Lorenzo Giammei*

AN INTEGRATED APPROACH TO CAUSALITY: THE ROLE OF CAUSAL GRAPHS

Abstract. Causal questions are central for most biomedical and social science studies. The main frameworks that allow the analysis of causal relations are Potential Outcomes and Causal Graphs. The approaches have often been compared, contrasting their relative strengths. This paper evaluates the implications of merging the two methodologies in an integrated approach. In particular, we assess how the limits of one can be compensated by the solutions provided by the other. The outlined approach employs causal graphs to discover and formalize a causal model that is then used as a guide to implementing potential outcomes identification strategies. The integrated approach could be beneficial to both frameworks. The assumptions of potential outcome methods can be assessed directly from a causal graph even in high dimensional contexts, thus making the obtained causal estimates more reliable. On the other hand, causal graphs can benefit from the several ad hoc identification strategies that have been developed in the potential outcomes literature.

Keywords: causality, causal graphs, potential outcomes.

1. Introduction

The study of cause and effect relations motivates most research in social, demographic and health sciences. Investigating causality usually means assessing if and how a certain intervention, often called treatment, affects an outcome of interest. The early work of Neyman and Iwaszkiewicz (1935), Fisher (1949) and Cox (1958) in the field of randomized experiments constituted a first step towards a rigorous analysis of causality. Based on these studies Rubin (1974) formalizes one of the most relevant approaches to causality: the *Potential Outcomes* (PO) framework. The framework has then been enriched with many contributions that proposed new methods and applications (Imbens and Rubin, 2015; Rosenbaum, 2018). PO have a strong connection with economics since its early stages as its concepts are rooted in the work of Tinbergen (1930) and Haavelmo (1943). PO methods are now widely applied in statistics and economics and many econometric textbooks solely rely on this approach (Angrist and Pischke, 2008; Imbens and Rubin, 2015). The other main approach to deal with causality is the *Causal Graph* framework. Note that causal graphs, also called Causal Bayesian Networks or Causal Diagrams, can be seen as part of a wider mdel called structural causal model (SCM) (Pearl, 2000). In a SCM the causal graph is also associated to a set of equations that describe causal relations between the nodes of the graph. Here we will however only focus on the causal graph component, that is sufficient for answering causal queries concerning the effect of interventions. Causal graphs are described in Pearl (2000) and share some elements with the previous work on path diagrams in Wright (1921). The framework has been subsequently developed and enriched with several contributions that extended its applicability and strengthened its results (Pearl, 2000; Tian and Pearl, 2002; Bareinboim and Pearl, 2016; Huang and Valtorta, 2006). Causal graphs are now frequently used in epidemiology, computer science and some social sciences, though they are still uncommon in economics.

The relative advantages of the two frameworks have been recently reviewed and compared in Imbens (2020) and Hünermund and Bareinboim (2019). Both papers show some specific causal problems where one approach is more appropriate than the other and vice-versa, thus revealing that, at least in part, the two are complementary and could benefit from each other. The idea of an integrated approach also starts to appear in some causal inference textbooks, such as Morgan and Winship (2015) and Cunningham (2021), however integrated applications are still very rare in practice. In this paper, we assess how PO and Causal Graphs can be combined and the implications of

* Sapienza University of Rome

carrying out such an approach. The basic ideas of the frameworks will be described focusing on when the limits of one way of proceeding are compensated by the other. Particular attention will be put on causal discovery techniques, a resource that is often overlooked when comparing PO and causal graphs. Throughout the paper, we provide some basic examples in which the combination of the frameworks can improve the results' quality and reliability.

Section 2 will outline the PO framework, its main assumptions, results and limits. Section 3 is instead devoted to Causal Graphs. The basic terminology is presented and the principal features are described with the help of some examples. Then we show how causal effect estimation can be performed from causal graphs and how the process can be integrated with PO methods. Finally, in section 4 we introduce the concept of causal discovery; we explain how structural learning algorithms work and why they can be valuable for causal effect estimation.

2. Potential Outcomes

The Potential Outcomes framework originates from the work of Splawa-Neyman et al. (1990) and Rubin (1974) on randomized controlled trials (RCT). The name of the framework comes from its peculiar notation $Y_i(t)$ that denotes the *potential outcome* for unit i when receiving the treatment level $T = t$. In the case of a binary treatment T takes value 1 if unit i is treated and 0 otherwise. Accordingly, $Y_i(1)$ represents the PO we would observe for unit i if it was treated and $Y_i(0)$ the potential outcome if unit i was a control. The causal effect of T on Y can therefore be computed by comparing summary statistics of the potential outcomes distribution. The resulting causal estimate is usually called the average treatment effect (ATE) and can be expressed in different ways, such as

$$ATE = E[Y_i(1) - Y_i(0)] \qquad \text{or} \qquad ATE = \frac{E[Y_i(1)]}{E[Y_i(0)]}.$$

However, the ATE cannot be estimated directly from data since only one of the potential outcomes is observed for each unit i. Units receive only one level of treatment, creating a missing data problem. This is sometimes referred to as the fundamental problem of causal inference (Holland, 1986). PO literature contributed to answering this problem in the context of randomized experiments. In this setting, treatment is assigned randomly to the units of the sample, thus rendering T independent of the potential outcomes $T_i \perp\!\!\!\perp (Y_i(0), Y_i(1))$. This scenario, together with the assumption that there is no interference between units (SUTVA)(Imbens and Rubin, 2015), ensure that an unbiased estimate of the ATE can be obtained by computing the difference

$$\bar{Y_t} - \bar{Y_c}, \qquad \text{with} \qquad \bar{Y_t} = \frac{1}{N_t} \sum_{i:T_i=1} Y_i \qquad \text{and} \qquad \bar{Y_c} = \frac{1}{N_c} \sum_{i:T_i=0} Y_i.$$

The indexes $i : T_i = t$ indicate to sum over the units that received a certain treatment level, N_t and N_c denote respectively the number of treated and control units.

