Finding and Listing Front-door Adjustment Sets

Hyunchai Jeong
Purdue University
jeong3@purdue.edu

Jin Tian
Iowa State University
jtian@iastate.edu

Elias Bareinboim
Columbia University
eb@cs.columbia.edu

Abstract

Identifying the effects of new interventions from data is a significant challenge found across a wide range of the empirical sciences. A well-known strategy for identifying such effects is Pearl’s front-door (FD) criterion [26]. The definition of the FD criterion is declarative, only allowing one to decide whether a specific set satisfies the criterion. In this paper, we present algorithms for finding and enumerating possible sets satisfying the FD criterion in a given causal diagram. These results are useful in facilitating the practical applications of the FD criterion for causal effects estimation and helping scientists to select estimands with desired properties, e.g., based on cost, feasibility of measurement, or statistical power.

1 Introduction

Learning cause and effect relationships is a fundamental challenge across data-driven fields. For example, health scientists developing a treatment for curing lung cancer need to understand how a new drug affects the patient’s body and the tumor’s progression. The distillation of causal relations is indispensable to understanding the dynamics of the underlying system and how to perform decision-making in a principled and systematic fashion [27, 37, 2, 30, 1, 23, 24].

One of the most common methods for learning causal relations is through Randomized Controlled Trials (RCTs, for short) [8]. RCTs are considered as the “gold standard” in many fields of empirical research and are used throughout the health and social sciences as well as machine learning and AI. In practice, however, RCTs are often hard to perform due to ethical, financial, and technical issues. For instance, it may be unethical to submit an individual to a certain condition if such condition may have some potentially negative effects (e.g., smoking). Whenever RCTs cannot be conducted, one needs to resort to analytical methods to infer causal relations from observational data, which appears in the literature as the problem of causal effect identification [26, 27].

The causal identification problem asks whether the effect of holding a variable $X$ at a constant value $x$ on a variable $Y$, written as $P(Y|do(X = x))$, or $P(Y|do(x))$, can be computed from a combination of observational data and causal assumptions. One of the most common ways of eliciting these assumptions is in the form of a causal diagram represented by a directed acyclic graph (DAG), where its nodes and edges describe the underlying data generating process. For instance, in Fig. 1a three nodes $X, Z, Y$ represent variables, a directed edge $X \rightarrow Z$ indicates that $X$ causes $Z$, and a dashed-bidirected edge $X \leftrightarrow Y$ represents that $X$ and $Y$ are confounded by unmeasured (latent) factors. Different methods can solve the identification problem, including Pearl’s celebrated do-calculus [26] as well as different algorithmic solutions [40, 43, 12].

In practice, researchers often rely on identification strategies that generate well-known identification formulas. One of the arguably most popular strategies is identification by covariate adjustment. Whenever a set $Z$ satisfies the back-door (BD) criterion [26] relative to the pair $X$ and $Y$, where $X$ and $Y$ represent the treatment and outcome variables, respectively, the causal effect $P(Y|do(x))$ can be evaluated through the BD adjustment formula $\sum_z p(y|x, z)p(z)$.
Despite the popularity of the covariate adjustment technique for estimating causal effects, there are still settings in which no BD admissible set exists. For example, consider the causal diagram $G$ in Fig. 1a. There clearly exists no set to block the BD path from $X$ to $Y$, through the bidirected arrow, $X \leftrightarrow Y$. One may surmise that this effect is not identifiable and the only one of evaluating the interventional distribution is through experimentation. Still, this is not the case. The effect $P(Y|do(x))$ is identifiable from $G$ and the observed distribution $P(x, y, z)$ over $\{X, Y, Z\}$ by another classic identification strategy known as the front-door (FD) criterion [26]. In particular, through the following FD adjustment formula provides the way of evaluating the interventional distribution:

$$P(Y|do(x)) = \sum_z P(z|x) \sum_{x'} P(y|x', z) P(x'). \quad (1)$$

We refer to Pearl and Mackenzie [28, Sec. 3.4] for an interesting account of the history of the FD criterion, which was the first graphical generalization of the BD case. The FD criterion is drawing more attention in recent years. For applications of the FD criterion, see, e.g., Hünemund and Bareinboim [13] and Glynn and Kashin [10]. Statistically efficient and doubly robust estimators have recently been developed for estimating the FD estimand in Eq. (1) from finite samples [9], which are still elusive for arbitrary estimands identifiable in a diagram despite recent progress [18, 19, 5, 20, 43].

