# ABSTRACT

Title of dissertation: CAUSAL PROGRAMMING

Joshua Brulé
Doctor of Philosophy, 2019

Dissertation directed by: Professor James A. Reggia
Department of Computer Science

Causality is central to scientific inquiry. There is broad agreement on the meaning of causal statements, such as "Smoking causes cancer", or, "Applying pesticides affects crop yields". However, formalizing the intuition underlying such statements and conducting rigorous inference is difficult in practice. Accordingly, the overall goal of this dissertation is to reduce the difficulty of, and ambiguity in, causal modeling and inference. In other words, the goal is to make it easy for researchers to state precise causal assumptions, understand what they represent, understand why they are necessary, and to yield precise causal conclusions with minimal difficulty.

Using the framework of structural causal models, I introduce a causation coefficient as an analogue of the correlation coefficient, analyze its properties, and create a taxonomy of correlation/causation relationships. Analyzing these relationships provides insight into why correlation and causation are often conflated in practice, as well as a principled argument as to why formal causal analysis is necessary. Next, I introduce a theory of causal programming that unifies a large number of previ-

ously separate problems in causal modeling and inference. I describe the use and implementation of a causal programming language as an embedded, domain-specific language called 'Whittemore'. Whittemore permits rigorously identifying and estimating interventional queries without requiring the user to understand the details of the underlying inference algorithms. Finally, I analyze the computational complexity in determining the equilibrium distribution of cyclic causal models. I show this is uncomputable in the general case, under mild assumptions about the distributions of the model's variables, suggesting that the structural causal model focus on acyclic causal models is a 'natural' limitation. Further extensions of the concept will have to give up either completeness or require the user to make additional — likely parametric — model assumptions.

Together, this work supports the thesis that rigorous causal modeling and inference can be effectively abstracted over, giving a researcher access to all of the relevant details of causal modeling while encapsulating and automating the irrelevant details of inference.

# CAUSAL PROGRAMMING

by

Joshua Brulé

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2019

Advisory Committee:
Professor James A. Reggia, Chair/Advisor
Professor Dana S. Nau
Professor Mihai Pop
Professor William Gasarch
Professor Derek A. Paley

# Dedication

To Mom and Dad, for encouraging me to ask difficult questions.

# Acknowledgments

This research was supported in part by ONR Award N00014-19-1-2044.

I am fortunate have many people to thank for their help and support during my time as a student.

My advisor, James Reggia, was instrumental in helping me learn how think like a scientist, balancing open-mindedness to new ideas and skepticism of claims that seem too strong to be true. He supported and encouraged me, even at times when I couldn't clearly explain what it was that I was doing, and worked to maximize the amount of time I had available to conduct research.

Each of my committee members had a crucial role in helping me develop as a researcher. Derek Paley was my first-ever advisor during a summer research experience for undergraduates, which was an incredibly rewarding experience and a major factor in my decision to apply to graduate school. Bill Gasarch supervised my first independent research project and helped me cultivate a playful attitude towards mathematics and theoretical computer science. Dana Nau taught the first class I took in artificial intelligence and did a remarkable job answering my incessant questions. Mihai Pop invited me to give a talk to his research group, and helped me gain a deep appreciation for the messy nature of real-world problems. And Rance Cleaveland helped me gain a deep appreciation for the beauty and power of formal methods and programming languages.

Ilya Shpitser and Elias Bareinboim both reached out to me and invited me to visit them and discuss research in causal inference. Each conversation with them

probably shaved months off of the time it took to finish my PhD.

Kris Micinski, Brendan Good and Alex Eftimiades repeatedly reviewed early drafts of my papers, a thankless task for which I am forever grateful.

All my friends and family made graduate school tolerable when the stress of research threatened to become to much to bear.

Finally, thanks to Mom and Dad for their unconditional love and support. None of this would have been possible without you.

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1: Introduction

> Causality is simple.
>
> —Judea Pearl

> I have been studying this stuff for over a decade now, and it is often not obvious to me.
>
> —Ilya Shpitser

## 1.1 Causality

Causality is central to scientific inquiry. Unsurprisingly, there is an enormous literature on the topic, with key contributions from philosophy [48], economics [34], statistics [72], genetics [89], artificial intelligence [57], and other disciplines. There is broad agreement on the meaning of statements such as, "Smoking causes cancer", or, "Applying pesticides affects crop yields", but the intuition underlying such statements is often difficult to formalize.

Causality is implicit in ordinary language [8], which makes it easy to introduce unwarranted assumptions, or fail to introduce necessary assumptions in analysis. In practice, this leads to ambiguity in modeling and inference. Non-experimental research is often presented as "indicative of" or "suggesting" causality. However,

outside of randomized, controlled experiments, it is not uncommon for researchers to disagree on what "causality" even means. When faced with the difficulty of rigorous causal modeling and inference, without the assumption of randomization, many researchers choose to make no causal claims at all.

Accordingly, the overarching goal of this dissertation is to reduce the ambiguity in, and difficulty of, causal modeling and inference; in other words, to make it easy for researchers to state precise causal assumptions, understand what they represent, understand why they are necessary, and to yield precise causal conclusions with minimal difficulty. Although concerned with all aspects of causal inference, this dissertation primarily focuses on predicting the effect of interventions from non-experimental data (i.e. data *not* drawn from a randomized, controlled trial), which is impossible without making additional assumptions. This dissertation aims to make these assumptions easy to state and understand, while being amenable to automated analysis.

What this goal entails may not be immediately clear; I claim that one of the reasons that "causality" is ambiguous in practice is because it is used to refer to several distinct, but closely related concepts. To clarify this point, I make frequent reference to two conceptual hierarchies that are useful in guiding understanding and analysis of causality: the Causal Hierarchy [74] and the Heckman Hierarchy [33].

## 1.2 The Causal Hierarchy and Heckman Hierarchy

The Causal Hierarchy, also referred to as the "Ladder of Causation" (Figure 1.1), distinguishes between the types of queries that can be made in analysis:

1. Associational / statistical, e.g. "If we *observe* that this patient has a particular symptom, will they recover?"

2. Interventional / causal, e.g. "If we *treat* this patient, will they recover?"

3. Counterfactual / hypothetical, e.g. "Given that this patient was treated, *had* they not been treated, *would* they have recovered?"

The Causal Hierarchy is a hierarchy in the sense that these are successively more general classes of queries. Associational queries are limited to observations from a single model; observing a patient's symptoms may provide information, but does not affect any outcome. Interventions *change* the original model; prescribing a treatment may cause the patient to recover, because it changes the factors that determine the health of the patient. Counterfactual queries are the most general, and can consider multiple possible worlds. The question of whether a patient who, in fact, recovered *would have* recovered, had they not been treated involves intervention (the original treatment), observation (the patient recovered), and another hypothetical world (where the patient was not treated). These alternative "possible worlds" or "potential outcomes" are commonly called counterfactuals because they may — but are not required to — include conditions contrary to fact.

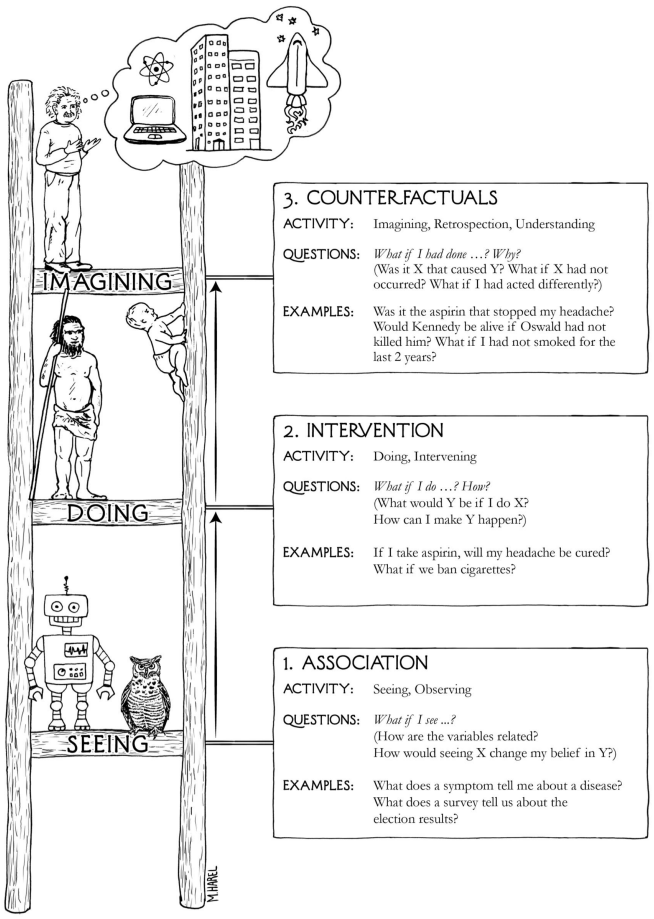



Figure 1.1: The Ladder of Causation [61], also called the Causal Hierarchy. Each level represents successively more general questions, which require more sophisticated modeling and inference to answer.

The Heckman Hierarchy (Table 1.1) distinguishes between the types of tasks in causal inference:

1. Definition of the set of counterfactuals (potential outcomes)

2. Identification from population distributions

3. Selection given actual data

The Heckman Hierarchy is a hierarchy in the sense that each task depends on the assumptions made in the previous task. The first task requires a scientific theory

| Task | Description | Requirements |
|---|---|---|
| 1 | Defining the set of counterfactuals | A scientific theory |
| 2 | Identifying parameters from population distributions | Mathematical analysis of point or set identification |
| 3 | Selecting models from actual data | Estimation and testing theory |

Table 1.1: The Heckman Hierarchy, adapted from [33]

to define the set of potential outcomes and provide rules for determining them. In other words, potential outcomes are generated by a function of their determining factors and manipulating these factors may generate different outcomes. For example, classical mechanics permits modeling an object's trajectory as a function of its initial velocity. With respect to the assumptions in the model, it becomes possible to predict alternative trajectories, had the initial conditions been different. Science is based on constructing and testing such models, whether they are based on the laws of physics, biological assumptions, or the expected utility hypothesis, to name a few examples. It is only meaningful to treat something as a cause if it could have — at least, in principle — been different and produced a different outcome.

With respect to the assumptions in a scientific theory, it becomes possible to analyze problems of identification. For example, an elementary model of mechanics may be parameterized by $g$, the acceleration imparted to objects due to Earth's gravity. The data gathered from any actual experiment to determine $g$ will be subject to random errors. However, a well-designed experiment, combined with proper analysis of the data will permit identifying $g$ in the limit of infinite data samples. In other words, identification requires finding unique mappings from population measures to model parameters.

Finally, model selection is the problem of inference in practice. This task lies in the domain of estimation and hypothesis testing theory. It is the question of what may be reasonably concluded from the actual data available, i.e. real world samples subject to sampling variation. For example, depending on their particular goal, a researcher may wish to accept or reject a particular hypothesis at some level of confidence, or to calculate point or interval estimates of a parameter. Effective model selection depends on the analysis of identification; a consistent estimator is one that is guaranteed to converge to the true value in the limit.

The different levels of the Causal and Heckman Hierarchies are easily conflated in practice. For example, propensity score matching is a method to estimate average treatment effect from non-experimental data. The method works by approximating estimands of the form $\sum_s P(y \mid s, x)P(s)$ over a high dimensional $S$, by calculating $\sum_l P(y \mid l, x)P(l)$, over a one dimensional $L$ [69] [57]. This is an estimator (task 3 in the Heckman Hierarchy) for a statistical quantity (level 1 in the Causal Hierarchy). With respect to certain model assumptions (task 1), this quantity is asymptotically equal to the average treatment effect of $X$ on $Y$ (level 2). In this sense, propensity score matching solves an identification problem (task 2).

Unfortunately, all of these conceptually separate concerns are considered part of the same method, leading to ambiguity. When the necessary model assumptions hold, propsensity score matching accurately estimates treatment effect; informally, if the necessary assumptions hold, the resulting estimate is 'as good as' an estimate obtained from experimental data. However, propensity score matching is generally performed without explicitly specifying a causal model, making it difficult to judge

if the model assumptions are reasonable for the problem at hand. In the best case, this is an attempt to conduct inference with minimal assumptions. In the worst case, it is 'blind' empiricism. Science is based on constructing and testing models and a scientific theory of causality must do the same. Data analysis, alone, leads nowhere without theory to guide it.

## 1.3 Research goal

In computer science, the essence of effective abstraction is to preserve all of the relevant details in a given context, while hiding/encapsulating the irrelevant details. Abstractions can be made at a mostly mathematical/conceptual level; for example, models of computation, such as the Turing machine or lambda calculus, capture the essential notion of computation, abstracting over details of computer architecture. Abstractions also refer to concepts designed to be implemented in code; for example, garbage collection hides the problem of memory management from a programmer by automatically allocating and freeing memory.

The work in this dissertation builds on the existing theory of structural causal models (SCMs). An SCM is a system of manipulable/modifiable equations that generate the set of potential outcomes (counterfactuals). This can be seen as the nonparametric generalization of the structural equation modeling (SEM) approach that is dominant in econometrics [1], as well as providing semantics for the Neyman-Rubin-Holland potential outcomes approach that is dominant in statistics and epidemiology [38]. SCMs are also closely related to probabilistic graphical models,

specifically, Bayesian networks. Probabilistic graphical models are statistical models (level 1), but the causal diagrams associated with SCMs can be seen as the causal generalization. In this dissertation, I consider structural causal models as a foundational theory of causality and present work to support the following thesis:

*Rigorous causal modeling and inference can be effectively abstracted over, giving a researcher access to all of the relevant details of modeling, while encapsulating and automating the details of causal inference.*

The following specific objectives guide this work:

- **Justify and clarify the necessity of formal causal analysis.** The maxim "Correlation is not causation" has reached the status of statistical cliche. However, it is generally accepted that correlation is indicative of causation, without being conclusive evidence of such. This suggests the following goal: determine in what sense correlation and causation are related by devising a means by which they can be directly compared. Specifically, design a taxonomy of correlation/causation relationships and visualize how these relationships may occur in practice.