The PO framework also provides several solutions to deal with non-experimental or observational data. What usually prevents observational data from being treated as experimental data is the presence of *confounders*. Confounders are variables that affect both the treatment and the outcome and can lead to biased causal estimates if not adequately accounted for. The concern worsens when confounders are unobserved since, in this situation, treatment effects could be impossible to identify. PO methods that deal with observational data aim at emulating an experimental context under specific assumptions. One of these assumptions, that tackles directly the problem of confounders, is called unconfoundedness or ignorability and can be defined as $T_i \perp\!\!\!\perp (Y_i(0), Y_i(1)) | X_i$, where X_i is a set of pre-treatment covariates. Unconfoundedness states that the treatment T_i is independent of the potential outcomes, given a set of pre-treatment variables X_i. The condition allows estimating the ATE as

$$ATE = E[Y_i(1) - Y_i(0)] = E[E[Y_i|T_i = 1, X_i] - E[Y_i|T_i = 0, X_i]]. \tag{1}$$

The formula in Equation 1 is also called adjusting for X and as long as unconfoundedness holds, it ensures an unbiased estimation of the ATE in the presence of confounders. Adjustment can be performed through various methods, including regression, matching and inverse probability weighting. Another PO method to derive causal estimates from observational data is the instrumental variable (IV) strategy (Angrist, 1990). In this context, there is an unobserved variable U, which violates the unconfoundedness assumption for the effect of T on Y. Since U is unobserved, it

is impossible to adjust for it in order to obtain unbiased estimates. However, if the treatment T is affected by another variable Z, it is still possible to estimate a causal effect, under an assumption called *exclusion restriction*. The assumption can be expressed as

$$Y_i(z,t) = Y_i(z',t) \qquad \text{for all } z, z',$$

imposing that potential outcomes do not vary with Z. PO literature refers to variables that satisfy the exclusion restriction as instrumental variables. However, exclusion restriction and unconfoundedness cannot be tested, and they are usually motivated by background theory concerning the causal relations between variables. This implies that justifying them becomes difficult if a priori knowledge is missing. Moreover, as the number of variables in the model increases, assessing the two assumptions' validity turns out to be a challenging task.

The PO framework includes many more identification strategies, such as difference-in-differences, regression discontinuity and synthetic control. For a review of the newest techniques, see Athey and Imbens (2017). These methods provide solutions to very specific causal problems and usually impose additional functional-forms restrictions on probability distributions, such as linearity, monotonicity or additivity.

3. Causal Graphs

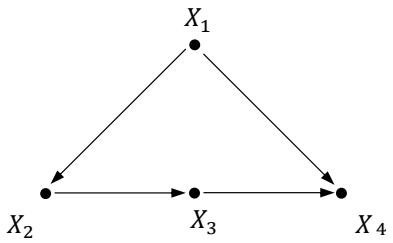
In this section, the Causal Graph framework will be described. First, we will introduce the basic terminology of graphs and the main elements of causal graph theory. Next, we will show how interventions are represented in the framework and how causal effects can be estimated employing graphs.

3.1 Terminology and basic concepts

A *graph* $G = (V, E)$ is a collection of vertices or nodes V and edges E. The edges can be directed or indirected. An edge that goes from a vertex V_i to another vertex V_j is a *directed edge*. Conversely, an edge without such orientation is an *undirected edge*. A graph that only contains directed edges is called a *directed graph*. When two nodes are connected by an edge, they are called *adjacent* nodes. If each pair of nodes belonging to V is connected by an edge, the graph is called a *complete graph*. Conversely, if none of the pairs is adjacent, the graph is an *empty graph*. A sequence of connected edges that starts from a node V_i and ends with node V_j, regardless of the directions of the edges, is called a *path*. In a *directed path* all the edges are oriented in the same direction along the path. A directed path, starting from V_j and ending in V_i, with $V_j = V_i$ is a *cycle*. A directed graph that contains no cycles is also called a *directed acyclic graph* (DAG) (Pearl, 2000). In the context of causal graphs, DAGs are employed to represent causal structures. The vertices of the DAG represent random variables, and its edges describe the causal relations between them. We will refer to variables and vertices in a DAG interchangeably from now on.

Consider the graph G in Figure 1. All the edges in the graph are directed, and they form no cycles; the graph is, therefore, a DAG. G describes the multivariate causal relations between a set of four random variables $\mathbf{X}$. The terminology of kinship is often used to indicate relationships between nodes according to the graph's structure. Since

Figure 1. A simple DAG



the DAG contains a directed edge going from X_1 to X_2, X_1 is called a *parent* of X_2 and the latter is a *child* of X_1. The path p along the ordered sequence of nodes (X_1, X_2, X_3, X_4) is a directed path since all the edges are oriented in the same direction along the path. X_1 is called an *ancestor* of each node belonging to$\{X_2, X_3, X_4\}$ since it precedes them

in p and the vertices in $\{X_2, X_3, X_4\}$ are *descendants* of X_1. Given that the edges are carriers of causal information, we can also say that X_1 is a direct cause of X_2 and X_4. The same is true for every ordered pair of random variables (X_i, X_j) connected by a directed edge that goes from X_i to X_j in the DAG.