Both the BD and FD criteria are only descriptive, i.e., they specify whether a specific set $Z$ satisfies the criteria or not, but do not provide a way to find an admissible set $Z$. In addition, in many situations, it is possible that multiple adjustment sets exist. Consider for example the causal diagram in Fig. 1b and the task of identifying the effect of $X$ on $Y$. The distribution $P(Y|do(x))$ can indeed be identified by the FD criterion with a set $Z = \{A, B, C\}$ given by the expression in Eq. (1) (with $Z$ replaced with $\{A, B, C\}$). Still, what if the variable $B$ is costly to measure or encodes some personal information about patients which is undesirable to be shared due to ethical concerns? In this case, the set $Z = \{A, C\}$ also satisfies the FD criterion and may be used. Even when both $B$ and $C$ are unmeasured, the set $Z = \{A\}$ is also FD admissible.

This simple example shows that a target effect can be estimated using different adjustment sets leading to different probability expressions over different set of variables, which has important practical implications. Each variable implies different practical challenges in terms of measurement, such as cost, availability, privacy. Each estimand has different statistical properties in terms of sample complexity, variance, which may play a key role in the study design [31, 11, 32, 36]. Algorithms for finding and listing all possible adjustment sets are hence very useful in practice, which will allow scientists to select an adjustment set that exhibits desirable properties. Indeed, algorithms have been developed in recent years for finding one or listing all BD admissible sets [38, 39, 41, 29, 42]. However, no such algorithm is currently available for finding/listing FD admissible sets.

The goal of this paper is to close this gap to facilitate the practical applications of the FD criterion for causal effects estimation and help scientists to select estimand with certain desired properties. Specifically, the contributions of this paper are as follows:

1. We develop an algorithm that finds an admissible front-door adjustment set $Z$ in a given causal diagram in polynomial time (if one exists). We solve a variant of the problem that imposes constraints $I \subseteq Z \subseteq R$ for given sets $I$ and $R$, which allows a scientist to constrain the search to include specific subsets of variables or exclude variables from search perhaps due to cost, availability, or other technical considerations.

2. We develop a sound and complete algorithm that enumerates all front-door adjustment sets with polynomial delay - the algorithm takes polynomial amount of time to return each new admissible set, if one exists, or return failure whenever it exhausted all admissible sets.

\[\text{Code is available at } \text{https://github.com/CausalAI LAB/FrontdoorAdjustmentSets}\]
2 Preliminaries

Notation. We write a variable in capital letters ($X$) and its value as small letters ($x$). Bold letters, $X$ or $x$, represent a set of variables or values. We use kinship terminology to denote various relationships in a graph $\mathcal{G}$ and denote the parents, ancestors, and descendants of $X$ (including $X$ itself) as $Pa(X)$, $An(X)$, and $De(X)$, respectively. Given a graph $\mathcal{G}$ over a set of variables $V$, a subgraph $\mathcal{G}_X$ consists of a subset of variables $X \subseteq V$ and their incident edges in $\mathcal{G}$. A graph $\mathcal{G}$ can be transformed: $\mathcal{G}_X$ is the graph resulting from removing all incoming edges to $X$, and $\mathcal{G}_X$ is the graph with all outgoing edges from $X$ removed. A DAG $\mathcal{G}$ may be moralized into an undirected graph where all directed edges of $\mathcal{G}$ are converted into undirected edges, and for every pair of nonadjacent nodes in $\mathcal{G}$ that share a common child, an undirected edge that connects such pair is added [22].

A path $\pi$ from a node $X$ to a node $Y$ in $\mathcal{G}$ is a sequence of edges where $X$ and $Y$ are the endpoints of $\pi$. A node $W$ on $\pi$ is said to be a collider if $W$ has converging arrows into $W$ in $\pi$, e.g., $\rightarrow W \leftarrow \rightarrow$ or $\leftarrow W \rightarrow$. $\pi$ is said to be blocked by a set $Z$ if there exists a node $W$ on $\pi$ satisfying one of the following two conditions: 1) $W$ is a collider, and neither $W$ nor any of its descendants are in $Z$, or 2) $W$ is not a collider, and $W$ is in $Z$ [25]. Given three disjoint sets $X$, $Y$, and $Z$ in $\mathcal{G}$, $Z$ is said to $d$-separate $X$ from $Y$ in $\mathcal{G}$ if and only if $Z$ blocks every path from a node in $X$ to a node in $Y$ according to the $d$-separation criterion [25], and we say that $Z$ is a separator of $X$ and $Y$ in $\mathcal{G}$.