- **Develop a (meta-)theory of causal programming.** Essentially, design another axis of abstraction, similar in scope to the Causal Hierarchy and Heckman Hierarchy, covering the types of problems in causal modeling and inference. Specifically, develop a set of abstractions that permit capturing a wide variety of problems of interest, unifying them into a single theoretical framework. This theoretical framework should provide a means to separate

the definition of a causal inference problem from the methods used to solve it, unify existing methods, and remain open to future extension.

- **Demonstrate the feasibility of causal programming.** Specifically, implement a causal programming language powerful enough to perform identification and estimation of interventional queries. Such a language should permit a researcher to declare causal models and queries with syntax similar to the underlying mathematics, and automatically perform inference. In addition, the implementation should support interactive 'notebook' usage to make it easy to iteratively refine models while viewing results, as well as remaining open to future extension.

- **Understand the limitations of the theory.** The focus of this dissertation is limited to nonparametric, recursive (acyclic) causal models. A related goal is to determine in what sense this is a natural limitation. Specifically, prove that the equilibrium distribution of cyclic causal models is, in general, uncomputable. From this, I argue that any further generalization is 'fundamentally' difficult and will likely need to give up either completeness or nonparametricity.

## 1.4 Overview

The rest of this dissertation is organized as follows:

Chapter 2 surveys the theoretical approaches to causality relevant to this dissertation. This includes Neyman-Rubin-Holland potential outcomes, structural

equation modeling, structural causal models, causal diagrams and graphical models, and means of inference. The relationship between these different approaches and their relationships to the Causal and Heckman hierarchies is analyzed. In addition surveying different methods in causal modeling and inference, this chapter argues that structural causal models serve as an appropriate foundation for a general theory of causality.

Chapter 3 introduces a causation coefficient as analogue to the correlation coefficient. A taxonomy of correlation/causation relationships is developed and analyzed. This provides a principled argument for the necessity of formal causal analysis — informal causal analysis will fail unpredictably.

Chapter 4 introduces the (meta-)theory of causal programming and its core abstractions: model distribution, query, and formula, defined in terms of structural causal models. A large number of existing problems can be encompassed in this framework, and several existing algorithms can be viewed as solving special cases of causal programming problems.

Chapter 5 introduces Whittemore, an implementation of causal programming as an embedded, domain specific language. The syntax, semantics and relevant implementation details are described for Whittemore's approach to solving identification and estimation problems. Whittemore provides a declarative, interactive approach to causal modeling and inference. It is declarative in the sense that the user does not have to be aware of the implementation of the underlying inference algorithms, and interactive in the sense that it can be used in a computational 'notebook' interface, providing immediate feedback to the user.

Chapter 6 explores a fundamental limit to the theory: the equilibrium distribution of cyclic, nonparametric models is proven to be uncomputable, given mild assumptions about the distributions of the model's variables. This demonstrates that causal programming's focus on recursive, nonparametric models is, in some sense, a natural limitation and suggests that further generalizations will encounter fundamental difficulties.

Finally, Chapter 7 concludes with a discussion of the contributions and limitations of the work described in this dissertation, as well as opportunities for future research.

# Chapter 2: Background: structural causal models

> A map is not the territory it represents, but, if correct, it has a similar structure to the territory, which accounts for its usefulness.
>
> —Alfred Korzybski

## 2.1 Introduction

The original work in this dissertation builds on the theory of structural causal models. This chapter provides an overview of previously existing theory and analyzes the relationship between the Neyman-Rubin-Holland potential outcome, structural equation modeling, and structural causal model approaches to causal inference. The relationship between causal diagrams and Bayesian networks is also discussed. Since the primary focus of this dissertation is on recursive (acyclic) models, substantive discussion of cyclic models is deferred to Chapter 6.

The core concept of a structural causal model is deceptively simple: an SCM is a manipulable/modifiable system of equations that generates a joint probability distribution over the variables under consideration. *Setting* variables to particular values (as opposed to observing or conditioning on particular values), generates new outcomes, i.e. new joint probability distributions that model the effect of an

idealized external intervention.

This can be seen in a simple example.[1] Consider the problem of a researcher attempting to model the effect of applying fumigants to fields. The researcher observes the level of fumigants applied by other farmers and their resulting crop yields. However, the level of fumigants applied by each farmer depends on the initial population of pests (e.g. a farmer is more inclined to apply fumigants if they expect the population of pests to be higher that year), and both the initial population of pests and the level of fumigants applied affect the final yield (i.e. the observations were not from a randomized, controlled trial). These assumptions can be captured in the following system of equations:

$$\texttt{pests} = f_1(\epsilon_1)$$

$$\texttt{fumigants} = f_2(\texttt{pests}, \epsilon_2)$$

$$\texttt{yield} = f_3(\texttt{pests}, \texttt{fumigants}, \epsilon_3)$$

where each $\epsilon_i$ is an 'error term', accounting for factors outside of the model and each $f_i$ is some function that determines each variable. One possible instantiation of these assumptions is as a linear Gaussian model:

$$\texttt{pests} = \epsilon_1$$

$$\texttt{fumigants} = \beta_1 \ \texttt{pests} + \epsilon_2$$

[1] This example is a simplified version of an agricultural example analyzed by Wainer [86] Pearl [57].

$$\texttt{yield} = \beta_2\ \texttt{pests} + \beta_3\ \texttt{fumigants} + \epsilon_3$$

where each $\epsilon_i$ is an independent, normally distributed ($N(\mu_i, \sigma_i)$) random variable, and each $\beta_i$ is a constant. This system of equations generates a joint probability distribution over `pests`, `fumigants` and `yield`, which is what the researcher originally observes (the 'observational distribution').

With respect to this model, an idealized intervention can be represented by *replacing* one or more of the generating equations. To model the action of applying a particular level, $x$, of fumigants, the equation determining `fumigants` is replaced by a constant, creating a new system of equations:

$$\texttt{pests} = \epsilon_1$$

$$\texttt{fumigants} = x$$

$$\texttt{yield} = \beta_2\ \texttt{pests} + \beta_3\ \texttt{fumigants} + \epsilon_3$$

which generates a new joint probability distribution representing the effect of action (the 'interventional distribution').

Computing the interventional distribution acts as a prediction. In general — assuming the model assumptions are correct — a researcher can predict a change to a system, by calculating the probability distribution that arises from a corresponding change to the model. The difficulty lies in making appropriate model assumptions. As a rule, stronger model assumptions make inference easier, but weaker model assumptions are more likely to correspond well to reality and be accepted by other

researchers.

Linearity is a relatively strong assumption. An example of a weaker assumption would be monotonicity, e.g. that an increase in the pest population always results in a decrease in the crop yield. Weaker still is the assumption that `yield` is a function of `fumigants`, `pests` and an independent error term, without committing to any assumption of what that function is or how the error term is distributed. The error term, or 'background' variable represents the factors that determine the final `yield`, that are not explicitly accounted for in the model.[2]

It is this last class of model assumptions that is the main focus of this dissertation. The problem is to calculate the interventional distribution for *every* model that is compatible with a given set of model assumptions. These model assumptions can be compactly represented in graphical form: a vertex of a graph corresponds to a variable, and a variable is assumed to be a function of its parents.

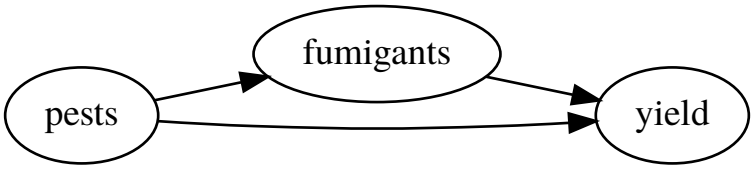



Figure 2.1: A causal diagram, representing the assumptions that `yield` is a function of `pests` and `fumigants`, and `fumigants` is a function of `pests`. Each variable is also (implicitly) assumed to be a function of an independent, arbitrarily distributed 'background' variable, representing factors outside of the model.

Structural causal models are certainly not the only approach to causal modeling and inference. However — in addition to acting as a survey of existing approaches — this chapter argues that structural causal models are an appropriate foundation

[2]In reality, there may be many such factors, but since there are no restrictions on the distribution of the error term, nor the determining function, all of these 'background' factors can be represented by a single variable.

for a general theory of causation. This argument is made with respect to a particular set of requirements for such a theory. These requirements do not apply to all scenarios of interest, but are argued to be necessary in the general case — a less powerful theory will be unable to express certain concepts of scientific interest.

## 2.2 Desiderata for causal modeling and inference

The first task in the Heckman Hierarchy is to define the set of counterfactuals or potential outcomes. This is directly related to a core concept in causality: variables are functions of their determining factors and manipulating/changing these factors generates new outcomes.

This is readily seen in a simple example. Consider Newton's second law, relating force, mass and acceleration, which elementary algebra permits being written in three different ways:

$$F = ma$$

$$m = \frac{F}{a}$$

$$a = \frac{F}{m}$$

Common intuition suggests that if the force applied to some object were increased, it would experience greater acceleration; the mass of the object would not spontaneously increase to compensate. In other words, force causes acceleration, but does not causally effect changes in mass. This is not clear from the standard presen-

tation of the equations, treating the equals sign as the equality relation, whereupon all three equations are equivalent. Under this interpretation of the equals sign, the equations specify a relationship that must be satisfied, but do not specify how the system would respond to an external action.

The three equations are different if the equals sign is treated as an assignment operator — this interpretation may be more familiar to programmers.[3] In this case, only the third equation captures the intuition that if force were increased, then acceleration would increase proportionally, as long as no other changes to the system were made. Likewise, if mass were increased, acceleration would decrease proportionally. Assumptions of this kind are commonly referred to as *ceteris paribus* assumptions, literally, "other things being equal". In particular, *ceteris paribus* is a mainstay of economic analysis [33].

This view of causality has its roots in neoclassical economics, especially in the work of Mill [52] and Marshall [51] and was made more precise with Haavelmo's account of (linear) structural equation models [29]. Rubin and Holland [38] provide a pithy motto that summarizes the main idea:

No causation without manipulation

This account of causality may not seem entirely satisfactory, depending on how 'manipulation' is interpreted. For example, few would object to thinking of the Sun's gravity as a cause of Earth's orbit, although, in practice, there are a number of obstacles to significantly manipulating the Sun's mass. This does not change the

[3] One might observe that the meaning would be clearer if a different symbol such as `:=` or `<-` was used to distinguish assignment from equality, but this convention has not been widely adopted.

expectation that if the Sun's mass were suddenly zero, then the Earth would not continue to orbit, no matter how implausible actually implementing such a change would be.

Implicit in such examples of causation is the idea that some effect ($Y$), *could have been different* if a cause ($X$) had been different, regardless of what was actually observed. Such hypotheticals are usually referred to as 'counterfactuals'. The idea of defining causality in terms of counterfactuals originates with Hume, defining a cause to be, "... where, if the first object had not been, the second never had existed" [40]. This idea was made more precise with Lewis' account of counterfactuals, using the possible world semantics of modal logic [48]:

> If $c$ and $e$ are two distinct actual events such that $e$ would not have occurred without $c$, then $c$ is a cause of $e$.

A classic example of a counterfactual sentence is, "If Nixon had pressed the button, there would have been a nuclear holocaust" [21], which features a number of important characteristics of counterfactuals in general. It cannot (nor should) be empirically tested, but it is still related to the observable world — note that Nixon did not ever order a nuclear strike, nor is the world a nuclear wasteland. The sentence also, indirectly, implies empirical consequences. If the original counterfactual sentence is true, anyone acting in a sufficiently similar scenario who does 'press the button' should expect a nuclear holocaust.

The ideas of manipulations and counterfactuals are related. One view is that a manipulation changes the original system or, more abstractly, generates a new

model that represents the effects of the change. Alternatively, the complete set of counterfactuals can be thought of as existing *a priori* and related to observable variables by consistency constraints. Whether or not counterfactuals 'actually' exist need not be a concern. Heckman summarizes the relevant metaphysical concerns as, "A model is in the mind. As a consequence, causality is in the mind" [33].

Crucially, the notion of counterfactuals is distinct from that of uncertainty — note that there is no notation in probability theory for "would have been". At the same time, statements of causality often include a probabilistic aspect. For example, most would interpret, "If the grass is wet, then it rained", as a statement that it is likely that rain caused grass to be wet, without committing to a fully deterministic model such as Newton's second law does.

To summarize: A satisfactory approach to causal modeling and inference requires the distinct concepts of manipulation, counterfactuals and uncertainty; treating any one of these as the same concept is a category error.

## 2.3 Potential outcomes

The potential outcome approach to causal inference, also known as the Rubin causal model, or the Neyman-Rubin-Holland model, provides a notation suitable for representing counterfactual statements. The statement that "$Y$ would have taken on value $y$, if $X$ had been $x$, for unit $u$" is written as:

$$Y_x(u) = y$$

Units are primitives in the potential outcomes approach, which does not define them further [38]. Examples of units include individual patients in a clinical setting, or individual plots of land in an agricultural study. Variables (e.g. $X$, $Y$) are real-valued functions defined for every unit; for example, $X$ is commonly defined to be treatment and $Y$ defined to be response to treatment in a clinical setting.