Every causal graph also consists of a joint probability distribution $P(\mathbf{X})$ over the variables described by the DAG. This distribution can be factorized according to the structure of the DAG as

$$P(x_1, \ldots, x_n) = \prod_i P(x_i|pa_i), \tag{2}$$

where pa_i indicate the parent set of variable X_i. The factorization implies that given a DAG G with node set $\mathbf{X}$, for each variable $X_i \in \mathbf{X}$, its parent set PA_i selected according to the structure of G, is sufficient for determining the probability of X_i. If a probability function P admits the factorization of Equation 2 relative to a DAG G, then G is said to satisfy the *causal Markov condition* and P is said to be Markov relative to G.

The edges of a DAG can assume specific configurations that provide additional information regarding the independence relations among variables of the model. Given the ordered triplet of nodes (X_i, X_j, X_k), if two directed edges goes from X_i and X_k to X_j but X_i and X_k are not adjacent, then X_j is called a *collider* or unshielded collider in the ordered triplet. Colliders are also referred to as non-emitting nodes. Conversely, given a path p, vertices belonging to p with at least an outgoing edge directed towards other adjacent nodes in p are called *emitting nodes*. An example of a configuration that only contains emitting nodes is when a directed edge goes from node X_j to node X_i, and another directed edge goes from X_j to a third node X_k. This configuration is called *chain*.

A DAG encodes information concerning conditional independence among the variables it represents through a criterion called *d-separation*. Consider a DAG G with node set $\mathbf{X}$, a pair of nodes $\{X_i, X_j\}$ belonging to $\mathbf{X}$ with $X_i \neq X_j$ and a set of nodes $S \subset \mathbf{X}$ not containing X_i and X_j. A path p between X_i and X_j is said to be blocked by a set $\mathbf{S}$ in G, if either

1. p contains at least one arrow-emitting node that belongs to $\mathbf{S}$
2. p contains at least a collision node that does not belong to $\mathbf{S}$ and has no descendent in $\mathbf{S}$.

Two nodes X_i and X_j are said to be *d-separated* given a set $\mathbf{S}$ if all the paths between the nodes are blocked by $\mathbf{S}$. When two nodes X_i and X_j are d-separated by a set $\mathbf{S}$, then X_i is independent of X_j conditional on $\mathbf{S}$. Note that two nodes can also be d-separated conditioning on an empty set if all the paths between them contain at least a collider or its descendants. In this case, the variables represented by the nodes are said to be marginally independent.

3.2 Causal graph analysis at interventional level

Causal graphs allow estimating the effect of interventions, or in other words, the effect of forcing a variable to take a certain value by an external action. Pearl (2000) introduces the *do-operator* $do(X = x)$, a notation to indicate that a variable X is forced by intervention to take value x. In order to be coherent with the terminology defined in Section 2 for the PO framework, we will refer to the effect of a treatment variable T on an outcome variable Y. The do-operator allows writing $P(Y|do(T = t))$ to denote the distribution of Y given an intervention that sets $T = t$. This is different form $P(Y|T = t)$ that instead represents the observational distribution of Y given $T = t$. The causal effect of T on Y can thus be obtained by comparing the quantity $P(Y|do(T = t))$ for different values of t, similarly to what is done in the PO framework where instead $Y(t)$ was the quantity of interest. However, when dealing with non-experimental data, causal effects cannot be estimated directly from data since the interventional distribution of Y is not an observed quantity.

One of the critical contributions of causal graphs is that their structure can serve as a guide to express interventional distributions in terms of observational quantities, thus making it possible to estimate causal effects. This is a crucial result since conditional distributions such as $P(Y|T = t)$, can be directly computed in a non-experimental context through the joint probability distribution associated with the DAG. A graphical condition called *back-door criterion* can be applied to a given causal graph to test if a subset of its nodes $\mathbf{S}$ is sufficient for identifying $P(Y|do(T = t))$ from observational data. A set of variables $\mathbf{S} \subseteq \mathbf{X}$ satisfies the back-door criterion relative to a graph G with node set $\mathbf{X}$, a treatment variable $T \in \mathbf{X}$ and an outcome variable $Y \in \mathbf{X}$ if:

1. no node in $\mathbf{S}$ is a descendant of T; and
2. $\mathbf{S}$ blocks all the paths between T and Y that contain a directed edge pointing towards T.

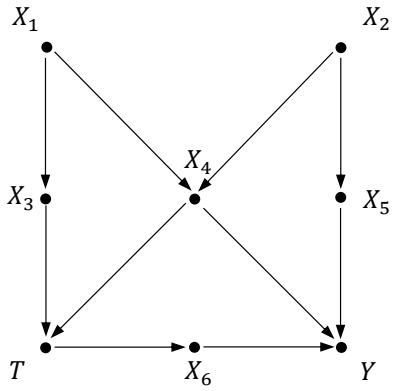
If the back-door criterion is satisfied by a set **S**, then interventional quantities can be expressed through observational ones as follows:

$$P(y|do(T=t)) = \sum_{\mathbf{S}} P(y|t,\mathbf{s})P(\mathbf{s}) \tag{3}$$

The formula used to compute the interventional probability distribution of the outcome in Equation 3 is also known as adjusting for **S**. Summary statistics of the interventional distributions can then be compared to compute the ATE. Obtaining an adjustment set **S** through the back-door criterion also ensures that **S** satisfies the unconfoundedness condition for estimating the effect of T on Y. Therefore, performing a matching procedure (Imbens and Rubin, 2015) by balancing the variable set **S**, would ensure obtaining unbiased estimates of the ATE. This is an example of how Causal graphs can be used as guides for assessing and justifying the assumptions some PO methods require.