Structural Causal Models (SCMs). We use Structural Causal Models (SCMs, for short) [27] as our basic semantical framework. An SCM is a 4-tuple $(U, V, F, P(u))$, where 1) $U$ is a set of exogenous (latent) variables, 2) $V$ is a set of endogenous (observed) variables, 3) $F$ is a set of functions $\{f_v\}_{v \in V}$ that determine the value of endogenous variables, e.g., $v \leftarrow f_v(pa_v, u_v)$ is a function with $PA_v \subseteq V \setminus \{v\}$ and $U_v \subseteq U$, and 4) $P(u)$ is a joint distribution over the exogenous variables $U$. Each SCM induces a causal diagram $\mathcal{G}$ [3 Def. 13] where every variable $v \in V$ is a vertex and directed edges in $\mathcal{G}$ correspond to functional relationships as specified in $F$ and dashed bidirected edges represent common exogenous variables between two vertices. Within the structural semantics, performing an intervention and setting $X = x$ is represented through the do-operator, $do(X = x)$, which encodes the operation of replacing the original functions of $X$ (i.e., $f_X(pa_X, u_X)$) by the constant $x$ and induces a submodel $M_x$ and an interventional distribution $P(v|do(x))$.

Classic Causal Effects Identification Criteria. Given a causal diagram $\mathcal{G}$ over $V$, an effect $P(y|do(x))$ is said to be identifiable in $\mathcal{G}$ if $P(y|do(x))$ is uniquely computable from the observed distribution $P(v)$ in any SCM that induces $\mathcal{G}$ [27] p. 77.

A path between $X$ and $Y$ with an arrow into $X$ is known as a back-door path from $X$ to $Y$. The celebrated back-door (BD) criterion [26] provides a sufficient condition for effect identification from observational data, which states that if a set $Z$ of non-descendants of $X$ blocks all BD paths from $X$ to $Y$, then the causal effect $P(y|do(x))$ is identified by the BD adjustment formula:

$$P(y|do(x)) = \sum_z P(y|x, z)P(z)$$

(2)

Another classic identification condition that is key to the discussion in this paper is known as the front-door criterion, which is defined as follows:

**Definition 1.** (Front-door (FD) Criterion [26]) A set of variables $Z$ is said to satisfy the front-door criterion relative to the pair $(X, Y)$ if

1. $Z$ intercepts all directed paths from $X$ to $Y$,
2. There is no unblocked back-door path from $X$ to $Z$, and
3. All back-door paths from $Z$ to $Y$ are blocked by $X$, i.e., $X$ is a separator of $Z$ and $Y$ in $\mathcal{G}_Z$.

If $Z$ satisfies the FD criterion relative to the pair $(X, Y)$, then $P(y|do(x))$ is identified by the following FD adjustment formula [26]:

$$P(y|do(x)) = \sum_z P(z|x) \sum_{x'} P(y|x', z)P(x').$$

(3)
3 Finding A Front-door Adjustment Set

In this section, we address the following question: given a causal diagram \( \mathcal{G} \), is there a set \( Z \) that satisfies the FD criterion relative to the pair \( (X, Y) \) and, therefore, allows us to identify \( P(y \mid do(x)) \) by the FD adjustment? We solve a more general variant of this question that imposes a constraint \( I \subseteq Z \subseteq R \) for given sets \( I \) and \( R \). Here, \( I \) are variables that must be included in \( Z \) (\( I \) could be empty) and \( R \) are variables that could be included in \( Z \) (\( R \) could be \( V \setminus (X \cup Y) \)). Note the constraint that variables in \( W \) cannot be included can be enforced by excluding \( W \) from \( R \). Solving this version of the problem will allow scientists to put constraints on candidate adjustment sets based on practical considerations. In addition, this version will form a building block for an algorithm that enumerates all FD admissible sets in a given \( \mathcal{G} \) - the algorithm \textsc{ListFDSets} (shown in Alg. 2 in Section 4) for listing all FD admissible sets will utilize this result during the recursive call.

We have developed a procedure called \textsc{FindFDSet} shown in Alg. 1 that outputs a FD adjustment set \( Z \) relative to \( (X, Y) \) satisfying \( I \subseteq Z \subseteq R \), or outputs \( \perp \) if none exists, given a causal diagram \( \mathcal{G} \), disjoint sets of variables \( X \) and \( Y \), and two sets of variables \( I \) and \( R \).

\textbf{Example 1.} Consider the causal graph \( \mathcal{G}' \), shown in Fig. 1b, with \( X = \{X\} \), \( Y = \{Y\} \), \( I = \emptyset \) and \( R = \{A, B, C, D\} \). Then, \textsc{FindFDSet} outputs \( \{A, B, C\} \). With \( I = \{D\} \) and \( R = \{A, B, C, D\} \), \textsc{FindFDSet} outputs \( \perp \) as no FD adjustment set that contains \( D \) is available.

