of attribute `score`. The distance between A and B and the distance between B and C are measured as equivalent if only the value difference is considered, e.g., using the Manhattan distance. However, from the tuples listed in the table we can see that, both the admission rates for A and B are 50%, and the admission rate for C is 0%. Thus, the causal effect of `score` on `decision` can facilitate to more accurately characterize the similarity in situations where A and B are closer than B and C with respect to the admission. Furthermore, the per-attribute distance should be instance dependent. For example, although both the score difference between tuples 3 and 2 and that between tuples 3 and 7 are the same A-to-B difference, they should not be equal since tuples 3 and 2 apply to the same major while tuple 7 applies to another.

Based on the first observation, we define the distance function on the basis of $\mathbf{Q}$, where $\mathbf{Q} = \mathrm{Par}(E)\backslash\{C\}$. For the second observation, we measure the causal effect on the decision of each attribute $Q_k \in \mathbf{Q}$. We model the change of Q_k from q_k to q'_k as two interventions that force Q_k to take that two values respectively while keeping all other attributes the same as $\mathbf{q}$. According to the definition of interventions [Pearl, 2009], we define as follows:

Lemma 2. *Given a tuple t, the response of the decision to the change of Q_k from q_k to q'_k is given by*

$$\mathrm{CE}(q_k, q'_k) = \mathrm{P}(e^+|do(\mathbf{q})) - \mathrm{P}(e^+|do(q'_k, \mathbf{q}\backslash\{q_k\})), \qquad (4)$$

where $\mathrm{P}(e^+|do(\mathbf{q}))$ is the effect of the intervention that forces $\mathbf{Q}$ to take value $\mathbf{q}$, and $\mathrm{P}(e^+|do(q'_k, \mathbf{q}\backslash\{q_k\}))$ is the effect of the intervention that forces Q_k to take value q'_k and other attributes in $\mathbf{Q}$ to take the same value as $\mathbf{q}$.

Note that $\mathrm{CE}(q_k, q'_k)$ is instance dependent. For different tuples, the measure would not be equal even for the same pair of q_k, q'_k if the tuples possess different profiles $\mathbf{q}$. Then, we define the distance function for tuples t, t' as follows.

Definition 3. *The distance between tuples t, t' is given by*

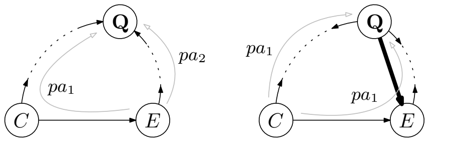
$$\mathrm{d}(t, t') = \sum_{k=1}^{|\mathbf{Q}|} \left| \mathrm{CE}(q_k, q'_k) \cdot \mathrm{VD}(q_k, q'_k) \right|, \qquad (5)$$

where $\mathbf{Q} = \mathrm{Par}(E)\{C\}$.

In our distance function, the production of $\mathrm{CE}(q_k, q'_k)$ and $\mathrm{VD}(q_k, q'_k)$ can be interpreted in two aspects: (1) $\mathrm{CE}(q_k, q'_k)$ can be considered as the weight of $\mathrm{VD}(q_k, q'_k)$, indicating how significant this value difference is with regard to the decision; (2) $\mathrm{VD}(q_k, q'_k)$ can also be considered as the weight of $\mathrm{CE}(q_k, q'_k)$, indicating to what extend this causal effect is relating to the similarity between the two values. A more general version may include the scale parameters α, β for the two metrics CE and VD:

$$\mathrm{d}(t, t') = \sum_{k=1}^{|\mathbf{Q}|} \left| \mathrm{CE}(q_k, q'_k)^{\alpha} \cdot \mathrm{VD}(q_k, q'_k)^{\beta} \right|.$$

In the following, we show how to calculate $\mathrm{CE}(q_k, q'_k)$. Directly using Equation (1) involves summing over all values of



(a) (b)

Figure 2: CBNs that show: (a) paths connecting E and $\mathbf{Q}$ in $\mathcal{G}_{\overline{C}\underline{\mathbf{Q}}}$; (b) paths connecting C and $\mathbf{Q}$ in $\mathcal{G}_{\overline{\mathbf{Q}}}$.

the attributes in $\mathbf{R} \cup \{C\} \backslash \mathbf{Q}$, which is too tedious. Thus, we make use of the inference rules in Proposition 1. The result is given by the following lemma.