Suppose we are interested in estimating $P(y|do(T=t))$ given a causal model represented by the DAG in Figure 2 (Pearl, 2000), with node set $\{\mathbf{X}, T, Y\}$ and a joint probability distribution $P(\mathbf{X}, T, Y)$. The knowledge of the DAG allows the application of the back-door criterion to select an adjustment set for causal effect estimation. The procedure reveals that adjusting for the set $\{X_3, X_4\}$ or $\{X_4, X_5\}$ ensures unbiased estimates of $P(y|do(T=t))$. Conversely, performing the adjustment procedure on a set $\mathbf{S} = \{X_4\}$ would produce biased estimates, since the set does not block all the back-door paths between X and Y.

*Figure 2. A DAG describing causal relations among a set of variables **X** a treatment T and an outcome Y*

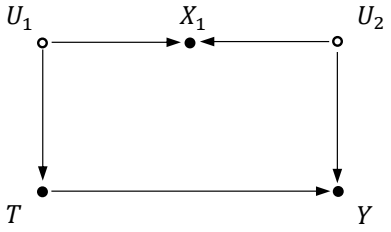


Let us now consider the graph in Figure 3. The DAG shows the presence of two unobserved or latent variables U_1 and U_2. The two nodes are denoted by a circle rather than a solid dot to indicate the variables are not observed. Even in the presence of latent variables, we can resort to the back-door criterion to assess if an adjustment set to estimate the effect of T on Y exists. In this scenario, we are particularly interested in checking if some of the sets that satisfy the back-door criterion are composed only by observed variables. In this situation, adjusting for $\{X_1\}$ would open the back-door path along the ordered tuple (T, U, X_1, X_2, Y), thus producing a biased estimate of the effect of T on Y. Conditioning on the empty set provides instead unbiased estimates of the causal effect, since the colliding path over the ordered triplet (U_1, X_1, U_2) is blocked as long as we do not condition on X_1. The bias introduced by conditioning on X_1 is also called $M - bias$, and it constitutes a solid motivating argument for employing causal graphs. Generally, the PO literature suggests to condition on all the observed pre-treatment variables in order to improve the quality of causal estimates (Imbens and Rubin, 2015). However, in this scenario and similar ones, conditioning on the observed variables leads instead to worse causal estimates, and causal graphs provide a rule, namely the back-door criterion, to avoid this sort of bias. For a review on how conditioning can affect causal estimates, given different contexts represented by causal graphs, see Cinelli et al. (2020).

The back-door criterion is not the only strategy that can be employed to estimate causal effects from a causal graph. (Pearl, 2000) describes a specific graphical configuration that allows causal effect identification, even when back-door adjustment is not feasible. The condition is called *front-door criterion* and states that given a DAG G with node set **X**, a set $\mathbf{S} \subset \mathbf{X}$ satisfies the front-door criterion for the effect of T on Y, both belonging to **X**, if:

1. **S** intercepts all directed paths from T to Y

Figure 3. A DAG with unobserved confounders



2. all the back-door paths from T to **S** are blocked
3. all the back-door paths from **S** to Y are blocked by T

If a set **S** that satisfies the front-door criterion for the effect of T on Y exists and $P(t, s) > 0$, then the causal effect of T on Y can be computed with the formula

$$P(y|do(T = t)) = \sum_s P(s|t) \sum_{t'} P(y|t', s)P(t'). \tag{4}$$

Combined and iterative use of back-door and front-door criterion constitute the building block to identify causal effects on complex DAGs. Pearl (2000) describes a set of rules based on the two criteria, also called *do-calculus*, that allows expressing interventional distributions in terms of observational distributions only, in an automated way. The procedure has been proved to be sound and complete, meaning that an algorithmic iteration of the rules of do-calculus always return a solution for the identification of causal effects, if such solution exists (Pearl, 2000; Tian and Pearl, 2002; Huang and Valtorta, 2006).

4. Causal discovery

Causal graphs are powerful models to describe the causal structure of a set of random variables. Moreover, they constitute a guide for selecting an identification strategy to estimate causal effects. However, the setting considered here always assumed a complete knowledge of the causal diagram.

Suppose we want to investigate the causal effect of a treatment variable T on an outcome variable Y from a dataset $D(\mathbf{X}, T, Y)$ where $\mathbf{X}$ is a set of other covariates. We also assume the existence of an unknown underlying causal model described by a DAG $G(V, E)$ and a joint probability distribution $P(V)$, from which $D(\mathbf{X}, T, Y)$ has been sampled. In order to obtain an unbiased estimate of $P(Y|do(T = t))$ we therefore study if it is possible to *learn* a causal graph from $D(\mathbf{X}, T, Y)$. In order to estimate the structure of the causal DAG, *structural learning algorithms* have been developed. These algorithms take a dataset as an input and, under a set of assumptions, recover a DAG and the associated joint probability distribution. This process is known as *causal discovery* (Spirtes et al., 2000).