\textsc{FindFDSet} runs in three major steps. Each step identifies candidate variables that incrementally satisfy each of the conditions of the FD criterion relative to \( (X, Y) \). First, \textsc{FindFDSet} constructs a set of candidate variables \( R' \), with \( I \subseteq R' \subseteq R \), such that every subset \( Z \) with \( I \subseteq Z \subseteq R' \) satisfies the second condition of the FD criterion (i.e., there is no BD path from \( X \) to \( Z \)). Next, \textsc{FindFDSet} generates a set of candidate variables \( R'' \), with \( I \subseteq R'' \subseteq R' \), such that for every variable \( v \in R'' \), there exists a set \( Z \) with \( I \subseteq Z \subseteq R' \) and \( v \in Z \) that further satisfies the third condition of the FD criterion, that is, all BD paths from \( Z \) to \( Y \) are blocked by \( X \). Finally, \textsc{FindFDSet} outputs a set \( Z \) that further satisfies the first condition of the FD criterion - \( Z \) intercepts all causal paths from \( X \) to \( Y \).

\textbf{Step 1 of \textsc{FindFDSet}}

In Step 1, \textsc{FindFDSet} calls the function \textsc{GetCand2ndFDC} (presented in Fig. 2) to construct a set \( R' \) that consists of all the variables \( v \in R \) such that there is no BD path from \( X \) to \( v \) (\( R' \) is set to empty if there is a BD path from \( X \) to \( I \)). Then, there is no BD path from \( X \) to any set \( I \subseteq Z \subseteq R' \) since, by definition, there is no BD path from \( X \) to \( Z \) if and only if there is no BD path from \( X \) to any \( v \in Z \).

\begin{algorithm}[H]
\caption{\textsc{FindFDSet}(\( \mathcal{G}, X, Y, I, R \))}
\begin{algorithmic}[1]
\STATE \textbf{Input:} \( \mathcal{G} \) a causal diagram; \( X, Y \) disjoint sets of variables; \( I, R \) sets of variables.
\STATE \textbf{Output:} \( Z \) a set of variables satisfying the front-door criterion relative to \( (X, Y) \) with the constraint \( I \subseteq Z \subseteq R \).
\STATE \textbf{Step 1:}
\STATE \quad \textbf{Step 1.1:} \( R' \leftarrow \text{GetCand2ndFDC}(\mathcal{G}, X, I, R) \)
\STATE \quad \textbf{Step 1.2:} \( \text{if} \ R' = \perp \text{ then: return } \perp \)
\STATE \textbf{Step 2:}
\STATE \quad \textbf{Step 2.1:} \( R'' \leftarrow \text{GetCand3rdFDC}(\mathcal{G}, X, Y, I, R') \)
\STATE \quad \textbf{Step 2.2:} \( \text{if} \ R'' = \perp \text{ then: return } \perp \)
\STATE \textbf{Step 3:}
\STATE \quad \textbf{Step 3.1:} \( \mathcal{G}' \leftarrow \text{GetCausalPathGraph}(\mathcal{G}, X, Y) \)
\STATE \quad \textbf{Step 3.2:} \( \text{if} \ \text{TestSep}(\mathcal{G}', X, Y, R'') = \text{True then:} \)
\STATE \quad \quad \text{return } Z = R'' \)
\STATE \quad \textbf{Step 3.3:} \( \text{else: return } \perp \)
\end{algorithmic}
\end{algorithm}

Figure 2: A function that outputs the set of candidate variables satisfying the second condition of the FD criterion.
GETCAND2NDFDC iterates through each variable \( v \in R \) and checks if there exists an open BD path from \( X \) to \( v \) by calling the function TESTSEP(\( \mathcal{G}_X, X, v, \emptyset \)). TESTSEP(\( \mathcal{G}, A, B, C \)) returns True if \( C \) is a separator of \( A \) and \( B \) in \( \mathcal{G} \), or False otherwise. Therefore, TESTSEP(\( \mathcal{G}_X, X, v, \emptyset \)) returns True if \( v \in G_X \) (i.e., there is no BD path from \( X \) to \( v \)), or False otherwise. If TESTSEP returns False, then \( v \) is removed from \( R' \) because every set \( Z \) containing \( v \) violates the second condition of the FD criterion relative to \( (X,Y) \).

Example 2. Continuing Example 1. With \( I = \emptyset \) and \( R = \{A, B, C, D\} \), GETCAND2NDFDC outputs a set \( R' = \{A, B, C\} \). \( D \) is excluded from \( R' \) since there exists a BD path from \( X \) to \( D \), and any set containing \( D \) violates the second condition of the FD criterion relative to \( (X,Y) \).

Lemma 1 (Correctness of GETCAND2NDFDC). GETCAND2NDFDC(\( \mathcal{G}, X, I, R \)) generates a set of variables \( R' \) with \( I \subseteq R' \subseteq R \) such that \( R' \) satisfies the third condition of the FD criterion relative to \( (X,Y) \). Further, every subset \( Z \subseteq R' \) satisfies the second condition of the FD criterion relative to \( (X,Y) \), and every set \( I \subseteq Z \subseteq R \) that satisfies the second condition of the FD criterion relative to \( (X,Y) \) must be a subset of \( R' \).