Lemma 3. *Given a tuple t,*

$$\mathrm{CE}(q_k, q_k') = \sum_C \Big(\mathrm{P}(e^+|c, \mathbf{q}) \cdot \mathrm{P}(c) - \mathrm{P}(e^+|c, q_k', \mathbf{q}\backslash\{q_k\}) \cdot \mathrm{P}(c)\Big).$$

Proof. As defined in Equation (4), we have $\mathrm{CE}(q_k, q_k') = \mathrm{P}(e^+|do(\mathbf{q})) - \mathrm{P}(e^+|do(q_k', \mathbf{q}\backslash\{q_k\}))$. By conditioning and summing over the values of C, we can write $\mathrm{P}(e^+|do(\mathbf{q}))$ as

$$\mathrm{P}(e^+|do(\mathbf{q})) = \sum_C \mathrm{P}(e^+|c, do(\mathbf{q})) \cdot \mathrm{P}(c|do(\mathbf{q})).$$

For term $\mathrm{P}(e^+|c, do(\mathbf{q}))$, we can show that E and $\mathbf{Q}$ are d-separated given C in $\mathcal{G}_{\overline{C}\underline{\mathbf{Q}}}$. Since all arcs pointing to C (in fact we have assumed that there is no arc pointing to C) and all arcs emerging from $\mathbf{Q}$ are deleted in $\mathcal{G}_{\overline{C}\underline{\mathbf{Q}}}$, there are only two types of paths connecting E and $\mathbf{Q}$, as illustrated in Figure 2a. The first type of paths pa_1 starts from an arc pointing to E and ends with an arc pointing to any node in $\mathbf{Q}$. Since C is the only parent of E in $\mathcal{G}_{\overline{C}\underline{\mathbf{Q}}}$, pa_1 must pass through C where C acts as the middle node in a fork. According to Condition 1 in Definition 1, pa_1 is blocked by C. The second type of paths pa_2 starts from an arc emerging from E and ends with an arc pointing to any node in $\mathbf{Q}$. pa_2 cannot consists entirely the arcs with the direction that points from E to $\mathbf{Q}$, otherwise the mono-directional path $E \to \cdots \mathbf{Q} \to E$ forms a circle in $\mathcal{G}$, which contradicts to that $\mathcal{G}$ is a DAG. Thus, there exists at least one collider on pa_2. According to Condition 2 in Definition 1, pa_2 is blocked by $\emptyset$. Therefore, E and $\mathbf{Q}$ are d-separated by C in $\mathcal{G}_{\overline{C}\underline{\mathbf{Q}}}$. By applying Rule 2 in Proposition 1, we have

$$\mathrm{P}(e^+|c, do(\mathbf{q})) = \mathrm{P}(e^+|c, \mathbf{q}).$$

For term $\mathrm{P}(c|do(\mathbf{q}))$, as illustrated in Figure 2b, there is no arc pointing to C, and all arcs pointing to $\mathbf{Q}$ are deleted in $\mathcal{G}_{\overline{\mathbf{Q}}}$. Thus, there is only type of paths pa_1, which starts with an arc emerging from C and ends with an arc emerging from any node in $\mathbf{Q}$. Clearly, there must exist at least one collider on the path. According to Condition 2 in Definition 1, pa_1 is blocked by $\emptyset$. Therefore, C and $\mathbf{Q}$ are d-separated in $\mathcal{G}_{\overline{\mathbf{Q}}}$. By applying Rule 3 Proposition 1, we have

$$\mathrm{P}(c|do(\mathbf{q})) = \mathrm{P}(c).$$

Thus, it follows that

$$\mathrm{P}(e^+|do(\mathbf{q})) = \sum_C \mathrm{P}(e^+|c, \mathbf{q}) \cdot \mathrm{P}(c).$$