A particular quantity of interest is treatment effect[4], which is defined as the difference in response when a particular unit is exposed to treatment ($X = t$) versus control ($X = c$):

$$Y_t(u) - Y_c(u)$$

Causal inference is difficult because, although there are many potential outcomes for any particular variable, it is only possible to observe one *actual* outcome. For example, it is impossible to treat and not treat the same patient.[5] Holland summarizes this as the Fundamental Problem of Causal Inference [38]:

> It is impossible to *observe* the value of $Y_t(u)$ and $Y_c(u)$ on the same unit and, therefore, it is impossible to *observe* the effect of $t$ on $u$.

Causal inference is impossible without making additional assumptions — data alone provides no knowledge of how observations will generalize to other circumstances. An example of a simple assumption that makes causal inference possible

[4]This quantity is sometimes referred to as 'causal effect'; this dissertation adopts the convention of referring to this as 'treatment effect' to avoid confusion with the structural causal model definition of causal effect.

[5]One might object that it is possible to not treat a patient initially, and then treat the same patient later, but these are different units. The patient's condition after waiting long enough to observe the effects of non-treatment is different than their initial condition.

is unit homogeneity, which can be thought of as 'laboratory conditions'. If different units are carefully prepared, it may be reasonable to assume that they are equivalent in all relevant aspects, i.e. $Y_t(u_1) = Y_t(u_2)$ and $Y_c(u_1) = Y_c(u_2)$. For example, it is often assumed that any two samples of a given chemical element are effectively identical. In these cases, treatment effect can be calculated directly as $Y_t(u_1) - Y_c(u_2)$. However, it is often the case that such tightly controlled conditions are impossible to maintain. Accordingly, the main focus of the potential outcomes approach is on average effects.

A probability distribution over the universe of units, $P(u)$, induces a probability distribution over the potential outcome variables. Formally [57]:

$$P(Y_x = y) = \sum_{\{u \mid Y_x(u)=y\}} P(u)$$

Since the potential outcome variables are random variables, it is meaningful to speak of average treatment effects. In particular, expected value ($E$) is a linear operator which permits writing:

$$E(Y_t - Y_c) = E(Y_t) - E(Y_c)$$

In other words, it is possible to estimate average treatment effect by estimating $E(Y_t)$ and $E(Y_c)$ individually. Unfortunately, it is not possible to sample from either of these random variables directly. $Y_t$ is treatment over the *entire* universe of units, a counterfactual world where every patient was exposed to treatment. Actual samples would be from the random variable $Y$. Although these variables are different, they

are still related to each other by the consistency axiom [23, 27]:

$$X(u) = x \implies Y(u) = Y_x(u)$$

In other words, if the variable $X$ is observed to take on value $x$, then the potential response $Y_x$ is simply the current value of $Y$. An immediate consequence of consistency is: $P(Y_x = y \mid X = x) = P(Y = y \mid X = x)$. Furthermore, if response to treatment is *independent* of treatment (written as $Y_x \perp\!\!\!\perp X$), then the following equalities hold:

$$E(Y \mid X = t) = E(Y_t)$$

$$E(Y \mid X = c) = E(Y_c)$$

In this case, average treatment effect can be estimated directly from the collected samples as $E(Y \mid X = t) - E(Y \mid X = c)$. This expression is sometimes referred to as the *prima facie* treatment effect [38].

However, there are many scenarios where selection of treatment is not independent of response to treatment. Consider the question of whether smoking is a cause of cancer. The *prima facie* effect may be significant. However, it is conceivable that there exists a latent genetic factor that predisposes individuals to smoke, and also makes them more susceptible to cancer.[6] This is an example of the well-known problem of confounding variables and the possibility of latent confounding variables

[6]Statistician Ronald Fisher is infamous for having spoke out against studies linking smoking to cancer, while being ardent tobacco user himself. He later died of cancer. However, his actual objections to the studies were not incorrect.

is especially difficult to rule out.

The potential outcomes approach generally operates by making (conditional) independence assumptions about potential outcome variables. Randomization, i.e. the samples were obtained in a randomized controlled trial, makes the assumption[7] $Y_x \perp\!\!\!\perp X$ especially plausible since — at least, theoretically — the selection of treatment or non-treatment for each unit is determined entirely by an independent source of randomness. In practice, there may be issues with imperfect compliance (i.e. some patients may fail to take the drugs they are assigned), but randomized controlled trials remain the 'gold standard' for causal evidence. [71]

Another common type of assumption that can permit causal inference is conditional ignorability, $(Y_x \perp\!\!\!\perp X \mid Z)$, which is the statement that $Y_x$ and $X$ are conditionally independent given $Z$, a set of covariates that are being 'adjusted' or 'controlled' for. For example, if it were known that there was a genetic factor $Z$ (and no *other* such factors) that caused both smoking and cancer, then it would be reasonable to assume conditional ignorability, which would permit the following derivation [57]:

[7] This condition is referred as 'no confounding', 'exogeneity' or 'ignorability', depending on the source [57, 69, 20].

$$
\begin{aligned}
P(Y_x = y) &= \sum_z P(Y_x = y \mid z) P(z) \\
&= \sum_z P(Y_x = y \mid x, z) P(z) \\
&= \sum_z P(Y = y \mid x, z) P(z) \\
&= \sum_z P(y \mid x, z) P(z)
\end{aligned}
$$

The notation belies a fundamental shift in perspective: the formula computes the probability of a *potential outcome*, $P(Y_x)$, entirely in terms of *observable* probabilities, $P(y \mid x, z)$ and $P(z)$, with respect to the assumption of conditional ignorability.

This is, in essence, the potential outcomes approach to the second causal inference task of identification. The first task, defining the set of counterfactuals, is implicitly performed by making conditional independence assumptions, e.g. $(Y_x \perp\!\!\!\perp X \mid Z)$. The second task, identification, is performed via algebraic manipulations, using the axioms of probability theory and the axioms associated with potential outcome variables. The third task is performed by calculating the appropriate probabilities, brining in the machinery of estimation and/or hypothesis testing theory as appropriate.

However, despite the power of the potential outcomes approach, an important practical issue has not been addressed: how is a researcher to determine if the conditional independence assumptions are true? The fundamental problem is deeper

than this: potential outcomes notation, alone, does not even provide a way to determine what it means for such a statement to be true! In the language of formal logic, potential outcomes notation provides *syntax* but not *semantics* for causal statements. Giving these statements meaning requires formalizing the notion of a causal model.

## 2.4 Causal models

Consider a simple economic model of propensity to consume, assuming all prices are constant. As an example, Haavelmo suggests a model where, "if the group of all consumers in society were repeatedly furnished with the total income or purchasing power $x$ per year, they would, on the average or 'normally,' spend a total amount $y$ equal to" [29]:

$$y = \beta x + \alpha$$

where $\alpha$ and $\beta$ are constants. It would be unreasonable to expect that, in any particular year, spending would be exactly equal to $y$. This is not merely a consequence of measurement errors — presumably there are a large number of additional factors that could affect spending that are not directly accounted for in this simple model. These additional factors can be indirectly represented by adding a residual or 'error term' to the original equation:

$$y = \beta x + \alpha + \epsilon$$

$$= \beta x + \epsilon$$

where $\epsilon$ is a (usually, normally distributed) random variable; note that since $\alpha$ is a constant and $\epsilon$ is a random variable, $\alpha + \epsilon$ can be replaced by just $\epsilon$. This is a simple example of a structural equation model (SEM).

Haavelmo is notable for being the first researcher to explicitly interpret such equations as predicting the result of idealized experiments. It may be the case that an analyst is merely trying to fit the equation to the past and hopes that the relation holds in the future, assuming no significant changes to the underlying system. A stronger assumption is that consumers will continue to respond in the same way to income, regardless of the sources from which their income originates. With respect to this assumption, it is possible to predict the result of an intervention (e.g. government spending or taxation) to set income at a given level. Pearl formalizes this interpretation of structural equations:

**Definition 2.4.1 (Structural Equations [57])** *An equation $y = \beta x + \epsilon$ is said to be* structural *if it is to be interpreted as follows: In an ideal experiment where we control $X$ to $x$ and any other set $Z$ of variables (not containing $X$ or $Y$) to $z$, the value $y$ of $Y$ is given by $\beta x + \epsilon$ where $\epsilon$ is not a function of the settings $x$ and $z$.*

Note that this definition assumes an idealized intervention to *set* $X$ to a par-

ticular value, as opposed to conditioning on $X$; it is the difference between passively observing a particular value (level 1 in the Causal Hierarchy) and taking action to set the value (level 2 in the Causal Hierarchy). In practice, many manipulations that are theoretically simple can turn out to be difficult or impossible implement. This is not a strike against the definition, but a warning to carefully model interventions as well as the causal relationships themselves.

The philosophical underpinnings of this definition are that of Laplacian (quasi-) determinism. The residual, $\epsilon$, represents all of the additional factors that determine $Y$ that are not directly modeled. In principle, if these factors were completely known, it would be possible to exactly determine how $Y$ would respond to any change. In this view, randomness is a statement of an analyst's ignorance, not inherent to the system itself.

Structural equations are related to the potential outcomes approach, which considers potential outcome variables to be real-valued functions of 'units'. Since units are primitives and not defined further, these functions are implicit. In structural equation modeling, these functions are explicit, where variables are functions of all of their determining factors.

One of the weaknesses of structural equation modeling is that it makes very strong assumptions — usually, linearity and the assumption that all variables are multivariate normal. It is perhaps unsurprising then that many analysts are reluctant to assign causal meaning to the equations and consider them to be merely a 'shorthand' way to represent a joint probability distribution. The linearity assumption, in particular, is very restrictive — the earlier example of Newton's second

law violates it. Consider, also, the smoking/cancer example, where $X$ is smoking, $Y$ is cancer, and $Z$ is a possible genetic factor that predisposes one to smoke and can cause cancer. These assumptions can be captured in the following system of equations:

**Example 2.4.1 (Smoking/cancer model)**

$$Z = f_Z(\epsilon_Z)$$

$$X = f_X(Z, \epsilon_X)$$

$$Y = f_Y(X, Y, \epsilon_Y)$$

Each $f_i$ is some — possibly nonlinear – function. $X, Y, Z$ are called 'endogenous variables' since they are determined by factors in the model. $\epsilon_X, \epsilon_Y, \epsilon_Z$ are called 'background variables' since they are determined by outside factors that are not directly accounted for.[8]

There are many — in fact, an uncountably infinite number of — models that are compatible with the smoking/cancer model assumptions. One instantiation of the assumptions is as an SEM:

**Example 2.4.2 (Linear smoking/cancer model)**

$$Z = \epsilon_Z$$

[8]These are sometimes referred to as 'exogenous' variables. Unfortunately, 'exogeneity' is often used to refer to a number of subtly different conditions between sets of variables in a causal model. To avoid confusion, the term 'background variable' will be used.

$$X = Z + \epsilon_X$$

$$Y = X + Y + \epsilon_Y$$

where each $\epsilon_i$ is an independently distributed normal random variable. Alternatively, $Y$ could be modeled as a logistic function of $X$ and $Z$:

**Example 2.4.3 (Logistic smoking/cancer model)**

$$Z = \epsilon_Z$$

$$X = Z + \epsilon_X$$

$$Y = \frac{1}{1 + e^{-(\beta_0 + \beta_1 Z + \beta_2 X)}}$$

The key point is that SCMs are not limited to any particular set of functional dependencies or distribution of background variables; SCMs are fully nonparametric. This nonlinear generalization of structural equation models with arbitrarily distributed background variables originates with Pearl and Verma [62] and has been referred to by several different terms including 'probabilistic causal models', 'graphical causal models' and 'structural causal models'. The term 'structural causal models' is used throughout this dissertation, since it appears least likely to name clash with other terms in the literature.

**Definition 2.4.2 (Structural Causal Model [3])** *A* structural causal model $M$ *is a tuple* $M = \langle U, V, F, P(u) \rangle$*, where:*

1. *A set $U$ of background (also called* exogenous*) variables, that are determined by factors outside the model*

2. *A set $V = \{V_1, \ldots V_n\}$ of variables, called* endogenous*, that are determined by variables in the model — that is, variables in $U \cup V$;*

3. *$F$ is a set of functions $\{f_1, \ldots, f_n\}$ such that each $f_i$ is a mapping from (the respective domains of) $U_i \cup PA_i$ to $V_i$, where $U_i \subseteq U$ and $PA_i \subseteq V \setminus V_i$ and the entire set $F$ forms a mapping from $U$ to $V$;*

4. *$P(u)$ is a probability function defined over the domain of $U$.*

Note that the definition of structural causal models requires that the set of equations, $F$, form a mapping from $U$ to $V$. In other words, that $F$ has a unique solution for $V$ as a function of $U$. A sufficient condition for this is that the system is recursive, i.e. there are no cyclic dependencies in the parent ($PA_i$) sets of the endogenous variables. A key difficulty with nonrecursive systems in structural causal models is that they may require solving systems of nonlinear equations.

## 2.5 Interventions and the do() operator

Structural causal models provide a straightforward definition of interventions. Consider an action to force some set of variables $X$ to take on particular values $x$; this is represented using the *do*() operator.