Step 2 of FINDFDSET

In Step 2, FINDFDSET calls the function GETCAND3RDIFDC presented in Fig. 3 to generate a set \( R'' \) consisting of all the variables \( v \in R' \) such that there exists a set \( Z \) containing \( v \) with \( I \subseteq Z \subseteq R' \) that further satisfies the third condition of the FD criterion relative to \( (X,Y) \) (i.e., all BD paths from \( Z \) to \( Y \) are blocked by \( X \)). In other words, \( R'' \) is the union of all \( Z \) with \( I \subseteq Z \subseteq R' \) that satisfies the third condition of the FD criterion.

GETCAND3RDIFDC iterates through each variable \( v \in R' \) and calls the function GETDEP(\( \mathcal{G}, X, Y, \{v\}, R' \)) in line 5. Presented in Fig. 4, GETDEP returns a subset \( Z' \subseteq R' \setminus \{v\} \) such that all BD paths from \( Z = \{v\} \cup Z' \) to \( Y \) are blocked by \( X \) (if there exists such \( Z' \)). If GETDEP returns \( \bot \), then there exists no \( Z \) containing \( v \) that satisfies the third condition of the FD criterion relative to \( (X,Y) \), so \( v \) is removed from \( R'' \).

Example 3. Continuing Example 2. Given \( I = \emptyset \) and \( R' = \{A, B, C\} \), GETCAND3RDIFDC outputs \( R'' = \{A, B, C\} \) because for each variable \( v \in R'' \), GETDEP finds a set \( Z' \) such that \( \{v\} \cup Z' \) satisfies the third condition of the FD criterion relative to \( (X,Y) \). For \( v = A, Z' = \emptyset \), for \( v = B, Z' = \{A\} \), and for \( v = C, Z' = \{A\} \).

Next, we explain how the function GETDEP(\( \mathcal{G}, X, Y, T, R' \)) works. First, GETDEP constructs an undirected graph \( \mathcal{M} \) in a way that the paths from \( T \) to \( Y \) in \( \mathcal{M} \) represent all BD paths from \( T \) to \( Y \) that cannot be blocked by \( X \) in \( \mathcal{G} \). The auxiliary function MORALIZE(\( \mathcal{G} \)) moralizes a given graph \( \mathcal{G} \) into an undirected graph. The moralization is performed on the subgraph over \( AN(T \cup X \cup Y) \) instead of \( \mathcal{G} \) based on the following property: \( T \) and \( Y \) are \( d \)-separated by \( X \) in \( \mathcal{G} \) if and only if \( X \) is a \( T-Y \) node cut (i.e., removing \( X \) disconnects \( T \) and \( Y \) in \( \mathcal{G} \))

Figure 3: A function that outputs the set of candidate variables potentially satisfying the second and third conditions of the FD criterion.
1: function GetDep(G, X, Y, T, R')
2:   Output: Z' \subseteq R' \setminus T, a set of variables such that T \cup Z' satisfies the third condition of the FD criterion relative to (X, Y).
3:   \( G' \leftarrow G_{A\setminus(T \cup U)} \)
4:   \( G' \leftarrow G' \) with all bidirected edges \( A \leftrightarrow B \) replaced by a latent node \( L_{AB} \) and two edges \( L_{AB} \rightarrow A \) and \( L_{AB} \rightarrow B \)
5:   \( G'' \leftarrow G'_T \)
6:   \( M \leftarrow \text{MORALIZE}(G'') \) then remove X
7:   \( Z' \leftarrow \emptyset, Q \leftarrow T \) and mark all \( v \in T \) as visited
8:   while Q \neq \emptyset do
9:      u \leftarrow Q.\text{POP}()
10:     if u \in Y then: return \bot
11:    NR \leftarrow \text{GETNEIGHBORS}(u, M) \cap R' \) that are not visited
12:    G'' \leftarrow G'' \setminus \text{NR}
13:    M \leftarrow \text{MORALIZE}(G'') \) then remove X
14:    N' \leftarrow \text{GETNEIGHBORS}(u, M) \) that are not visited
15:    NR' \leftarrow \{w \in NR \mid \text{there exists an incoming arrow into } w \in G\}
16:    N \leftarrow N' \cup NR', Z' \leftarrow Z' \cup NR
17:    Q.\text{INSERT}(N) \) and mark all \( w \in N \) as visited
18: end while
19: return Z'
20: end function

Figure 4: A function that facilitates the construction of a set that satisfies the third condition of the FD criterion.

Y where \( II \) are not disconnected in \( G_{T \cup Z} \). After adding NR to \( Z' \), M must be reconstructed in a way that reflects the setting where all outgoing edges of NR are removed. BFS will be performed on such modified M.