Similarly we can calculate $\mathrm{P}(e^+|do(q_k', \mathbf{q}\backslash\{q_k\}))$. As a result, we have

$$\mathrm{CE}(q_k, q_k') = \sum_C \Big(\mathrm{P}(e^+|c, \mathbf{q}) \cdot \mathrm{P}(c) - \mathrm{P}(e^+|c, q_k', \mathbf{q}\backslash\{q_k\}) \cdot \mathrm{P}(c)\Big).$$

□

4.2 Discovery Algorithm

Now we describe our discrimination discovery algorithm (*CBN-DD*). The pseudocode of the algorithm is presented in Algorithm 1. Given the target tuple t, the algorithm first finds the set $\mathbf{Q}$ in the CBN. Then, the algorithm is divided into three stages. In the first stage (lines 2-8), all tuples are ranked using the distance function (Equation (5)). Since the distance function is defined on $\mathbf{Q}$, all the tuples are divided into homogeneous subgroups according to their value assignment of $\mathbf{Q}$. All tuples in a subgroup are measured the same distance to t. Thus, the algorithm ranks the tuples on the basis of the subgroups. For each value assignment $\mathbf{q}^l$ of $\mathbf{Q}$, define subgroup $\mathbf{g}^l = \mathbf{g}^{l,+} \cup \mathbf{g}^{l,-}$ containing all the tuples t' with $\mathbf{q}' = \mathbf{q}^l$, where tuples with $c' = c^+$ are contained in $\mathbf{g}^{l,+}$ and tuples with $c' = c^-$ are contained in $\mathbf{g}^{l,-}$. Then, the distance $d[l]$ is calculated between t and the first tuple in $\mathbf{g}^l$ (denoted as $\mathbf{g}^l[0]$). The ranking orders are stored in an array L. In the second stage (lines 9-17), tuples that are ranked top-$2K$ are selected, starting from subgroup $\mathbf{g}^{L[0]}$ which ranks the highest. Within each subgroup $\mathbf{g}^{L[i]}$, we select the same number of tuples from $\mathbf{g}^{L[i],+}$ and $\mathbf{g}^{L[i],-}$ to guarantee strict one-to-one pairing. The tuples selected from $\mathbf{g}^{L[i],+}$ are added into set $\mathbf{S}^+$, and tuples selected from $\mathbf{g}^{L[i],-}$ are added into set $\mathbf{S}^-$. The algorithm selects as much as possible tuples from one subgroup, and then moves on to the next. The loop ends until the number of the selected tuples reaches $2K$. Finally, in the third stage (lines 18-20), p_1 and p_2 are calculated from $\mathbf{S}^+$ and $\mathbf{S}^-$, and the discrimination judgment is made based on $p_1 - p_2$.

The total computational complexity of the algorithm is $O(M \cdot |\mathbf{Q}| + K)$, where M is the number of value assignments of $\mathbf{Q}$. Usually, M is far smaller than the total number of tuples since $\mathbf{Q}$ is a subset of all the attributes.

5 Experiments

To evaluate the proposed discrimination discovery algorithm, we have conducted experiments by using the Dutch Census of 2001 [Netherlands, 2001], which is widely used in discrimination discovery literature. The dataset consists of 60421 tuples, each of which is described by 12 attributes. We treat `sex` (female and male) as the protected attribute and `occupation` (occupation_w_low_income, occupation_w_high_income) as the decision attribute. The CBN is constructed by TETRAD [Glymour and others, 2004], an open-source platform for causal modeling. We employ the original PC algorithm [Spirtes *et al.*, 2000] and the significance level $\alpha = 0.05$ for the structure learning. For experiment details including the CBN please refer to our technical report [Zhang *et al.*, 2016].