Structural learning algorithms can be divided in three families: *constraint-based* algorithms, *score-based* algorithms and *hybrid* algorithms. Constraint-based algorithms learn the graph's structure via conditional independence statements emerging from data. They usually start with a complete graph, and then if two variables turn out to be marginally or conditionally independent, the edge connecting them is deleted. This procedure is repeated iteratively until a stopping criterion is satisfied. Score-based algorithms rely on a given score function that measures how well a certain DAG describes a dataset. These algorithms usually begin by computing the score of an initial graph. The diagram is then modified by introducing, deleting or reversing edges, and its score is computed again for each modification. The graph recording the best score at the end of the procedure is retained as the algorithm's output. Hybrid algorithms aim to exploit the advantages of score-based and constraint-based algorithms by merging them in a single procedure. Generally, they begin with a *restrict* phase where the parents of each node are selected through tests of conditional independence, similarly to what happens in constraint-based algorithms. The second phase is called *maximize* and consists in selecting a DAG in the restricted DAG family outlined by phase one by optimizing a given score

function. Hybrid algorithms include the Max-Min Hill Climbing (Tsamardinos et al., 2006) and $H2PC$ (Gasse et al., 2014). Once the graph is learnt, a joint probability distribution over the nodes of the graph can be obtained through maximum likelihood estimation. This phase usually involves computing maximum likelihood estimates subject to the independence constraints encoded in the graph. Estimates can be retrieved in the case of discrete variables or when dealing with continuous variables under the assumption of linearity (Spirtes et al., 2000).

The section will continue with a description of the assumptions that structural algorithms usually require. We will then explain how different algorithms work and show the functioning of two representative procedures.

4.1 Common assumptions and background knowledge

The assumptions of causal discovery algorithms usually focus on the relation between the causal graph and the distribution employed to learn it. A usually required assumption is *faithfulness*. A graph G faithfully represents a dataset D, if and only if all and only the list of d-separations emerging from D are true in G. This ensures an exact correspondence between conditional independence relations of the distribution from which the graph is learnt and those entailed by the causal Markov condition applied to G. Another key assumption for learning algorithms is *causal sufficiency*. The assumption states that a given set of variables $\mathbf{X}$ is causally sufficient for a population if and only if in the population every common cause of any two or more variables belonging to $\mathbf{X}$ is in $\mathbf{X}$ or has the same value for all units in the population. Implementing a constraint-based algorithm also requires making statistical decisions concerning how to assess conditional independence. Several tests can be employed to check if conditional independence holds, and violations of the assumptions required by the tests can generate unreliable independence statements. For a review of the implications of choosing a given independence test and what happens when the required assumptions do not hold, see Spirtes et al. (2000).

Structural learning algorithms are usually employed when information concerning the causal graph is not available. However, in practice it is common to deal with scenarios where the knowledge of the causal graph is partial. This incomplete knowledge can be introduced in structural learning procedures by imposing constraints on the structure of the obtained network. For example, if is known that a variable X_i cannot cause a second variable X_j, the directed edge that goes from X_i to X_j is forced to be absent. Note that this constraint does not imply the presence or absence of a directed edge going from X_j to X_i. Conversely, if background knowledge suggests that X_i affects X_j, a directed edge from X_i to X_j can be imposed. A consequence of including previous knowledge in the learning phase is that the graph is not entirely obtained through the information contained in the data. The constraints on the structure of the graph restrict the search space of the algorithms and often reduce both uncertainty and computational time.

4.2 Constraint-based algorithms

Constraint-based algorithms learn causal graphs from conditional independence relations contained in the data. They can usually take both discrete and linear continuous data as input: in the first case, the algorithm performs conditional independence tests on cell counts; in the latter, covariance matrices are used to test vanishing partial correlations. The obtained conditional independence statements, if possible, are then translated into graphical form according to the rules of d-separation. Constraint-based algorithm include the PC algorithm(Glymour et al., 1991), the IG algorithm (Verma and Pearl, 1990) and the most recent Grow-Shrink algorithm (Margaritis, 2003). All the algorithms share the idea of learning a graph from the independence structure of the data but employ different heuristics. Constraint-based algorithms generally assume causal sufficiency, namely observing all the common causes of two or more variables in the model. This is a strong assumption, difficult to achieve in observational contexts. Some constraint-based algorithms have been proposed to deal with models where causal sufficiency does not hold. One of the most used is the fast causal inference (FCI) algorithm (Spirtes et al., 2000). The algorithm is a variation of the PC algorithm and retrieves asymptotically correct causal structures in the presence of latent common causes, provided the observed distribution and the graph satisfy the faithfulness condition.

One of the most used algorithms in the constraint-based family is the *PC Algorithm*. The procedure begins with a complete undirected graph, in which edges are progressively deleted when they describe relations between variables that are found to be conditionally independent. Faithfulness and causal sufficiency are assumed. A pseudocode of a recent variation of the algorithm, called *PC-stable* (Colombo and Maathuis, 2014) is displayed in Algorithm 1. In the original PC algorithm the obtained graph could be affected by the ordering of the variables in the dataset used to learn

the graph. In the new version, instead, the ordering does not affect the results, thus the name PC-stable. The procedure begins by learning a graph containing only undirected edges from conditional independence statements retrieved from the dataset. Then the orientation of the edges is determined according to a set of graphical rules. In the pseudocode we will denote directed and undirected edges between to nodes X_i and X_j, respectively with the notation $X_i \rightarrow X_j$ and $X_i - X_j$. Moreover we will use $adj(X_i)$ to denote the set composed by the nodes adjacent to X_i and $\mathbf{X}\backslash\{X_i\}$ to indicate the variable set $\mathbf{X}$ excluding variable X_i.