GetDep checks if there exists any set of nodes N to be visited further. N consists of two sets: 1) \( N' \), all observed neighbors of u that are still reachable from u, even after removing all outgoing edges of NR, and 2) \( NR' \subseteq NR \) where for every node \( w \in NR \), there exists an incoming arrow into \( w \) in G. All nodes in NR' must be checked because there might exist some BD path \( \pi \) from u to y \( \in Y \) that cannot be blocked by X in G. If \( \pi \) cannot be disconnected from u to y, then the set Z will violate the third condition of the FD criterion relative to (X, Y).

The BFS continues until either a node \( y \in Y \) is visited, or no more nodes can be visited. If GetDep returns a set Z', then we have that all BD paths from T to Y that cannot be blocked by X in G have been disconnected in \( G_{T \cup Z} \) while ensuring that there exists no BD path from Z to Y that cannot be blocked by X in G. Therefore, Z satisfies the third condition of the FD criterion relative to (X, Y).

Otherwise, if GetDep returns \( \bot \) (i.e., \( y \) is visited), then there does not exist any Z containing T that satisfies the third condition of the FD criterion relative to (X, Y). This is because there exists a BD path \( \pi \) from \( t \in T \) to \( y \) that cannot be blocked by X in G; removing outgoing edges of all \( w \in R' \) that intersect \( \pi \) cannot disconnect \( \pi \) from \( t \) to \( y \).

Example 4. Expanding on Example 3 to show the use of function GetDep. Consider the case when \( v = B \). Then, Q = T = \{B\} \) and u = B is popped from Q at line 9. We have NR = \{A\}, N' = \emptyset, NR' = \{A\}, N = \{A\}, and Z' = \emptyset. Since N is inserted to Q at line 17, u = A is popped from Q in the next iteration of while loop. Then, NR = \emptyset, N' = \emptyset, NR' = \emptyset, and N = \emptyset. Since Q is empty, the while loop terminates and GetDep returns Z' = \{A\}.

Example 5. Illustrating the use of function GetDep. Let I = \emptyset, R' = \{B, C\}, and v = B. \( Q = T = \{B\} \) and u = B is popped from Q at line 9. We have NR = \emptyset, N' = \{A\}, NR' = \emptyset, N = \{A\}, and Z' = \emptyset. Since N is inserted to Q at line 17, u = A is popped from Q in the second iteration of while loop. \( NR = NR' = \{C\}, N' = N = \{C, D, Y\}, Z' = \{C\}, \) and Q = \{C, D, Y\}. On the third iteration, u = C is popped from Q. NR = NR' = N' = N = \emptyset and Q = \{D, Y\}. On the fourth iteration, u = D is popped from Q. NR = NR' = N' = N = \emptyset and Q = \{Y\}. Next, u = Y is popped from Q. Since u \( \in \{Y\} \), GetDep returns \( \bot \) at line 10. There exists no set \( Z' \subseteq (R' \setminus T) = \{C\} \) such that T \cup Z' satisfies the third condition of the FD criterion relative to (\{X\}, \{Y\}).
Lemma 2 (Correctness of \textsc{GetCand3rdFDC}). \textsc{GetCand3rdFDC}(\mathcal{G}, X, Y, I, R') in Step 2 of Alg. 7 generates a set of variables \(R''\) where \(I \subseteq R'' \subseteq R'\). \(R''\) consists of all and only variables \(v\) such that there exists a subset \(Z\) with \(I \subseteq Z \subseteq R'\) and \(v \in Z\) that satisfies the third condition of the FD criterion relative to \((X, Y)\). Further, every set \(Z\) with \(I \subseteq Z \subseteq R\) that satisfies both the second and the third conditions of the FD criterion must be a subset of \(R''\).

Remark: Even though every set \(Z\) with \(I \subseteq Z \subseteq R'\) that satisfies the third condition of the FD criterion must be a subset of \(R''\), not every subset \(Z \subseteq R''\) satisfies the third condition of the FD criterion, as illustrated by the following example.

Example 6. In Example 3, \textsc{GetCand3rdFDC} outputs \(R'' = \{A, B, C\}\). However, for \(Z = \{B\}\), the BD path \(B \leftarrow A \rightarrow D \rightarrow Y\) is not blocked by \(X\); for \(Z = \{C\}\), the BD path \(C \leftarrow A \rightarrow D \rightarrow Y\) is not blocked by \(X\).

On the other hand, we show that \(Z = R''\) itself satisfies the third condition of the FD criterion, as shown in the following.

Lemma 3. \(R''\) generated by \textsc{GetCand3rdFDC} (in Step 2 of Alg. 7) satisfies the third condition of the FD criterion, that is, all BD paths from \(R''\) to \(Y\) are blocked by \(X\).