**Definition 2.5.1 (Effect of action [57])** *Let $M$ be a structural causal model, $X$ a set of variables in $V$, and $x$ a particular realization of $X$. The* effect of action

*$do(X = x)$ on $M$ is given by the submodel $M_x$.*

**Definition 2.5.2 (Submodel [57])** *Let $M$ be a structural causal model, $X$ a set of variables in $V$, and $x$ a particular realization of $X$. A submodel $M_x$ of $M$ is the causal model:*

$$M_x = \langle U, V, F_x, P(u) \rangle$$

*where:*

$$F_x = \{f_i : v_i \notin X\} \cup \{X = x\}$$

A submodel produced by $do(X = x)$ can be thought of as the result of 'wiping out' each $f_i$ that determines each $X_i$, and replacing $f_i$ with the constant $x_i$, a process which Pearl colorfully refers to as performing "surgery on equations" [57]. As an example, consider an idealized intervention to determine the causal effect of smoking on cancer. In the original model, $M$, the decision to smoke ($X$) is a function of a background variable ($\epsilon_X$) and a genetic factor ($Z$) that both predisposes one to smoke and affects cancer risk. The intervention, $do(X = x)$, effectively 'cuts out' the confounding from the genetic factor and produces a new model, $M_x$, in which the factors that determine $Z$ and $Y$ are unchanged, but $X$ has been set to the value

$x$:

**Example 2.5.1 (Smoking/cancer submodel)**

$$Z = f_Z(\epsilon_Z)$$

$$X = x$$

$$Y = f_Y(X, Y, \epsilon_Y)$$

Given the definitions of a submodel and effect of action, the relationship between potential outcomes and structural causal models is remarkably straightforward:

**Definition 2.5.3 (Potential Response [57])** *Let $X$ and $Y$ be two subsets of variables in $V$. The* potential response *of $y$ to action $do(X = x)$, denoted $Y_x(u)$, is the solution for $Y$ of the set of equations $F_x$, that is, $Y_x(u) = Y_{M_x}(u)$.*

The probability of $y$, given the action $do(X = x)$ is denoted[9] by either $P(Y_x)$ or $P(y \mid do(x))$, and is induced by the probability distribution over the background variables, $P(u)$, and the submodel, $M_x$:

$$P(Y_x = y) = P(y \mid do(x)) = \sum_{\{u \mid Y_{M_x}(u) = y\}} P(u)$$

Note that $P(y \mid do(x))$ is also referred to as the *causal effect* of $X$ on $Y$, when viewed as a function from $X$ to the space of probability distributions on $Y$ [57].

[9]Other notations, such as $P_x(y)$ or $P(y \mid \hat{x})$ are in use.

This establishes the theoretical connection between potential outcomes and structural causal models. It also highlights the philosophical differences between traditional structural equation modeling and potential outcome analysis. In structural equation modeling, equations are usually assumed to be linear, with the random variables being multivariate normal. In other words, structural equation modeling relies on strong and explicit model assumptions. Potential outcome analysis is effectively the opposite: the model assumptions are weak and implicit. Independence assumptions between potential outcome variables implicitly constrain the set of possible models under consideration, but do not provide much guidance on determining what that set is.

Do notation also establishes the demarcation line between each level of the Causal Hierarchy. Syntactically, an associational/statistical query is any query that does not contain $do()$ or any potential outcome variables; this is exactly what is expressible with the syntax of ordinary probability theory. Semantically, such a query is concerned with a pre-intervention model. Syntactically, interventional/causal queries may contain $do()$. Semantically, such queries are questions about *post*-intervention models.

Finally, counterfactual queries are the most general and include the full range of what is expressible in potential outcomes notation. For example, the *effect of treatment on the treated*, e.g. the outcome ($Y$) of treating a patient with $X = x$, given that $X$ attains value $x'$ naturally, is expressible as $P(Y_x = y \mid x')$, which is inexpressible in the $do()$ notation. It is a question of probabilities in a post-intervention model, but involves conditioning on variables in a pre-intervention model.

Structural causal models permit analyzing general counterfactual queries: the key insight that permits doing so is that although counterfactual questions consider multiple counterfactual worlds simultaneously, the corresponding models share the same background variables ($\epsilon_i$) and functional dependencies ($f_i$), thus, conditioning on an event in the 'actual' world provides information about counterfactuals and vice-versa. Since the primary focus of this dissertation is on interventional queries, this section will not consider the analysis of counterfactual queries further.

## 2.6 Marschak's maxim and causal diagrams

Heckman coined 'Marschak's Maxim', in honor of an insight by Marschak [50]:

Forecasting policies may require only partial knowledge of the system.

From the definition, a complete specification of a structural causal model requires specifying the functions that determine each endogenous variable and the probability distribution over the background variables. The former is often difficult to know; the later is often impossible, considering that the background variables are usually the very factors that cannot be directly accounted for.

Marschak's maxim is a reminder that a partial specification of a model may still be sufficient to conduct causal inference. A set of independence assumptions made in a potential outcomes analysis implicitly denotes a set of possible models. Causal diagrams are another approach.

Every causal model induces a causal diagram, where each vertex in the diagram corresponds to an endogenous variable, $V_i$, and directed edges point from

members of $PA_i$ to $V_i$. If the background variables are jointly independent and each background variable appears in only one $PA_i$ set, the model is called Markovian [57]. Otherwise, the model is called semi-Markovian. Dependencies between endogenous variables due to background variables are denoted by dashed, bidirectional edges.[10] For example, if the genetic factor in the smoking/cancer example were known and measurable, the model would be Markovian; otherwise, the model would be semi-Markovian and $X$ and $Y$ would have a dashed, bidirectional edge between them to denote the dependency (figure 2.2).

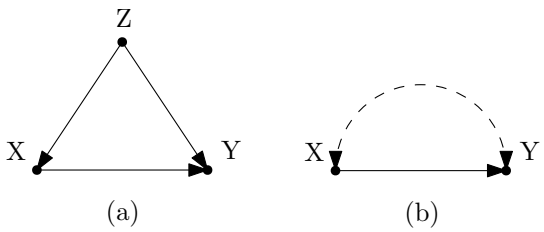



Figure 2.2: Markovian (a) and semi-Markovian (b) causal diagrams. In (a), $X$, $Y$, and $Z$ are each functions of their parents and an independently distributed background variable. In (b), $X$ and $Y$ share a background variable; equivalently, $\epsilon_X$ and $\epsilon_Y$ are not assumed to be independent.

There is a useful correspondence between causal diagrams and causal models. Every causal model induces a causal diagram, and every causal diagram has at least one model (in fact, infinitely many) that would induce it. A casual diagram can be thought of as denoting a set of models where each endogenous variable is assumed to be a function its parents, without committing to an assumption of what the function is.

Causal diagrams are also closely related to probabilistic graphical models,

[10] An alternative convention is to enter observable variables as solid nodes and latent variables as hollow nodes.

specifically, Bayesian networks. The probably distribution $P(v)$ induced by a Markovian causal model respects the Markov condition.

**Definition 2.6.1 (Markov condition [59])** *If a probability function $P$ admits the factorization:*

$$P(x_1, \ldots, x_n) = \prod_i P(x_i \mid pa_i)$$

*relative to DAG $G$, we say that $G$ and $P$ are Markov compatible.*

Equivalently [4], variables are conditionally independent of their nondescendants, given their parents. This is precisely the definition of (non-causal) Bayesian networks. The crucial difference lies in the underlying assumptions. 'Ordinary' Bayesian networks are (level 1) associational/statistical models; they provide an efficient way to factor a joint probability distributions, but make no assumptions as to how the distribution will change in response to an intervention. This can be seen in a near-trivial example: consider the networks $X \to Y$ and $Y \to X$; these have identical factorizations under the Markov property, but different behavior under intervention. In general, causal assumptions are necessary for causal conclusions. This is concisely summarized in Cartwright's maxim [10]:

No causes in, no causes out.

Inference in the structural causal model approach is generally performed with respect to the assumptions entailed by a causal diagram. For example, calculating the causal effect of $X$ on $Y$ in the smoking/cancer model is a simple adjustment for direct causes.

**Theorem 2.6.2 (Adjustment for Direct Causes [57])** *Let $PA_i$ denote the set of direct causes of variable $X_i$ and let $Y$ be any set of variables disjoint of $\{X_i \cup PA_i\}$. The effect of the intervention $do(X_i = x_i)$ on $Y$ is given by:*

$$P(Y \mid do(x_i)) = \sum_{pa_i} P(y \mid x_i, pa_i) P(pa_i)$$

Applying this theorem to the smoking/cancer example (Figure 2.2 (a)) yields:

$$P(y \mid do(x)) = \sum_{z} P(y \mid x, z) P(z)$$

Formally, it is said that the causal diagram and the probabilities $P(y \mid x, z)$ and $P(z)$ *identify* $P(y \mid do(x))$. Note that if $Z$ is latent, then $P(y \mid do(x))$ is not identifiable. Intuitively, there is no way of knowing if correlation between $X$ and $Y$ is due to the latent factor, or due to the effect of $X$ on $Y$.

Unsurprisingly, this is the same result as from the potential outcomes analysis. In the potential outcomes approach, the set of causal models under consideration is implicitly specified by the conditional independence assumptions between potential outcome variables. Causal diagrams more explicitly denote the set of models under consideration, and consider properties like conditional independences to be a consequence of the model assumptions entailed in the diagram. In both cases, Marschak's maxim is in play. A complete specification of the model is not needed to calculate the causal effect; the formula correctly calculates $P(y \mid do(x))$ for all models under consideration.

Note that incorrectly adjusting for variables can produce biased estimates

of causal effect. Consider a model where the treatment affects recovery, but the treatment also affects blood pressure, which, in turn, affects recovery (Figure 2.3). An adjustment for direct causes, i.e.

$$\sum_{\texttt{blood-pressure}} P(\texttt{recovery} \mid \texttt{treatment}, \texttt{blood-pressure}) P(\texttt{blood-pressure})$$

should not be performed in this case, since blood-pressure is not a direct cause of treatment. Intuitively, adjusting for blood pressure would 'block' the causal effect of the treatment that is mediated through blood pressure.

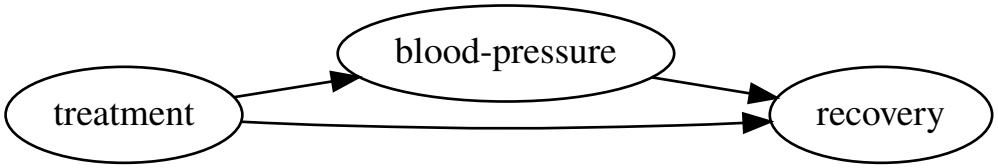



Figure 2.3: A causal diagram where effect of treatment on recovery is mediated by blood pressure. Estimating the treatment effect by adjusting for blood-pressure is incorrect in this case.

## 2.7 Philosophical foundations

The philosophical underpinnings of structural causal models depend on a few well-accepted principles: counterfactuals as the basis for a theory of causality, models as the basis for a semantic theory of truth, and probability theory as the basis for a theory of uncertainty. It is a futile exercise to argue that these principles are somehow *a priori* correct; the practical question is to what degree these principles are useful, generally accepted, and compatible with human intuition.

Probability theory's pedigree is impeccable: in addition to Kolmogorv's ax-

iomatization, Cox's theorem [84] provides a formal argument for probability theory as the basis of analyzing uncertainty, and de Finetti's Dutch Book argument requires that wagers/prices obey the axioms of probability theory in order to be coherent [17]. Empirically, probability theory has been an enormous success: it is effectively the basis of all of statistics and machine learning. It may be the case that, for specific scenarios, a researcher wishes to consider problems of causality without introducing the notion of uncertainty. However, any general theory of causality needs to formally consider uncertainty to be taken seriously.

The semantic theory of truth, i.e. the idea that models provide meaning for sentences, is similarly well accepted in modern mathematics, philosophy and computer science. The truth of a sentence in a formal language is judged relative to a formal model that provides an interpretation of said sentence. This is the core principle behind model theory in mathematics and formal methods in computer science, and it would seem profoundly strange to not respect it in the domain of causal analysis. It has been suggested that this is the main reason that the potential outcomes approach has not been widely adopted outside of statistics [57] — it is too difficult for most would-be users to judge if model assumptions are reasonable without referring to actual examples of models. A rough analogy can be made with Hoare logic, which provides axioms and inference rules to prove the correctness of computer programs. Attempting causal inference without explicit models is like trying to prove the correctness of programs while being unable to explicitly *write down a program.*[11]

[11] Anecdotally, Elias Bareinboim reports being unable to find a single statistician capable of spec-

The principle that a theory of causality falls out of a theory of counterfactuals as a by-product lacks the essentially universal acceptance of probability theory and model theory. In fact, some statisticians explicitly argue against counterfactuals [16], citing them as untestable or metaphysical. However, counterfactual thinking matches the intuitive way that human beings think of causality: for an event to be a cause of another, it must have been possible, in principle, for the cause to have been different. Mere untestability should not be grounds to disqualify questions from analysis; science routinely considers questions that have no obvious means of empirical validation, and tries to fit explanations to systems that cannot be directly controlled. Notably, the idea that different possible outcomes are generated by manipulating the factors that determine them is essentially universally accepted in economic analysis. Arguably, economics is the science most concerned with predicting the effects of interventions, despite a general lack of ability to conduct controlled experiments.

Given these principles, structural causal models are not merely one possible approach to causal modeling and inference; they are a foundation for a general theory of causality. As an analogy, consider the lambda calculus as a foundational theory of computation. The lambda calculus captures the essential notion of computation. There are numerous extensions, e.g. type theory, and it remains unclear what it means to *prove* that the Church-thesis is correct, but it provides (one possible) foundation for a general theory of computation. In particular, the lambda calculus subsumed the existing notions of computability as special cases. Analogously,

ifying an example model where $(Y_z \perp\!\!\!\perp Z_x \mid Z, X)$ and $(Y_z \not\perp\!\!\!\perp Z_x \mid Z)$ (personal communication).

structural causal models unify and extend the potential outcomes and structural equation modeling approaches.

These philosophical principles leave the actual problem of inference with structural causal models unaddressed. When a structural causal model is fully specified, there is very little inference left to be done: calculating the effect of an intervention can be done in a straightforward manner from definition of the $do()$ operator. In practice, fully specifying a structural causal model is generally impractical for problems of interest; modeling a real system involves considering possible equivalence classes of models, since the full details remain unknown.