From the CBN, an arc from `sex` to `occupation` is observed, and the set $\mathbf{Q}$ is identified as {`age`, `education_level`, `country_birth`, `economic_status`}. We randomly select 200 tuples as the targets for discrimination testing. For each target tuple, we search similar tuples

Algorithm 1: Discrimination Discovery (*CBN-DD*)

```
$\mathbf{Q} = \mathrm{Par}(E)\backslash\{C\}$;
foreach value assignment $\mathbf{q}^l$ of $\mathbf{Q}$ do
    $\mathbf{g}^{l,+} = \{t' | \mathbf{q}' = \mathbf{q}^l, c' = c^+\}$;
    $\mathbf{g}^{l,-} = \{t' | \mathbf{q}' = \mathbf{q}^l, c' = c^-\}$;
    $\mathbf{g}^l = \mathbf{g}^{l,+} \cup \mathbf{g}^{l,-}$;
    $d[l] = \mathrm{d}(t, \mathbf{g}^l[0])$;
end
Rank $\mathbf{g}^l$ in ascending order according to $d[l]$ and store the
  orders in $L[l]$;
$n = 0; i = 0; \mathbf{S}^+ = \mathbf{S}^- = \emptyset$;
while $n < K$ do
    select $\min(|\mathbf{g}^{L[i],+}|, |\mathbf{g}^{L[i],-}|, K - n)$ tuples $\mathbf{t}^+$ from $\mathbf{g}^{l,+}$;
    select $\min(|\mathbf{g}^{L[i],+}|, |\mathbf{g}^{L[i],-}|, K - n)$ tuples $\mathbf{t}^-$ from $\mathbf{g}^{l,-}$;
    $\mathbf{S}^+ = \mathbf{S}^+ \cup \mathbf{t}^+$;
    $\mathbf{S}^- = \mathbf{S}^- \cup \mathbf{t}^-$;
    $n+ = |\mathbf{t}^+|$;
    $i + +$;
end
$p_1$ = the positive decision rate in $\mathbf{S}^+$;
$p_2$ = the positive decision rate in $\mathbf{S}^-$;
return $p_1 - p_2 > \tau$ ? true : false;
```

Table 2: Summarized results for 200 target tuples.

(a) Tuples identified as discriminated

K	*CBN-DD*	*KNN-DD*	diff
10	150	141.3	62.5
50	151.6	159.4	56
90	152.9	158.8	55.7

(b) Accuracy

K	*CBN-DD*		*KNN-DD*	
	TP	TN	TP	TN
10	73.3	63.1	46	66.2
50	85.3	77.6	42.2	76.2
90	81.5	83.9	38.1	81.2

from the whole dataset. The threshold τ is set as 0.05. We compare our algorithm (abbrev. as *CBN-DD*) with the algorithm in [Luong *et al.*, 2011] (abbrev. as *KNN-DD*). We repeat the test 10 times. The average number of tuples identified as discriminated by the two algorithms as well as the number of different judgments for each tuple are shown in Table 2a. As can be seen, there is significant difference between the judgments made by the two algorithms.