Algorithm 1: PC-stable

Input: A sample $D = (\mathbf{X})$ from a set of random variables $\mathbf{X} = \{X_1, ..., X_N\}$ and a chosen statistical test of conditional independence
Output: A family of Markov-equivalent DAGs

1 Form a complete undirected graph G with vertex set $\{X_1, ..., X_N\}$;
2 Set $l = -1$;
3 **repeat**
4 $l = l + 1$;
5 **forall** *vertices* X_i *in* G **do**
6 Set $a(X_i) = adj(G, X_i)$
7 **end**
8 **repeat**
9 select a (new) adjacent pair of nodes $(X_i, X_j), i \neq j$ in G such that $|a(X_i)\backslash X_j| \geq l$;
10 **repeat**
11 Choose a (new) set $\mathbf{S} \subseteq a(X_i)\backslash\{X_j\}$ of size l;
12 **if** *the statistical test reveals that* X_i *is conditionally independent from* X_j *given* $\boldsymbol{S}$ **then**
13 delete the edge connecting the pair (X_i, X_j) from G;
14 set $\mathbf{S}_{X_i X_j} = \mathbf{S}$, denoting the set that separates X_i and X_j
15 **end**
16 **until** X_i *and* X_j *are no longer adjacent in* G *or all possible subsets* $\boldsymbol{S}$ *of size* l *have been considered*;
17 **until** *all pairs of adjacent nodes* $(X_i, X_j), i \neq j$ *in* G *such that* $|a(X_i)\backslash\{X_j\}| \geq l$ *have been considered*;
18 **until** *all pairs of adjacent nodes* (X_i, X_j) *in* G *satisfy* $|a(X_i)\backslash\{X_j\}| \leq l$;
19 **foreach** *triplet* $\{X_i, X_k, X_j\}$ *such that* X_i *is adjacent to* X_k*, the latter is adjacent to* X_j*, but the pair* $\{X_i, X_j\}$ *is not adjacent to* X_j*, if* $X_k \notin \boldsymbol{S}_{X_i, X_j}$ **do**
20 orient $X_i - X_k - X_j$ with the colliding configuration $X_i \rightarrow X_k \leftarrow X_j$.
21 **end**
22 Set more arc directions by repeated application the following rules:
23 **if** X_i *is adjacent to* X_j *and there is a directed edge from* X_i *to* X_j **then**
24 replace X_i - X_j with $X_i \rightarrow X_j$
25 **end**
26 **if** *there are two paths* $X_i - X_k \rightarrow X_j$ *and* $X_i - X_l \rightarrow X_j$ *and* X_k *is not adjacent to* X_l *and there is a directed edge from* X_i *to* X_j **then**
27 replace X_j - X_k with $X_j \rightarrow X_k$
28 **end**
29 **if** X_i *and* X_k *are not adjacent but* $X_i \rightarrow X_j$ *and* X_j - X_k **then**
30 replace $X_j - X_k$ with $X_j \rightarrow X_k$
31 **end**

Given a dataset $D = (\mathbf{X})$ describing a set of random variables $\mathbf{X} = \{X_1, ..., X_N\}$, the PC-stable algorithm begins by forming a complete undirected graph G over $\mathbf{X}$. Then, step 5 stores the adjacency sets $adj(G, X_i)$ for each node X_i according to the current structure of G. Given an index l which start from 0 and increase at each iteration, the procedure checks if a set $\mathbf{S}$ of size l, that d-separates two nodes X_i and X_j exists. Note that $\mathbf{S}$ must be formed by nodes belonging to $adj(G, X_i)$ obtained in step 5 and that the size of $adj(G, X_i)\backslash X_j$ must be greater or equal than l. If the procedure finds a set $\mathbf{S}$ of size l that makes X_i and X_j conditionally independent, the edge between them is deleted from G and $\mathbf{S}$ is retained. The procedure is repeated for every node pair (X_i, X_j) and for every possible size l $\mathbf{S}$ associated to it, until an $\mathbf{S}$ that ensures d-separation is found or every $\mathbf{S}$ of size l has been explored. The algorithm then increases l by a unit and repeats the procedure from step 5, until every pair of adjacent nodes (X_i, X_j) in G satisfies $|a(X_i)\backslash\{X_j\}| \leq l$. In other words, at each iteration, the structure of G is updated by removing edges between conditional independent variables. Once the undirected graph is obtained, steps 19-31 orient the edges according to specific edge configurations. The rules dictated by the algorithm ensure that cycles are not generated and avoid the creation of a new colliding configuration that would modify the conditional independence relations.

The output of the PC-stable algorithm is a *completed partially DAG*(CPDAG), a DAG where some of the edges are undirected. This kind of graph is used to represent a family of independence-equivalent DAGs. Regardless of how the undirected edges of the graph are oriented, the colliding configurations remain the same, thus ensuring that

the all the DAGs associated to a CPDAG encode the same conditional independencies. The output of the PC-stable algorithm is therefore coherent with the objective of translating the conditional independences contained in the data into graphical form. Moreover, it has been proven that, if the assumptions hold, the results provided by the algorithm are sound and complete (Colombo and Maathuis, 2014).

4.3 Score-based algorithms

Score-based algorithms aim at recovering the graph structure from data by optimizing a score function. Generally, this kind of algorithm explores several graph structures and assigns a score to each of them; at the end of the procedure, the graph with the maximal score is retained. Score-based algorithms usually assume faithfulness as well as causal sufficiency. Algorithms belonging to this family include the *greedy search*, the *simulated annealing* and *genetic algorithms* (Russell and Norvig, 2009).

The *greedy search* is one of the most used score-based algorithms, and its steps are shown in the pseudocode of Algorithm 2. The procedure iteratively modifies the edges of an initial DAG, computes the score of each graph and retains the best-scoring structure. When the score does not increase with an iteration, the obtained graph is provided as the algorithm's output.