The core problem is to determine what kinds of equivalence classes of models should be considered. The previous section introduces causal diagrams, which are the main object of study in the rest of this dissertation. Compared to structural causal models themselves, there are not strong philosophical principles to justify causal diagrams as the canonical model representation.

One argument is that the assumptions entailed by a causal diagram are completely nonparametric — there are no model assumptions, other than that each variable is a function of its parents. In this sense, the assumptions embodied in a causal diagram are minimal, while remaining compatible with the requirement that models be explicit — it is straightforward to fully specify an example compatible structural causal model, given a diagram. However, it is not uncommon to use additional semi-parametric assumptions in analysis. For example, the monotonicity of a function in a model may permit identification, while completely nonparametric assumptions do not [34]. Conversely, a researcher may wish to base their analysis

on weaker assumptions, e.g. to enter the fact that *either* $X$ causes $Y$ or $Y$ causes $X$, without knowing which.

Ultimately, this dissertation focuses on causal diagrams as a matter of practicality: the existence of a concise set of powerful inference rules for causal diagrams makes inference particularly amenable to automated inference.

## 2.8 Inference with structural causal models

The best-studied set of inference rules for structural causal models are known as the "causal calculus" or "do-calculus", described by Pearl [56]. Along with the axioms of probability theory, these rules form the theoretical foundation for much of the analysis in the rest of this dissertation:

**Theorem 2.8.1 (Causal calculus [56])**

$$P(y \mid \hat{x}, z, w) = P(y \mid \hat{x}, w) \text{ if } (Y \perp\!\!\!\perp Z \mid X, W)_{G_{\overline{X}}} \qquad \text{(Rule 1)}$$

$$P(y \mid \hat{x}, \hat{z}, w) = P(y \mid \hat{x}, z, w) \text{ if } (Y \perp\!\!\!\perp Z \mid X, W)_{G_{\overline{X}\underline{Z}}} \qquad \text{(Rule 2)}$$

$$P(y \mid \hat{x}, z, w) = P(y \mid \hat{x}, w) \text{ if } (Y \perp\!\!\!\perp Z \mid X, W)_{G_{\overline{X},\overline{Z(W)}}} \qquad \text{(Rule 3)}$$

Note that $\hat{x}$ is used as an abbreviation for $do(x)$ in this dissertation, whenever the density of equations threatens to render the do notation unreadable.

In the causal calculus inference rules, $W$, $X$, $Y$, and $Z$ are arbitrary disjoint

sets of nodes in a causal DAG $G$. $G_{\overline{X}}$ denotes the graph obtained by deleting from $G$ all arrows pointing into nodes in $X$. $G_{\underline{X}}$ denotes the graph resulting from deleting all arrows emanating from $G$. $Z(W)$ is the set of $Z$-nodes that are not ancestors of any $W$-node. $(Y \perp\!\!\!\perp Z|X,W)_G$ denotes the conditional independence of $Y$ and $Z$ given $X$ and $W$ in all models compatible with $G$.

Causal diagrams are particularly useful as equivalence classes of models because they permit determining conditional independences between (sets of) variables via a simple criteria known as d-separation [26]:

**Theorem 2.8.2 (d-seperation)** *A path $p$ is said to be d-separated by a set of nodes $Z$ if and only if*

- *$p$ contains a chain $i \to m \to j$ or a fork $i \leftarrow m \to j$ such that the middle node $m$ is in $Z$, or*
- *$p$ contains an inverted fork (collider) $i \to m \leftarrow j$ such that the middle node $m$ is not in $Z$ and such that no descendants of $m$ is in $Z$.*

  *A set $Z$ is said to d-separate $X$ from $Y$ if and only if $Z$ blocks every path from a node in $X$ to a node in $Y$.*

**Theorem 2.8.3 (Probabilistic implications of d-separation)** *If sets $X$ and $Y$ are d-separated by $Z$ in a DAG $G$, then $X$ is independent of $Y$ conditioned on $Z$ in every distribution compatible with $G$. Conversely, if $X$ and $Y$ are not d-separated by $Z$ in a DAG $g$, then $X$ and $Y$ are dependent conditional on $Z$ in at least one distribution compatible with $G$.*

The converse is even stronger — the absence of d-separation implies dependence in *almost all* distributions compatible with $G$ [57].

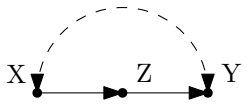



Figure 2.4: A semi-Markovian causal diagram that permits identification of $P(y \mid do(x))$; the causal effect can be determined in terms of the observational probability distribution by repeatedly applying rules of probability theory and the causal calculus.

Causal inference can be performed by repeatedly applying these rules, in conjunction with the rules of probability theory. For example [57], $P(y \mid \hat{x})$ can be identified for all models entailed by the assumptions in figure 2.4, in terms of the observational probability distribution as follows:

$$
\begin{aligned}
P(y \mid \hat{x}) &= \sum_z P(y \mid z, \hat{x}) P(z \mid \hat{x}) && \text{(law of total probability)} \\
&= \sum_z P(y \mid \hat{z}, \hat{x}) P(z \mid \hat{x}) && \text{(rule 2)} \\
&= \sum_z P(y \mid \hat{z}) P(z \mid \hat{x}) && \text{(rule 3)} \\
&= \sum_z \left[ \sum_x P(y \mid x, \hat{z}) P(x \mid \hat{z}) \right] P(z \mid \hat{x}) && \text{(law of total probability)} \\
&= \sum_z \left[ \sum_x P(y \mid x, \hat{z}) P(x) \right] P(z \mid \hat{x}) && \text{(rule 3)} \\
&= \sum_z \left[ \sum_x P(y \mid x, z) P(x) \right] P(z \mid \hat{x}) && \text{(rule 2)} \\
&= \sum_z \left[ \sum_x P(y \mid x, z) P(x) \right] P(z \mid x) && \text{(rule 2)}
\end{aligned}
$$

The difficulty of applying these inference rules has prompted the development of algorithms based on the causal calculus (discussed in Chapter 4). For certain classes of problems, these algorithms automatically perform causal inference, and have completeness guarantees — if they fail to yield an answer, it is because it is impossible to calculate the given causal effect for every model entailed by the given causal diagram. These algorithms can be thought of as an abstraction over the rules of the causal calculus and probability theory; the algorithms are proven correct with respect to these rules, but a researcher using the algorithms does not have to have detailed knowledge of the rules to perform inference.

One of the main goals of this dissertation is to develop higher-order abstractions, making it possible to perform rigorous causal inference without having to be aware of the underlying algorithms. In other words, the goal is to make causal inference fully *declarative*; a researcher should be able to enumerate what they already know and what they wish to determine, with the actual inference performed automatically.

It may not be immediately clear that the formalism of structural causal models is even necessary in practice. Although it is generally understood that correlation does not imply causation, it appears to act as useful guide in practice. A high degree of correlation 'suggests' causality, with experimental evidence providing confirmation. Why this may often appear to be the case, and an argument as to why it is insufficient, is the topic of the next chapter.

# Chapter 3: A taxonomy of correlation/causation relationships

> Correlation≠causation: the first thing taught in causal inference classes, and the last thing learned.
>
> —Gwern Branwen

## 3.1 Introduction

The maxim "Correlation is not causation" has reached the status of statistical clich. The difference is readily apparent if the Causal Hierarchy is taken seriously: correlation is an associational/statistical measure, i.e. a first-level query of the hierarchy. Interventional/causal queries are second level. This emphasis on firewalling the statistical from the causal leads directly to a new mystery: why do the two remain so easily confused in practice?

Other maxims have been proposed, such as, "Correlation is not causation, but it sure is a hint." or "Correlation is necessary but not sufficient for causation." [82]. The former is true, but underspecified. The latter is true in one sense, and demonstrably false in another. To help clarify these questions, this chapter introduces a causation coefficient, $\gamma$, to compare correlation and causation directly, a taxonomy to classify all of the possible relationships, and $\rho\gamma$ plots to visualize sets of causal

models.

Both the proposed taxonomy and $\rho\gamma$ plots are based on the causation coefficient. The correlation/causation taxonomy outlines the different ways in which correlation may (fail to) coincide with causation, with the goal of making it easier for researchers to recognize these effects in practice. In addition, example models of three variables are included with each classification in the taxonomy, rendering the taxonomy a constructive proof that the existence, or lack thereof, of correlation provides no guarantees about causation.

The $\rho\gamma$ plots are accompanied with an analysis of how correlation and causation relate in the 'average' model; this recovers some of the intuition as to why correlation 'suggests' causation and an explanation why they are so easily confused in practice.

## 3.2 The causation coefficient

The Pearson product-moment correlation coefficient, $\rho$, is a standard measure of correlation between random variables. This is commonly described as a measure of how well the relationship between $X$ and $Y$ can be modeled by a linear relationship with $\rho = -1/+1$ being a perfect negative/positive linear relationship and 0 representing no linear relationship at all. The population correlation coefficient is defined as a normalized covariance [88]:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sqrt{Var[X]Var[Y]}} = \frac{E[XY] - E[X]E[Y]}{\sqrt{(E[X^2] - E[X]^2)(E[Y^2] - E[Y]^2)}}$$

For discrete random variables, this is a function of the joint probability mass function (for continuous random variables that admit a probability density function, the summations are replaced with integrals):

$$\rho_{X,Y} = \frac{\sum_x \sum_y xyP(x,y) - \sum_x xP(x) \sum_y yP(y)}{\sqrt{(\sum_x x^2P(x) - (\sum_x xP(x))^2)(\sum_y y^2P(y) - (\sum_y yP(y))^2)}}$$

The causation coefficient relies on the observation that the correlation coefficient can, by the law of total probability, be rewritten as a function of the conditional distribution $P(y \mid x)$, and marginal distribution, $P(x)$, instead of in terms of the joint density:

$$\rho_{X,Y} = \frac{\sum_x \sum_y xyP(y \mid x)P(x) - \sum_x xP(x) \sum_x \sum_y yP(y \mid x)P(x)}{\sqrt{Var[X](\sum_x \sum_y y^2P(y \mid x)P(x) - (\sum_x \sum_y yP(y \mid x)P(x))^2)}}$$

Syntactically, the causation coefficient, $\gamma_{X \to Y}$, is defined by replacing $P(y \mid x)$ with $P(y \mid \hat{x})$ and $P(x)$ with $\hat{P}(x)$; note that $P(y \mid \hat{x})$ is simply an alternative notation for the causal effect, $P(y \mid do(x))$. $\hat{P}(x)$ is the *distribution of interventions*, described below. As a convenience, the following terms are also de-

fined: $Var[\hat{X}] = \sum_x x^2 \hat{P}(x) - (\sum_x x \hat{P}(x))^2$ and $Var[Y_{\hat{X}}] = \sum_x \sum_y y^2 P(y|\hat{x}) \hat{P}(x) - (\sum_x \sum_y y P(y|\hat{x}) \hat{P}(x))^2$. The full definition of $\gamma_{X \to Y}$ is then:

$$\gamma_{X \to Y} = \frac{\sum_x \sum_y xy P(y \mid \hat{x}) \hat{P}(x) - \sum_x x \hat{P}(x) \sum_x \sum_y y P(y \mid \hat{x}) \hat{P}(x)}{\sqrt{Var[\hat{X}] Var[Y_{\hat{X}}]}}$$

Like the correlation coefficient, the causation coefficient assumes values in the range $[-1, 1]$; intuitively, the causation coefficient can be thought of as what the correlation coefficient would have been, had the data been drawn from a randomized controlled trial. The distribution of interventions is, literally, a probability distribution over the independent variable, $X$. In the discrete case, it can be thought of as a set of weights for averaging the possible causal effects. It may be distinct from the marginal observational distribution of $X$, $P(x)$, since the relative sizes of cohorts in an observational study may be different than the relative sizes of different treatment (and control) groups in an randomized controlled trial.

As an example, consider a scenario where patients decide for themselves whether or not to take some treatment ($X$), and observe whether or not they recover ($Y$). This can be modeled with Bernoulli (binary) random variables for $X$ and $Y$, with 0 representing no treatment / failure to recover and 1 representing treatment / recovery. The probability of patients deciding for themselves whether or not to take the drug, in this observational study, is the marginal probability $P(x)$. In clinical terms, $P(X = 0)$ and $P(X = 1)$ are the relative sizes of the cohorts.

However, even in an idealized observational study, $P(y \mid x)$ would not provide

definitive information on whether treatment actually improves patient outcomes. For example, a drug could cause unpleasant side effects in the patients that would have received the greatest benefit, leading those patients to choose not to take the drug. An idealized randomized controlled trial would permit an analyst to directly measure $P(y \mid \hat{x})$, as randomization explicitly cuts out confounding.

The relative sizes of the cohorts in an observational study may be different than the relative sizes of the treatment and control groups in a corresponding randomized controlled trial — this is the use of the distribution of interventions $\hat{P}$. Experiments are often designed to have equal group sizes as this typically provides maximum statistical power, but this is by no means universal. Also, it is not uncommon for patients to drop out or otherwise be disqualified from studies, so the cohorts will often be unequal in practice.

The *natural causation coefficient*, denoted $\gamma_{X \to Y}$ or $\gamma$, is defined for $\hat{P}(x)$ equal to the pre-intervention marginal distribution, $P(x)$. This corresponds to an experimental trial where the treatment groups are scaled to be proportional to the relative sizes seen in the observational study.

The *maximum entropy causation coefficient*, denoted $\gamma_{H,X \to Y}$ or $\gamma_H$, is the causation coefficient where $\hat{P}(x)$ is a maximum entropy probability distribution. For random variables with bounded support, this is the uniform distribution and corresponds to equal treatment group sizes.