To measure the accuracy of the discrimination identification, we manually modify the dataset to obtain a data with ground truth. We first create a "clean" dataset by completely removing the attribute `sex` from the dataset and randomly assigning a gender to each tuple based on the original population. The "clean" dataset contains no bias against gender and hence each tuple can be labeled as "non-discriminated". Then, we manually introduce bias into the data. We simulate a situation where a domain user is responsible for making decisions and has a strong prejudice against females. So, he changes the decisions of 100 female tuples from positive to negative. We aim to evaluate how accurately two algorithms can identify them. For comparison, we also randomly select another 100 tuples that are labeled "non-discriminated" from the dataset and add them for testing. Similarly, the test is repeated 10 times and the average results of true positive (TP) and true negative (TN) are shown in Table 2b. It can be seen that, when $K = 50$, *CBN-DD* reports 85.3 tuples as discriminated among the 100 tuples that are true positive, and 77.6 tuples as non-discriminated among the 100 tuples that are true negative. In general, *CBN-DD* outperforms *KNN-DD* in both TP and TN for various values of K. For *CBN-DD*, TP decreases when $K = 10$ and $K = 90$. This is probably because of the randomness in selecting tuples when K is too small, and having to select dissimilar tuples when K is too large. How to suggest an appropriate K is left for future work.

The average CPU time for one target tuple is 0.3s for *CBN-DD* and 20.3s for *KNN-DD*, showing that *CBN-DD* is much more computationally efficient than *KNN-DD*. Due to the large computational cost of *KNN-DD*, we only test 200 tuples in our evaluations.

6 Related Work

A number of data mining techniques have been proposed to discover discrimination in the literature. Pedreshi et al. proposed to extract from the dataset classification rules which represent certain discrimination patterns [Pedreshi *et al.*, 2008; Pedreschi *et al.*, 2009]. If the presence of the protective attribute increases the confidence of a classification rule, it indicates possible discrimination in the data set. Based on that, [Mancuhan and Clifton, 2014] further proposed to use the Bayesian network to compute the confidence of the classification rules for detecting discrimination. Differently, conditional discrimination, where part of discrimination may be explained by other legally grounded attributes, was studied in [Zliobaite *et al.*, 2011; Hajian *et al.*, 2015]. [Bonchi *et al.*, 2015] proposed a random walk method based on the Suppes-Bayes causal network. In [Wu and Wu, 2015], the authors proposed the use of loglinear modeling to capture and measure discrimination and developed a method for discrimination prevention by modifying significant coefficients from the fitted loglinear model. All the above work suffer from legal weaknesses. Our work follows [Luong *et al.*, 2011], which is based on the legally grounded situation testing methodology.

Another issue related to anti-discrimination is discrimination prevention, which aims to build non-discriminatory predictive models when the historical data contains discrimination [Kamiran and Calders, 2009; 2012; Calders and Verwer, 2010; Kamishima *et al.*, 2011]. In all the proposed methods, discrimination needs to be identified and measured first before it can be removed. Our work complements discrimination prevention in that we provide technique for capturing and measuring discrimination, which advances the understanding related to both discrimination discovery and prevention.

7 Conclusions

In this paper, we have investigated the discrimination discovery problem on the basis of the situation testing methodology. We improve the method in [Luong *et al.*, 2011] with the support of the CBN. We have defined a distance function on the direct causes of the decision, which takes into consideration the value difference as well as the causal effect of each attribute on the decision. The empirical assessments using the real data have been conducted. The results show that both the identification accuracy and efficiency have been significantly improved with our proposed algorithm.

Acknowledgment

This work was supported in part by U.S. National Institute of Health (1R01GM103309).

References

[Bendick, 2007] Marc Bendick. Situation testing for employment discrimination in the united states of america. *Horizons stratégiques*, (3):17–39, 2007.

[Bonchi *et al.*, 2015] Francesco Bonchi, Sara Hajian, Bud Mishra, and Daniele Ramazzotti. Exposing the probabilistic causal structure of discrimination. *arXiv preprint arXiv:1510.00552*, 2015.

[Calders and Verwer, 2010] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2):277–292, 2010.

[Colombo and Maathuis, 2014] Diego Colombo and Marloes H Maathuis. Order-independent constraint-based causal structure learning. *The Journal of Machine Learning Research*, 15(1):3741–3782, 2014.

[Foster, 2004] Sheila R Foster. Causation in antidiscrimination law: Beyond intent versus impact. *Hous. L. Rev.*, 41:1469, 2004.