Algorithm 2: Greedy Search

Input: A sample $D = (\mathbf{X})$ from a set of random variables $\mathbf{X} = \{X_1, ..., X_N\}$ a score function $\mathcal{F}(G, D)$
Output: A DAG

1 Form an empty graph G with vertex set $\{X_1, ..., X_N\}$;
2 Calculate the score of G given D, $S_G = \mathcal{F}(G, D)$;
3 Set $S_{max} = S_G$;
4 Set $G_{max} = G$;
5 **repeat**
6 **foreach** *possible edge addition, removal or inversion in* G_{max} *that produces a modified DAG* $G^\star$ **do**
7 compute $S_{G^\star} = \mathcal{F}(G^\star, D)$;
8 **if** $S_{G^\star} > S_{max}$ *and* $S_{G^\star} > S_G$ **then**
9 set $G = G^\star$ and $S_G = S_{G^\star}$
10 **end**
11 **end**
12 **if** $S_G > S_{max}$ **then**
13 set $S_{max} = S_G$ and $G_{max} = G$
14 **end**
15 **until** S_{max} *of current iteration is smaller then* S_{max} *of previous iteration*;

Given a dataset $D = (\mathbf{X})$ and a score function $\mathcal{F}(G, D)$, the algorithm first two steps consist in computing the score of an initial, usually empty, graph G with vertex set $\mathbf{X}$. Next, the score of the graph is set as the maximal score S_{max} and the initial graph G is set as the best-scoring DAG G_{max}. In step 6, the best-scoring DAG is modified by deleting, adding or inverting an edge, thus generating a new DAG $G^\star$. The score of $G^\star$ is computed, and if it is greater than the best score of the iteration S_G and greater than the absolute best score S_{max} then $G^\star$ becomes the new best score of the iteration S_G. All the possible modifications to G_{max} are explored this way, and if the best-obtained score of the iteration is greater than the best absolute score, then the latter is set to the current S_G and G_{max} is set equal to the current G. The procedure is then repeated from step 6 for the new G_{max}. The algorithm stops when applying all the possible modifications to the DAG G_{max}, obtained in the previous iteration, does not generate an increased S_{max}. In this case, G_{max} constitutes the output of the algorithm.

4.4 Causal discovery and potential outcomes

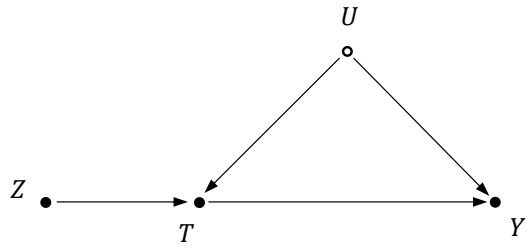
We have already shown how PO methods can benefit from specifying a causal graph to outline causal relations between variables. If the causal knowledge is available, drawing a causal graph can help assess unconfoundedness in a high dimensional context, using a graphical condition called the back-door criterion. Causal discovery methods constitute an additional resource if we are interested in estimating causal effects with PO methods, but the knowledge of the causal graph is partial or absent. Suppose we want to estimate the effect of treatment T on an outcome Y given a dataset $D(\mathbf{X}, T, Y)$, where $\mathbf{X}$ are additional random variables, that could directly or indirectly affect T and Y. In addition, let us assume that the available subject matter knowledge concerning the variable causal structure is very limited and thus does not allow drawing a causal graph. In order to estimate causal effects with a PO method such

as matching, we have first to assess if unconfoundedness holds. However, since the causal graph over $\{\mathbf{X}, T, Y\}$ is unknown, we cannot directly select an adjustment set $\mathbf{S}$ that satisfies the back-door criterion.

Causal discovery provides a solution to this scenario. If we cannot exclude the absence of unobserved common causes, we can learn the graph from $D(\mathbf{X}, T, Y)$ employing an algorithm that only requires the faithfulness assumption, such as the FCI algorithm. The algorithm's output can be then used to assess which PO identification strategy is adequate to estimate the causal effect of T on Y. If instead, it is reasonable to assume both causal sufficiency and faithfulness, we can opt for an algorithm such as the greedy search or PC-stable. In both cases, we know that if the assumptions hold, the obtained causal structures are asymptotically correct, and a sufficient adjustment set can be selected by applying the back-door criterion. The adjustment set can then be used to derive the interventional distribution through the adjustment formula, or directly estimate the ATE with a method of choice, such as regression, matching or inverse probability weighting.

Alternatively, learning the graph from data could reveal or confirm if a specific PO identification strategy is feasible. Assume that applying a structural learning algorithm on a given dataset generates the DAG in Figure 4. If

Figure 4. Instrumental variable DAG



we are interested in the effect of T on Y, we cannot directly estimate causal effects because of the presence of the unobserved confounder U, and no observed adjustment set that satisfies the back-door criterion. However, the graph configuration reveals that variable Z satisfies the exclusion restriction assumption of instrumental variables described in Section 2. This means that we can employ an IV strategy to achieve causal effect identification. Also in this case, the assumptions required by PO methods are made transparent by causal graph implementation. In this particular example, those assumptions are also strengthened by the structural learning procedure that allows exclusion restrictions to be derived directly from the data.

5. Discussion

Estimating causal effects is a central subject for biomedical and social sciences. However, investigating causal claims is an ambitious objective, especially when dealing with observational data. The most affirmed causality frameworks are Potential Outcomes and Causal Graphs. The two approaches are often contrasted to evaluate which one is most effective. PO methods offer efficient ad hoc solutions to specific causal problems. However, their assumptions are considered difficult to assess, especially as the number of variables increases. On the other hand, causal graphs allow the formalization of complex causal problems in a generalized way. Nevertheless, their high generality can sometimes be perceived as a distance from real empirical problems and incapacity of including context-specific restrictions in the model.