Other distributions of interventions are possible, to reweigh the effects of certain interventions relative to others in the computation of the causation coefficient. These should be denoted explicitly as $\gamma_{\hat{P}}$. For example, a certain drug may be known

to be helpful in certain small doses, but worse than no treatment at all in larger doses, in which case both the natural and maximum entropy coefficients could be misleading. In such cases, a distribution of interventions corresponding to current best practices may be more informative.

## 3.3 Some properties of the causation coefficient

The causation coefficient is closely related to the average treatment effect and *invariance*, the causal equivalent of independence. This makes the causation coefficient particularly useful for building a taxonomy of possible correlation/causation relationships later in this chapter.

**Definition 3.3.1 (Average treatment effect [12])** *The average treatment effect is the average difference between the outcomes when a patient is treated and when a patient is not treated. For Bernoulli random variables, this is:*

$$ATE(X \to Y) = P(Y = 1 \mid do(X = 1)) - P(Y = 1 \mid do(X = 0))$$

This is the probabilistic causal model equivalent of the Rubin-Neyman-Holland definition of average treatment effect [37]. Positive ATE implies that treatment is, on average, superior to non-treatment, while negative ATE implies the opposite.

**Theorem 3.3.2** *For Bernoulli distributed $X$ and $Y$, $\gamma_{X \to Y}$ is equal to:*

$$\gamma_{X\to Y} = \text{ATE}(X \to Y)\sqrt{\frac{Var[\hat{X}]}{Var_{\hat{X}}[Y]}}$$

**Proof.** Consider the numerator of $\gamma$. For Bernoulli random variables:

$$P(y \mid do(x=1))\hat{P}(x=1) - \hat{P}(x=1)(P(y=1 \mid do(x=1))\hat{P}(x=1) + P(y=1 \mid do(x=0))\hat{P}(x=0))$$

$$= \hat{P}(x=1)(P(y=1 \mid do(x=1)) - \hat{P}(x=1)P(y=1 \mid do(x=1)) - \hat{P}(x=0)P(y=1 \mid do(x=0)))$$

$$= \hat{P}(x=1)(P(y=1 \mid do(x=1)) - \hat{P}(x=1)P(y=1 \mid do(x=1)) - (1-\hat{P}(x=1))P(y=1 \mid do(x=0)))$$

$$= \hat{P}(x=1)(P(y=1 \mid do(x=1)) - P(y=1 \mid do(x=0)) - \hat{P}(x=1)(P(y=1 \mid do(x=1)) - P(y=1 \mid do(x=0))))$$

$$=(P(y=1 \mid do(x=1)) - P(y=1 \mid do(x=0)))\hat{P}(x=1)(1-\hat{P}(x=1))$$

$$=\text{ATE}(X \to Y)Var[\hat{X}]$$

Therefore, $\gamma_{X\to Y} = \text{ATE}(X \to Y)(Var[\hat{X}]/Var_{\hat{X}}[Y])^{1/2}$. $\square$

Since variance is strictly positive for nondegenerate Bernoulli distributions, this implies that $\gamma$ has the same sign as the average treatment effect. In other words, positive $\gamma$ is equivalent to treatment causing, on average, a better outcome than non-treatment, with the opposite being the case for negative $\gamma$.

The definition of the independence of random variables $X$ and $Y$ is: $\forall x, y \;\; P(x, y) = P(x)P(y)$ or, equivalently: $\forall x, y \;\; P(y \mid x) = P(y)$. In other words, observing $X$ provides no information about $Y$ (and vise-versa). The causal equivalent is *invariance* of $Y$ to $X$: $\forall x, y \;\; P(y \mid \hat{x}) = P(y)$; that is to say, no possible intervention on $X$ can affect $Y$ [4]. Unlike independence, invariance is not symmetric. This dissertation suggests the term *mutually invariant* for when both $Y$ is invariant to $X$ and $X$ is invariant to $Y$.

For Bernoulli random variables, $X$ and $Y$ are uncorrelated ($\rho = 0$) if and only if they are independent. The analogous condition holds for the causation coefficient. For Bernoulli distributed $X$ and $Y$, $\gamma_{X\to Y} = 0$ if and only if $Y$ is invariant to $X$.

**Theorem 3.3.3** *For Bernoulli $X$, $Y$, $\gamma_{X\to Y} = 0$ if and only if $Y$ is invariant to $X$.*

**Proof.** Consider the definition of average treatment effect, $\text{ATE}(X \to Y) = P(y = 1 \mid do(x = 1)) - P(y = 1 \mid do(x = 0))$. Average treatment effect is zero if and only if $P(y = 1 \mid do(x = 1)) = P(y = 1 \mid do(x = 0))$. Since the support of a Bernoulli random variable is $\{0, 1\}$, both probabilities must be 0.5; therefore, $Y$ invariant to $X$. Since $\gamma$ has the same sign as the average treatment effect, $\gamma_{X\to Y} = 0$ if and only if $Y$ is invariant to $X$. $\square$

Note that both the correlation and causation coefficients have difficulty capturing nonlinear relationships between variables. In general, independence implies $\rho = 0$ and invariance implies $\gamma = 0$, but the converse does not hold for many distributions.

As a simple example, Table 3.1 contains interventional distributions where $Y$ is not invariant to $X$, but the maximum entropy causation coefficient $\gamma_H = 0$. The natural causation coefficient may be positive, negative or zero depending on the observational (pre-intervention) distribution $P(x)$.

| $P(y \mid \hat{x})$ | y=0 | y=1 |
|---|---|---|
| x=-1 | 1/3 | 2/3 |
| x=0 | 2/3 | 1/3 |
| x=1 | 1/3 | 2/3 |

Table 3.1: Non-invariant interventional distributions where $\gamma_H = 0$

## 3.4 Example: treatment of kidney stones

As noted in the Background chapter, randomization of an independent variable effectively 'cuts' all incoming edges to that node in a causal diagram, removing potential confounding variables. In the context of a randomized controlled trial, the correlation coefficient and causation coefficient coincide; an estimate of one is an estimate of the other.

When the available data is *not* from a randomized controlled trial, an estimate of the causation coefficient can be thought of as an estimate of what the correlation coefficient would be in a randomized controlled trial. As an example, Table 3.2 is a

summary of data from a non-experimental study on the treatment of kidney stones [11]. The subgroups ($Z$) refer to kidney stone size. The study can be modeled with binary treatment ($X$) and response ($Y$) variables, with the decision to perform percutaneous nephrolithotomy (PCNL) as $X = 0$ and surgery as $X = 1$; similarly, failure to recover and recovery are modeled as $Y = 0$ and $Y = 1$, respectively.

| | Small | Large | *Overall* |
|---|---|---|---|
| Open surgery | 81/87 (0.93) | 192/263 (0.73) | *273/350 (0.78)* |
| PCNL | 234/270 (0.87) | 55/80 (0.69) | *289/350 (0.83)* |
| *Overall* | *315/357 (0.88)* | *247/343 (0.72)* | **562/700 (0.80)** |

Table 3.2: Success rate of treatment for kidney stones; successful/total (probability)

The naive model is that kidney stone size does not affect treatment or recovery; this assumption corresponds to the causal diagram in Figure 3.1(a). In such a case, the natural causation coefficient equals the population correlation coefficient; with this dataset, the estimate of these coefficients would be $\approx -0.057$. This is also the case for Figure 3.1(b), where $Z$ partially mediates the treatment's effect on recovery. Adjusting for $Z$ would be incorrect; intuitively, this would 'block' $Z$'s effect on recovery and result in biased estimates of causal effect.

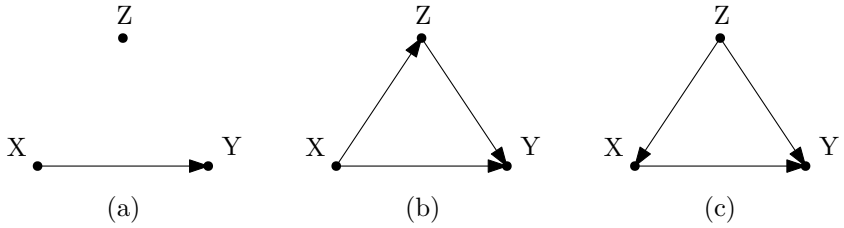



Figure 3.1: Some of the possible causal diagrams for modeling kidney stone treatment. In (a), $Z$ does not affect treatment or response. In (b) $Z$ (partially) mediates the treatments effect on recovery. In (c), $Z$ causally effects treatment and recovery, and an adjustment for direct causes should be performed.

In reality, the size of kidney stones ($Z$) affected which treatment was per-

formed; that is to say, Figure 3.1(c) is the correct causal diagram. Correctly estimating $P(y \mid do(x))$ requires an adjustment for direct causes (Theorem 2.6.2), i.e. $P(y \mid do(x)) = \sum_z P(y \mid x, z)P(z)$. For this dataset, the estimate of the causation coefficient is $\approx 0.068$. Since the cohorts are equal in size, the natural causation coefficient and maximum entropy causation coefficient are identical.

Note that this is the opposite sign as the correlation coefficient (equal to the causation coefficient in the 'naive' model). Since the causation coefficient has the same sign as the average causal effect, this suggests that open surgery ($X = 1$) is the superior treatment.

This 'reversal' effect caused by conditioning on a subgroup is well-known as Simpson's paradox and requires causal information to resolve correctly [58]. There is a subtlety worth addressing: Simpson's paradox refers the phenomena where the association between a pair of variables $(X, Y)$ reverses sign upon conditioning on a third variable, $Z$, regardless of the value of $Z$. However, the existence of such a variable does not imply that it *should* be conditioned on. This is closely related to the problems of 'p-hacking' in the scientific literature, in which researchers selectively analyze or collect data until a publishable result is found, which is a misreporting of the true effect sizes [32].

## 3.5 Taxonomy of correlation/causation relationships

For Bernoulli random variables, it is possible to exhaustively characterize the possible relationships between $X$ and $Y$, given the sign of $\rho$ and $\gamma$. These have impor-

tant interpretations: positive $\rho$ is positive correlation, i.e. $X$ and $Y$ have a positive relationship in observational studies, and positive $\gamma$ is positive causation, which has the interpretation that treatment, on average, is superior to non-treatment.

The causation coefficient provides a unified way to consider all of the possible relationships at once. $\rho$ and $\gamma$ can each be positive, negative or zero which implies 9 possible relationships. These are grouped under 5 categories in the following table.

| | $\rho$ | $\gamma$ |
|---|---|---|
| invariant and independent | 0 | 0 |
| common causation | +/- | 0 |
| inverse causation | +/- | -/+ |
| unfaithful | 0 | +/- |
| genuine causation | +/- | +/- |

Table 3.3: Correlation/causation relationship by sign of coefficients

In this table, "+/-" refers the coefficient taking on a positive or a negative value, e.g. inverse causation refers to a model producing positive $\rho$ and negative $\gamma$ *or* negative $\rho$ and positive $\gamma$. Many of the relationships described in the following section are well-known and existing names are used whenever appropriate, along with examples from existing studies. In addition, for each possible relationship, a simple causal model including 3 Bernoulli distributed variables (treatment $X$, response $Y$ and confounder $Z$) that produce the described relationship to demonstrate that all of these outcomes are possible, even for the simplest confounded models. All of the following models are compatible with the causal diagram in Figure 3.1(c).

Note that the taxonomy is based on *population* coefficients, i.e. the relationships that will persist, even in the limit of infinite data samples. Other correlation/causation relationships that lie outside this taxonomy are discussed at the

summary at the end of this chapter.

### 3.5.1 Invariant and independent

Two variables that are invariant and independent are completely unrelated — neither observing nor manipulating one can provide information about or change the other. Invariance and independence is usually the default assumption when studying a system; in hypothesis testing, the null hypothesis is "no effect". The notion of light cone provides an example familiar to physicists - the principle of locality and the theory of special relativity imply that no object outside of our light cone can ever affect us.

Invariant and independent variables can be trivially mathematically modeled. I introduce such an example to demonstrate how I will model the other possible relationships between correlation and causation. Let $\epsilon_X, \epsilon_Y, \epsilon_Z$ be fair coins.[1] These are the model's background variables, i.e. the random factors outside of the model that determine the variables within the model.

$X$ will generally model a cause or treatment and $Y$ an effect or response. An example model with invariant and independent $X$ and $Y$ is simply:

$$Z = \epsilon_Z$$

$$X = \epsilon_X$$

[1] A 'fair coin' is commonly used in probability and statistics to refer to independent Bernoulli distributed random variables with $p = 0.5$

$$Y = \epsilon_Y$$

| $P(x,y)$ | y=0 | y=1 | $P(x)$ |
|---|---|---|---|
| x=0 | 1/4 | 1/4 | 1/2 |
| x=1 | 1/4 | 1/4 | 1/2 |
| $P(y)$ | 1/2 | 1/2 | |

Table 3.4: Observational distribution of invariant and independent model

| $P(y\|\hat{x})$ | y=0 | y=1 |
|---|---|---|
| x=0 | 1/2 | 1/2 |
| x=1 | 1/2 | 1/2 |

Table 3.5: Interventional distributions of invariant and independent model

$X$ and $Y$ are clearly invariant and independent and the correlation and causation coefficients are 0. In other models, $Z$ will act as a confounding variable, but in this case, none of the variables causally effect each other.

### 3.5.2 Common causation

Hans Reichenbach appears to be the first to propose the "Principle of the Common Cause" claiming, "If an improbable coincidence has occurred, there must exist a common cause" [68]. Elaborating on this, he suggests that correlation between events $A$ and $B$ indicates either that $A$ causes $B$, $B$ causes $A$ or $A$ and $B$ have a common cause[2] This philosophical claim naturally suggests the following definition:

[2]There are systems with correlated variables that do not have a common cause. For example, Bell's theorem states that a theory of local hidden variables is incompatible with quantum mechanics. These systems do not respect the causal Markov condition and are excluded from analysis here. Arguably, this correlation without causation is *why* these systems are so often considered counterintuitive.