Step 3 of \textsc{FindFDS}

Finally, in Step 3, \textsc{FindFDS} looks for a set \(Z \subseteq R''\) that satisfies the first condition of the FD criterion relative to \((X, Y)\), that is, \(Z\) intercepts all causal paths from \(X\) to \(Y\). To facilitate checking whether a set \(Z\) intercepts all causal paths from \(X\) to \(Y\), we introduce the concept of causal path graph defined as follows.

Definition 2. (Causal Path Graph) Let \(\mathcal{G}\) be a causal graph and \(X, Y\) disjoint sets of variables. A causal path graph \(\mathcal{G}'\) relative to \((\mathcal{G}, X, Y)\) is a graph over \(X \cup Y \cup \text{PCP}(X, Y)\), where \(\text{PCP}(X, Y) = (\text{De}(X) \cup X) \cap \text{An}(Y)\) constructed as follows:

1. Construct a subgraph \(\mathcal{G}'' = \mathcal{G}_{X \cup Y \cup \text{PCP}(X, Y)}\).
2. Construct a graph \(\mathcal{G}' = \mathcal{G}''_{XY}\), then remove all bidirected edges from \(\mathcal{G}'\).

A function \textsc{GetCausalPathGraph}(\(\mathcal{G}, X, Y\)) for constructing a causal path graph is presented in Fig. 9 in the Appendix.

Example 7. Consider the causal graph \(\mathcal{G}'\) shown in Fig. 1b with \(X = \{X\}\), and \(Y = \{Y\}\). The causal path graph \(\mathcal{G}''\) relative to \((\mathcal{G}', \{X\}, \{Y\})\) is shown in Fig. 5b. All causal paths from \(\{X\}\) to \(\{Y\}\) in \(\mathcal{G}'\) are present in \(\mathcal{G}''\).

After constructing a causal path graph \(\mathcal{G}'\) relative to \((\mathcal{G}, X, Y)\), we use the function \textsc{TestSep}(\(\mathcal{G}', X, Y, Z\)) to check if \(Z\) is a separator of \(X\) and \(Y\) in \(\mathcal{G}'\). Based on the following lemma, \(Z\) satisfies the first condition of the FD criterion relative to \((X, Y)\) if and only if \textsc{TestSep} returns True.

Lemma 4. Let \(\mathcal{G}\) be a causal graph and \(X, Y, Z\) disjoint sets of variables. Let \(\mathcal{G}'\) be the causal path graph relative to \((\mathcal{G}, X, Y)\). Then, \(Z\) satisfies the first condition of the FD criterion relative to \((X, Y)\) if and only if \(Z\) is a separator of \(X\) and \(Y\) in \(\mathcal{G}'\).

Given the set \(R''\) that contains every set \(Z\) with \(I \subseteq Z \subseteq R\) that satisfies both the second and the third conditions of the FD criterion (Lemma 2), it may appear that we need to search for a set \(Z \subseteq R''\) that satisfies the first condition of the FD criterion. We show instead that all we need is to check whether the set \(R''\) itself satisfies the first condition which has been shown to satisfy the second and third conditions by Lemma 3. This result is summarized in the following lemma.

\footnote{A notation introduced by van der Zander et al. [41] to denote the set of variables on proper causal paths from \(X\) to \(Y\).}
We have developed an algorithm named L

Algorithm 2 ListFDSets \((G, X, Y, I, R)\)

1: Input: \(G\) a causal diagram; \(X, Y\) disjoint sets of variables; \(I, R\) sets of variables.
2: Output: Listing front-door adjustment set \(Z\) relative to \((X, Y)\) where \(I \subseteq Z \subseteq R\).
3: if \(\text{FindFDSet}(G, X, Y, I, R) \neq \emptyset\) then:
4: \(\text{if } I = R\) then: Output \(I\)
5: else:
6: \(v \leftarrow \text{any variable from } R \setminus I\)
7: \(\text{ListFDSets}(G, X, Y, I \cup \{v\}, R)\)
8: \(\text{ListFDSets}(G, X, Y, I, R \setminus \{v\})\)

4 Enumerating Front-door Adjustment Sets

Our goal in this section is to develop an algorithm that lists all FD adjustment sets in a causal diagram. In general, there may exist exponential number of such sets, which means that any listing algorithm will take exponential time to list them all. We will instead look for an algorithm that has an interesting property known as polynomial delay \[38\]. In words, poly-delay algorithms output the first answer (or indicate none is available) in polynomial time, and take polynomial time to output each consecutive answer as well. Consider the following example.