This paper described how the two frameworks could be implemented together in an integrated approach. Causal graphs can be used as a guide to evaluating which PO method can be implemented and if its assumptions hold. The graph can be outlined directly if the causal structure is entirely known or learned from data if the causal knowledge is partial or absent. This versatility guarantees coverage of most empirical problems. The results of PO methods are thus strengthened by causal graphs, since assumptions such as unconfoundedness and exclusion restrictions can be directly assessed from the structure of the DAG. At the same time causal graphs can benefit from all the context-specific identification strategies provided by the literature of potential outcomes. Combining the two methodologies thus results in an effective synergic approach that enhances both frameworks' peculiar characteristics.

References

ANGRIST, J. D. (1990): Lifetime Earnings and the Vietnam Era Draft Lottery: Evidence from Social Security Administrative Records, *The American Economic Review*, 313–336.

ANGRIST, J. D. AND J.-S. PISCHKE (2008): *Mostly Harmless Econometrics*, Princeton university press.

ATHEY, S. AND G. W. IMBENS (2017): The State of Applied Econometrics: Causality and Policy Evaluation, *Journal of Economic Perspectives*, 31, 3–32.

BAREINBOIM, E. AND J. PEARL (2016): Causal Inference and the Data-Fusion Problem, *Proceedings of the National Academy of Sciences*, 113, 7345–7352.

CINELLI, C., A. FORNEY, AND J. PEARL (2020): A Crash Course in Good and Bad Controls, *SSRN Journal*.

COLOMBO, D. AND M. H. MAATHUIS (2014): Order-Independent Constraint-Based Causal Structure Learning. *J. Mach. Learn. Res.*, 15, 3741–3782.

COX, D. R. (1958): Planning of Experiments. .

CUNNINGHAM, S. (2021): *Causal Inference: The Mixtape*, Yale University Press.

FISHER, R. A. (1949): The Design of Experiments, .

GASSE, M., A. AUSSEM, AND H. ELGHAZEL (2014): A Hybrid Algorithm for Bayesian Network Structure Learning with Application to Multi-Label Learning, *Expert Systems with Applications*, 41, 6755–6772.

GLYMOUR, C., P. SPIRTES, AND R. SCHEINES (1991): Causal Inference, *Erkenntnis*, 35, 151–189.

HAAVELMO, T. (1943): The Statistical Implications of a System of Simultaneous Equations, *Econometrica, Journal of the Econometric Society*, 1–12.

HOLLAND, P. W. (1986): Statistics and Causal Inference, *Journal of the American statistical Association*, 81, 945–960.

HUANG, Y. AND M. VALTORTA (2006): Pearl's Calculus of Intervention Is Complete, in *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence*, 217–224.

HÜNERMUND, P. AND E. BAREINBOIM (2019): Causal Inference and Data Fusion in Econometrics, *arXiv preprint arXiv:1912.09104*.

IMBENS, G. W. (2020): Potential Outcome and Directed Acyclic Graph Approaches to Causality: Relevance for Empirical Practice in Economics, *Journal of Economic Literature*, 58, 1129–1179.

IMBENS, G. W. AND D. B. RUBIN (2015): *Causal Inference in Statistics, Social, and Biomedical Sciences*, Cambridge University Press.

MARGARITIS, D. (2003): Learning Bayesian Network Model Structure from Data, Tech. rep., Carnegie-Mellon Univ Pittsburgh Pa School of Computer Science.

MORGAN, S. L. AND C. WINSHIP (2015): *Counterfactuals and Causal Inference*, Cambridge University Press.

NEYMAN, J. AND K. IWASZKIEWICZ (1935): Statistical Problems in Agricultural Experimentation, *Supplement to the Journal of the Royal Statistical Society*, 2, 107–180.

PEARL, J. (2000): *Models, Reasoning and Inference*, vol. 19.

ROSENBAUM, P. (2018): *Observation and Experiment*, Harvard University Press.

RUBIN, D. B. (1974): Estimating Causal Effects of Treatments in Randomized and Nonrandomized Studies. *Journal of educational Psychology*, 66, 688.

RUSSELL, S. AND P. NORVIG (2009): *Artificial Intelligence: A Modern Approach. Prentice Hall*.

SPIRTES, P., C. N. GLYMOUR, R. SCHEINES, AND D. HECKERMAN (2000): *Causation, Prediction, and Search*, MIT press.

SPLAWA-NEYMAN, J., D. M. DABROWSKA, AND T. P. SPEED (1990): On the Application of Probability Theory to Agricultural Experiments. Essay on Principles. Section 9. *Statistical Science*, 5, 465–472.

TIAN, J. AND J. PEARL (2002): A General Identification Condition for Causal Effects, in *Aaai/Iaai*, 567–573.

TINBERGEN, J. (1930): Determination and Interpretation of Supply Curves: An Example, *Zeitschrift fur Nationalokonomie*, 1, 669–679.

TSAMARDINOS, I., L. E. BROWN, AND C. F. ALIFERIS (2006): The Max-Min Hill-Climbing Bayesian Network Structure Learning Algorithm, *Machine learning*, 65, 31–78.

VERMA, T. AND J. PEARL (1990): Causal Networks: Semantics and Expressiveness, in *Machine Intelligence and Pattern Recognition*, Elsevier, vol. 9, 69–76.

WRIGHT, S. (1921): Systems of Mating. I. The Biometric Relations between Parent and Offspring, *Genetics*, 6, 111.