Common Causation $X$ and $Y$ are said to experience common causation when $X$ and $Y$ are mutually invariant but not independent.

This effect is sometimes referred to as a 'spurious relationship' or 'spurious correlation' — a term Pearson originally coined in [63]. This risks conflating three distinct concepts: the interventional distribution from which $\gamma$ is calculated, the *population* observational distribution from which $\rho$ is calculated, and the *finite-sample* observational distribution, from which the sample correlation coefficient, $r$ is computed. Consider the following scenarios:

- A small number of samples are taken from statistically independent $X$ and $Y$, but due to random sampling errors, the sample correlation coefficient suggests that $X$ and $Y$ are correlated.
- A large number of samples are taken from causally independent $X$ and $Y$, but due to a latent confounding variable, $X$ and $Y$ are correlated.

The second scenario is common causation. The first scenario is spurious correlation due to random sampling error, informally, 'coincidental correlation'. To report such results as indicative of causality is to make two critical errors: conflating the finite-sample observational distribution with the population observational distribution and conflating the observational distribution with the interventional distribution.

An example of a common cause can be found in a study on myopia and ambient lighting at night [65]. Development of myopia (short-sightedness) is correlated with

night-time light exposure in children, although the latter does not cause the former. The common cause is that short-sighted parents are likely to have short-sighted children, and also more likely to set up night-lights.

As a simple example of common causation, consider the following model: Let $\epsilon_X, \epsilon_Y, \epsilon_Z$ be fair coins and $X$, $Y$ and $Z$ be defined by the following three equations:

$$Z = \epsilon_Z$$

$$X = Z \wedge \epsilon_X$$

$$Y = Z \wedge \epsilon_Y$$

| $P(x,y)$ | y=0 | y=1 | $P(x)$ |
|---|---|---|---|
| x=0 | 5/8 | 1/8 | 3/4 |
| x=1 | 1/8 | 1/8 | 1/4 |
| $P(y)$ | 3/4 | 1/4 | |

Table 3.6: Observational distribution of common cause model

| $P(y\|\hat{x})$ | y=0 | y=1 |
|---|---|---|
| x=0 | 3/4 | 1/4 |
| x=1 | 3/4 | 1/4 |

Table 3.7: Interventional distributions of common cause model

From the observational distribution, it is clear that $X$ and $Y$ are correlated ($\rho = 1/3$) and from the interventional distributions, $X$ and $Y$ are invariant ($\gamma = 0$).

### 3.5.3 Inverse causation

A classic veridical paradox observed by Martin Gardner is the relationship between tuberculosis and dry climate [25]. At one point, Arizona, with one of the driest climates in the United States was found to also have the largest share of tuberculosis deaths. This is because tuberculosis patients greatly benefit from a dry climate, and many moved there. This is isomorphic to the treatment of kidney stones example, but with $X$ as location (Arizona vs. not-Arizona), $Y$ as tuberculosis death, and $Z$ as having tuberculosis. The following definition is proposed to characterize this type of scenario.

Inverse causation $X$ and $Y$ are said to experience inverse causation when the correlation coefficient $\rho$ and natural causation coefficient $\gamma$ have the opposite sign.

Inverse causation is of special importance when considering clinical treatment; $\gamma$ has the same sign as the average causal effect. A case of inverse causation is a case where the correct treatment option is the opposite of what a naive interpretation of correlation would suggest.

As a simple example of inverse causation, consider the following model: Let $\epsilon_Z$ be a fair coin and $\epsilon_Y$ be Bernoulli distributed with $p = 3/4$. The following model exhibits inverse causation with $\rho = -1/2$ and $\gamma = 1/4$:

$$Z = \epsilon_Z$$

$$X = Z$$

$$Y = \begin{cases} \neg Z & \text{if } \epsilon_Y = 1 \\ X & \text{if } \epsilon_Y = 0 \end{cases}$$

| $P(x,y)$ | y=0 | y=1 | $P(x)$ |
|---|---|---|---|
| x=0 | 1/8 | 3/8 | 1/2 |
| x=1 | 3/8 | 1/8 | 1/2 |
| $P(y)$ | 1/2 | 1/2 | |

Table 3.8: Observational distribution of inverse causation model

| $P(y\|do(x))$ | y=0 | y=1 |
|---|---|---|
| x=0 | 5/8 | 3/8 |
| x=1 | 3/8 | 5/8 |

Table 3.9: Interventional distributions of inverse causation model

"Inverse causation" is not a standard term. It is suggested here to avoid confusion with other, similar sounding terms. 'Anti-causation' is inappropriate, as 'anti-causal filters' in digital signal processing are filters whose output depend on future inputs. 'Reverse causation' is also inappropriate - this is already in popular use to refer to mistakenly believing that $Y$ has a causal effect on $X$, when $X$ actually causes $Y$.

### 3.5.4 Unfaithfulness

As discussed in the Background chapter, the causal Markov condition entails a set of conditional independences between variables corresponding to nodes in a DAG $G$. Sprites [79] introduced the *faithfulness* assumption (also referred to as *stability* [57]) as the converse.

**Definition 3.5.1 (Faithfulness condition [79])** *A distribution $P$ is* faithful *to a DAG $G$ if no conditional independence relations other than the ones entailed by the Markov property are present.*

This is a global condition, applying to a joint probability distribution and a DAG. I suggest the following condition as the local analogue for two random variables $X$ and $Y$ in a causal model, which can only occur if the (global) faithfulness condition is violated:

Unfaithful $X$ and $Y$ are said to be unfaithful if they are independent but not invariant.

**Theorem 3.5.2** *If $X$ and $Y$ are unfaithful in causal model $M$, then the observational distribution $P$ and causal diagram $G$ associated with $M$ violate the faithfulness condition.*

**Proof.** Assume without loss of generality that $Y$ is not invariant to $X$, then $P(y \mid \hat{x}$ is a non-constant function of $x$. Therefore, $X$ is an ancestor of $Y$ in the associated causal diagram and $X$ and $Y$ are d-connected. However, $X$ and $Y$ are independent, an independence relation not entailed by the Markov condition. Therefore the observational distribution $P$ is not faithful to $G$. $\square$

For Bernoulli random variables, $X$ and $Y$ are unfaithful if and only if $\rho = 0$ and $\gamma \neq 0$.

The following model is a simple example where $X$ and $Y$ are unfaithful. Let $\epsilon_Z, \epsilon_Y$ be fair coins. Then in the following model, $\rho = 0$ and $\gamma = 1/2$:

$$Z = \epsilon_Z$$

$$X = Z$$

$$Y = \begin{cases} \neg Z, & \text{if } \epsilon_Y = 1 \\ X, & \text{if } \epsilon_Y = 0 \end{cases}$$

| $P(x,y)$ | y=0 | y=1 | $P(x)$ |
|---|---|---|---|
| x=0 | 1/4 | 1/4 | 1/2 |
| x=1 | 1/4 | 1/4 | 1/2 |
| $P(y)$ | 1/2 | 1/2 | |

Table 3.10: Observational distribution of unfaithful X and Y

| $P(y\|do(x))$ | y=0 | y=1 |
|---|---|---|
| x=0 | 3/4 | 1/4 |
| x=1 | 1/4 | 3/4 |

Table 3.11: Interventional distributions of unfaithful X and Y

#### 3.5.4.1 Friedman's thermostat and the traitorous lieutenant

As an example of a model that exhibits unfaithfulness, consider "Friedman's Thermostat", comparing a central bank to a thermostat. A correctly functioning thermostat would keep the indoor temperature constant, regardless of the external temperature by adjusting the furnace settings.[3] Observation would show external temperature and furnace settings to be anti-correlated with each other and internal

[3]Friedman introduced the thermostat analogy in the context of a central bank controlling money supply [22]. Its use as a general analogy for correlation and causation has been popularized by Rowe [70].

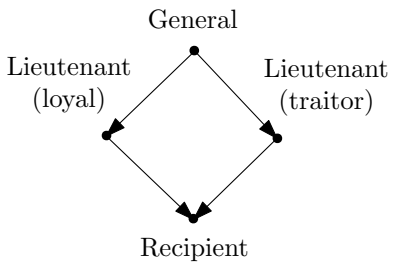



Figure 3.2: The traitorous lieutenant problem

temperature to be uncorrelated with both. This does not correspond to the true causal effect that external temperature and furnace settings have on internal temperature and the lack of a direct causal effect between external temperature and furnace settings. Friedman argued analogously that from the mid 1980s to early 2000s, the Federal reserve successfully controlled money supply to stabilize prices, despite changes in demand for money [22].

The sharp-eyed reader will note that Freidman's thermostat is not a recursive (acyclic) causal model. An example of unfaithfulness with a recursive (acyclic) causal model can be seen in the following "Traitorous Lieutenant" problem. Consider the problem of a general trying to send a one-bit message. The general has two lieutenants available to act as messengers, however, one of them is a traitor and will leak whatever information they have to the enemy. The general observes the following protocol: to send a 1, the general either gives the first lieutenant a 1 and the other a 0, or the first a 0 and the second a 1, with equal probability. To send a 0, the general either gives both lieutenants a 0, or both lieutenants a 1, with equal probability. The recipient of the message XORs both lieutenants' bits to recover the original message.

#### 3.5.4.2 The measure of unfaithful models

Arguably, $X$ and $Y$ being unfaithful is the worst possible case - further study on the relationship between two variables may be considered unwarranted due to the lack of any detectable correlation, despite the existence of a causal effect between the two.

Unfaithful models serves as a counterexample to Tufte's maxim, "Correlation is necessary but not sufficient for causation". However, there is a sense in which it is true. *Almost all* models are faithful in a formal sense - models that do not respect the faithfulness condition have Lebeguse measure zero in probability spaces where model parameters have continuous support and are independently distributed [79]. However, this does not mean that such models can be dismissed out of hand; they are vanishingly unlikely to occur by chance, but can be deliberately engineered, as seen with Freidman's Thermostat and the Traitorous Lieutenant examples.

It may be tempting to conclude that Tufte's maxim holds in practice; that, since unfaithful models have measure zero, they can be considered 'pathological' and reasonably excluded from most analysis. However, there is still an important sense in which this fails; *nearly* unfaithful distributions have nonzero 'surprisingly large' measure[4] [83]. In the limit of infinite samples (i.e. the population distribution), correlation in such models will be nonzero but with a finite number of samples, the level of correlation will be statistically indistinguishable from zero.

[4]Formally, the measure of $\lambda$-strong-unfaithful distributions converges to 1 exponentially in the number of nodes.

### 3.5.5 Genuine causation and confounding bias

The remaining possibility is that the causation and correlation coefficient share the same sign. A special case is when they are *equal*, which, for Bernoulli random variables is equivalent to *no-confounding*; the definition of no-confounding is provided by Pearl [60].

**Definition 3.5.3 (No-confounding)** *$X$ and $Y$ are not confounded if and only if* $P(y|\hat{x}) = P(y|x)$

By the definition of the natural causal coefficient, no-confounding implies $\rho = \gamma$. For Bernoulli random variables, the converse also holds.

When $\gamma$ and $\rho$ are not equal, but share the same sign, then correlation indicates a genuine causal effect, although the strength of the causal effect may be greater or weaker than the magnitude of the correlation coefficient. This dissertation suggests that this class of models be referred to as showing genuine causation (with confounding bias).

Genuine causation with negative confounding bias corresponds to $\gamma > \rho$ and can be thought of as a weaker version of a confounding effect that can produce unfaithfulness or inverse causation. In such cases, the true causal effect will be stronger than correlation suggests. As an example, consider the following model $\rho = 1/2$ and $\gamma = 3/4$. Given $\epsilon_Z$ is a fair coin and $\epsilon_Y$ is Bernoulli with $p = 1/4$

$$Z = \epsilon_Z$$

$$X = Z$$

$$Y = \begin{cases} \neg Z, & \text{if } \epsilon_Y = 1 \\ X, & \text{if } \epsilon_Y = 0 \end{cases}$$

| $P(x,y)$ | y=0 | y=1 | $P(x)$ |
|---|---|---|---|
| x=0 | 3/8 | 1/8 | 1/2 |
| x=1 | 1/8 | 3/8 | 1/2 |
| $P(y)$ | 1/2 | 1/2 | |

Table 3.12: Observational distribution for genuine causation with negative bias model

| $P(y\|do(x))$ | y=0 | y=1 |
|---|---|---|
| x=0 | 7/8 | 1/8 |
| x=1 | 1/8 | 7/8 |

Table 3.13: Interventional distributions for genuine causation with negative bias model

Genuine causation with positive confounding bias example can be thought of as a common cause effect, combined with genuine causation and therefore the true causal effect will be weaker than the correlation suggests. In the following model, $\rho \approx 0.745$, the natural causation coefficient, $\gamma \approx 0.447$ and the maximum entropy causation coefficient, $\gamma_H = 0.5$. Given $\epsilon_X, \epsilon_Y, \epsilon_Z$ are fair coins:

$$Z = \epsilon_Z$$

$$X = Z \wedge \epsilon_X$$

$$Y = \begin{cases} Z, & \text{if } \epsilon_Y = 1 \\ X, & \text{if } \epsilon_Y = 0 \end{cases}$$

| $P(x,y)$ | y=0 | y=1 | $P(x)$ |
|---|---|---|---|
| x=0 | 5/8 | 1/8 | 3/4 |
| x=1 | 0 | 1/4 | 1/4 |
| $P(y)$ | 5/8 | 3/8 | |