[Glymour and others, 2004] Clark Glymour et al. The TETRAD project. http://www.phil.cmu.edu/tetrad, 2004.

[Hajian and Domingo-Ferrer, 2013] Sara Hajian and Josep Domingo-Ferrer. A methodology for direct and indirect discrimination prevention in data mining. *Knowledge and Data Engineering, IEEE Transactions on*, 25(7):1445–1459, 2013.

[Hajian *et al.*, 2015] Sara Hajian, Josep Domingo-Ferrer, Anna Monreale, Dino Pedreschi, and Fosca Giannotti. Discrimination-and privacy-aware patterns. *Data Mining and Knowledge Discovery*, 29(6):1733–1782, 2015.

[Kamiran and Calders, 2009] Faisal Kamiran and Toon Calders. Classifying without discriminating. In *2nd International Conference on Computer, Control and Communication (IC4 2009)*, pages 1–6. IEEE, 2009.

[Kamiran and Calders, 2012] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.

[Kamishima *et al.*, 2011] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. Fairness-aware learning through regularization approach. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, pages 643–650. IEEE, 2011.

[Koller and Friedman, 2009] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[Luong *et al.*, 2011] Binh Thanh Luong, Salvatore Ruggieri, and Franco Turini. k-nn as an implementation of situation testing for discrimination discovery and prevention. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 502–510. ACM, 2011.

[Mancuhan and Clifton, 2014] Koray Mancuhan and Chris Clifton. Combating discrimination using bayesian networks. *Artificial intelligence and law*, 22(2):211–238, 2014.

[Neapolitan and others, 2004] Richard E Neapolitan et al. *Learning bayesian networks*, volume 38. Prentice Hall Upper Saddle River, 2004.

[Netherlands, 2001] Statistics Netherlands. Volkstelling. https://sites.google.com/site/faisalkamiran/, 2001.

[Pearl, 2001] Judea Pearl. Direct and indirect effects. In *Proceedings of the seventeenth conference on uncertainty in artificial intelligence*, pages 411–420. Morgan Kaufmann Publishers Inc., 2001.

[Pearl, 2009] Judea Pearl. *Causality*. Cambridge university press, 2009.

[Pedreschi *et al.*, 2009] Dino Pedreschi, Salvatore Ruggieri, and Franco Turini. Measuring discrimination in socially-sensitive decision records. In *Society for Industrial and Applied Mathematics. Proceedings of the SIAM International Conference on Data Mining*, page 581. Society for Industrial and Applied Mathematics, 2009.

[Pedreshi *et al.*, 2008] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. Discrimination-aware data mining. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 560–568. ACM, 2008.

[Romei and Ruggieri, 2014] Andrea Romei and Salvatore Ruggieri. A multidisciplinary survey on discrimination analysis. *The Knowledge Engineering Review*, 29(05):582–638, 2014.

[Rorive, 2009] Isabelle Rorive. Proving discrimination cases: The role of situation testing. 2009.

[Ruggieri *et al.*, 2010] Salvatore Ruggieri, Dino Pedreschi, and Franco Turini. Data mining for discrimination discovery. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(2):9, 2010.

[Spirtes *et al.*, 2000] Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search*, volume 81. MIT press, 2000.

[Wu and Wu, 2015] Yongkai Wu and Xintao Wu. Using log-linear model for discrimination discovery and prevention. Technical Report DPL-2015-002, University of Arkansas, 2015.

[Zhang *et al.*, 2016] Lu Zhang, Yongkai Wu, and Xintao Wu. Situation testing-based discrimination discovery: A causal inference approach. Technical Report DPL-2016-002, University of Arkansas, 2016.

[Zliobaite *et al.*, 2011] Indre Zliobaite, Faisal Kamiran, and Toon Calders. Handling conditional discrimination. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 992–1001. IEEE, 2011.