Example 9. Consider the three causal graphs in Fig. 6. In \(G\) shown in Fig. 6a, there exists 9 valid FD adjustment sets relative to \((\{X\}, \{Y\})\). In \(G'\), presented in Fig. 6b two variables \(A_3\) and \(B_3\) are added from \(G\), forming an additional causal path from \(X\) to \(Y\). 27 FD adjustment sets relative to \((\{X\}, \{Y\})\) are available in \(G'\). If another causal path \(X \rightarrow A_4 \rightarrow B_4 \rightarrow Y\) is added to \(G'\), then there are 81 FD adjustment sets relative to \((\{X\}, \{Y\})\). As shown in Fig. 6c in a graph \(G''\) with similar pattern with causal path \(X \rightarrow A_i \rightarrow B_i \rightarrow Y, i = 1, \ldots, n\), there are at least \(3^n\) number of FD adjustment sets.

We have developed an algorithm named ListFDSets, shown in Alg. 2 that lists all FD adjustment sets \(Z\) relative to \((X, Y)\) satisfying \(I \subseteq Z \subseteq R\) with polynomial delay, given a causal diagram \(G\), disjoint sets of variables \(X\) and \(Y\), and two sets of variables \(I\) and \(R\).

Example 10. Consider the causal graph \(G'\) shown in Fig. 1b with \(X = \{X\}, Y = \{Y\}\). \(I = \emptyset\) and \(R = \{A, B, C, D\}\). ListFDSets outputs \(\{A, B, C\}, \{A, B\}, \{A, C\}, \{A\}\) one by one, and finally stops as no more adjustment sets exist.
Although this assumption is quite common throughout the causal inference literature, more recent results have shown that while the BD criterion provides a complete set of valid adjustment sets, not all causal effects can be identified using this criterion. This is because the BD criterion is based on a directed acyclic graph (DAG) representation of the causal relationship, which may not capture all possible causal structures.

Theorem 2. Let \( \mathcal{G} \) be a causal graph, \( X, Y \) disjoint sets of variables, and \( I, R \) sets of variables. \( \text{LISTFDSETS}(\mathcal{G}, X, Y, I, R) \) enumerates all and only sets \( Z \) with \( I \subseteq Z \subseteq R \) that satisfy the FD criterion relative to \((X, Y)\) in \( O(n^4(n+m)) \) delay where \( n \) and \( m \) represent the number of nodes and edges in \( \mathcal{G} \).

5 Discussion and Conclusions

This work has some limitations and can be extended in several directions. First, Pearl’s FD criterion is not complete with respect to the BD adjustment formula (Eq. 3). While the BD criterion has been algebraicized to a complete criterion for BD adjustment [35], it is an interesting open problem to come up with a complete criterion for sets satisfying the FD adjustment. Second, this work assumes that the causal diagram is given (or inferred based on scientists’ domain knowledge and/or data). Although this assumption is quite common throughout the causal inference literature, more recent work has moved to finding BD admissible sets given incomplete or partially specified causal diagrams, e.g., maximal ancestral graphs (MAGs) [41], partial ancestral graphs (PAGs) [29], and completed partially directed acyclic graphs (CPDAGs) [29]. There are algorithms capable of performing causal effect identification in a data-driven fashion from an equivalence class [14] [15] [16] [17]. It is an
interesting and certainly challenging future work to develop algorithms for finding FD admissible sets in these types of graphs. Some recent work has proposed data-driven methods for finding and listing BD admissible sets, using an anchor variable, when the underlying causal diagram is unknown \[7,6,33\]. A criterion for testing FD-admissibility of a given set using data and an anchor variable is also available \[4\]. Other interesting future research topics include developing algorithms for finding minimal, minimum, and minimum cost FD adjustment sets, which are available for the BD adjustment sets \[42\], as well as algorithms for finding conditional FD adjustment sets \[13,9\]. Having said all of that, we believe that the results developed in this paper is a necessary step towards solving these more challenging problems.

After all, we started from the observation that identification is not restricted to BD adjustment, and Pearl’s FD criterion provides a classic strategy for estimating causal effects from observational data and qualitative knowledge encoded in the form of a causal diagram. The criterion is drawing more attention in recent years and statistically efficient and doubly robust estimators have been developed for estimating the FD estimand from finite samples. In this paper, we develop algorithms that given a causal diagram \( \mathcal{G} \), find an admissible FD set (Alg. \[1\] FINDFdset, Thm. \[1\]) and enumerate all admissible FD sets with polynomial delay (Alg. \[2\] LISTFdsets, Thm. \[2\]). We hope that the methods and algorithms proposed in this work will help scientists to use the FD strategy for causal effects estimation in the practical applications and are useful for scientists in study design to select covariates based on desired properties, including cost, feasibility, and statistical power.

References


Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Check Section [5]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] We do not expect this work has negative societal impacts.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] Check the Appendix.

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]