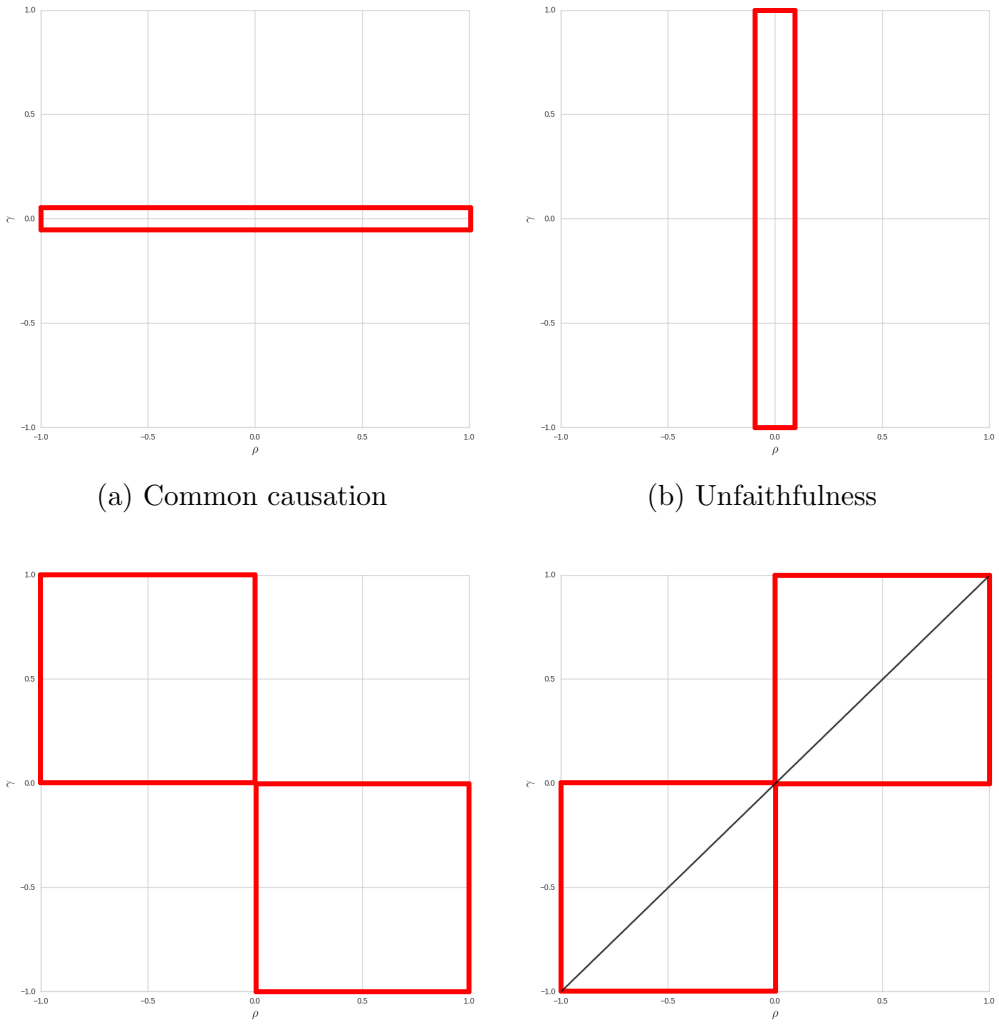
Table 3.14: Observational distribution for genuine causation with positive bias model

| $P(y\|do(x))$ | y=0 | y=1 |
|---|---|---|
| x=0 | 3/4 | 1/4 |
| x=1 | 1/4 | 3/4 |

Table 3.15: Interventional distributions for genuine causation with positive bias model

## 3.6 Visualizing and measuring $\gamma\rho$

Plotting $\gamma$ against $\rho$ provides a way to visualize and measure a distribution of models, forms of *meta*-model analysis. A $\gamma\rho$ plot is a graph where each point represents a single model. The taxonomy of correlation/causation relationships can be visually represented in such a graph. The origin, i.e. $\rho = 0, \gamma = 0$ corresponds to independence and invariance. The horizontal line $\gamma = 0$ corresponds to common causation. The vertical line $\rho = 0$ corresponds to unfaithful models. The upper left and lower right quadrants are models that exhibit inverse causation. The other two quadrants are models that exhibit genuine causation, with the line $y = x$ denoting no-confounding. All of these relationships are shown in Figure 3.3.




(a) Common causation

(b) Unfaithfulness

(c) Inverse causation

(d) Genuine causation

Figure 3.3: $\gamma\rho$ plots visualizing the correlation/causation taxonomy; each point on a plot corresponds to a *model*. The vertical $\gamma$-axis can be seen as strength of causation between two variables in the model, and the horizontal $\rho$-axis the strength of correlation between two variables in the model. For models with Bernoulli random variables, the line $\rho = \gamma$ corresponds to no-confounding.

The example models described in the correlation/causation taxonomy section provide a constructive proof that correlation provides no guarantees about causation, even for models with only three variables. Correlation is indeed not causation, but this does not explain why this mistake is easy to make in practice.

Some insight can be found by considering the space of all linear models with a single confounding variable $Z$. Given jointly independent error terms, $\epsilon_X, \epsilon_Y, \epsilon_Z$ with finite variance and support over the entire real line, this class of linear causal models can be parameterized by $\sigma^2_{\epsilon_X}, \sigma^2_{\epsilon_Y}, \sigma^2_{\epsilon_Z}, \alpha_Z, \beta_X, \beta_Z$

$$Z = \epsilon_Z$$

$$X = \alpha_Z Z + \epsilon_X$$

$$Y = \beta_X X + \beta_Z Z + \epsilon_Y$$

Since these models are linear, and covariance is bilinear, the population correlation coefficient can be calculated analytically, regardless of the underlying distribution of the error terms:

$$\rho_{X,Y} = \frac{\beta_X \sigma^2_{\epsilon_X} + (\alpha_Z^2 \beta_X + \alpha_Z \beta_Z)\sigma^2_{\epsilon_Z}}{\sqrt{(\sigma^2_{\epsilon_X} + \alpha_Z^2 \sigma^2_{\epsilon_Z})(\beta_X^2 \sigma^2_{\epsilon_X} + \sigma^2_{\epsilon_Y} + (\alpha_Z \beta_X + \beta_Z)^2 \sigma^2_{\epsilon_Z})}}$$

The natural causation coefficient can also be calculated directly from the definitions of the causation coefficient and causal effect:

$$\gamma_{X\to Y} = \frac{\beta_X \sigma^2_{\epsilon_X} + \alpha^2_Z \beta_X \sigma^2_{\epsilon_Z}}{\sqrt{(\sigma^2_{\epsilon_X} + \alpha^2_Z \sigma^2_{\epsilon_X})(\beta^2_X \sigma^2_{\epsilon_X} + \sigma^2_{\epsilon_Y} + (\alpha^2_Z \beta^2_X + \beta^2_Z)\sigma^2_{\epsilon_Z})}}$$

The "typical" relationship between correlation and causation can be analyzed by constructing a probability distribution for the parameters of the linear model. $\alpha_Z, \beta_X, \beta_Z$ have support over the entire real line; $\sigma^2_{\epsilon_X}, \sigma^2_{\epsilon_Y}, \sigma^2_{\epsilon_Z}$ have support over $(0, \infty)$. Assuming mean 0 and variance 1, the maximum entropy distributions are $N(0, 1)$ and $\exp(1)$, respectively.

Given these jointly independent distributions over the parameter space, it is possible to sample random linear *models*. This is not drawing random samples of $X$ and $Y$ from a linear model, but rather, drawing random linear models from the space of possible linear models of $X$, $Y$ and $Z$, as described above.

Monte Carlo integration yields estimates of the probability of encountering the possible relationships between correlation and causation, in the class of models being sampled. Specifically, given a random model, the probability that it shows inverse causation $\approx 0.122$, genuine causation with negative bias $\approx 0.364$ and genuine causation with positive bias $\approx 0.514$. A kernel density estimation plot of this model space can be seen in Figure 3.4.

This matches closely to intuition. One would expect that, on average, a strong positive correlation indicates a strong positive causal effect - this can be seen in the upper right quadrant, where the (smooth) density estimation is darkest. Inverse causation is possible, although less likely, and unfaithful models have measure 0, which accounts for their unintuitive nature. It is vanishingly unlikely to encounter

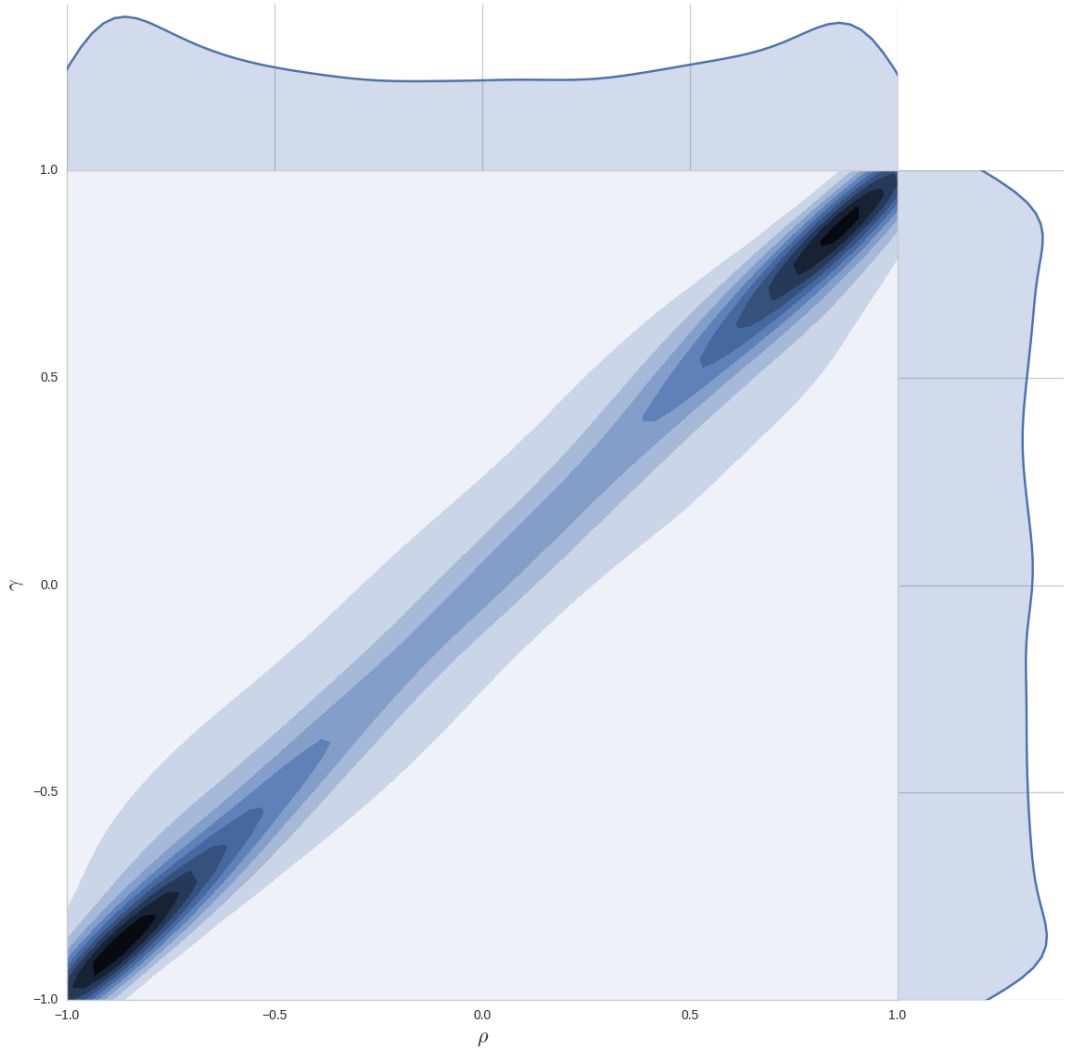



Figure 3.4: A smoothed plot of $\gamma$ vs. $\rho$ calculated for random linear models of 3 variables (treatment $X$, response $Y$ and confounder $Z$). In the majority of models, correlation and causation nearly coincide, especially in the high-density regions in the upper-right and lower-left quadrants. However, there remains a non-trivial percentage of models where correlation and causation do not coincide; in particular, the fraction of models exhibiting inverse causation $\approx 0.122$

an unfaithful model unless it was deliberately created. However, more pessimistically, the measure of models that are *almost* unfaithful is not zero. This is an analysis of population coefficients; finite-sample errors mean that we cannot dismiss unfaithfulness as being irrelevant.

The choice of a maximum entropy distribution in this analysis is based on the principle of maximum entropy, which states that the appropriate prior distribution, given the absence of any other information, is the maximum entropy distribution. This is supported by arguments Jaynes and Wallis [41] [42], although these arguments are by no means universally accepted. However, the important result is not that inverse causation only occurs in $\approx 12\%$ of models but that these results are consistent with intuition that correlation is 'usually' indicative of causation.

$\gamma\rho$ plots are not limited to this particular analysis — they can be used in any analysis where a distribution over model space is available.

## 3.7 Summary of correlation/causation fallacies

Correlation does not imply causation, and, contrary to popular belief, the converse holds as well. Yet, in practice, the two measures are often compatible with each other. The taxonomy presented in this chapter, and the visualization of model space suggests that no single epigram that will suffice to warn researchers about how they can be confused in practice.

Cases of genuine causation (and invariance and independence) correspond to the naive notion that correlation and causation approximately coincide. The rest

of the taxonomy captures the scenarios in which they are not: inverse causation, common causation and, albeit with a vanishingly small measure of models, unfaithfulness; these are the population (Heckman task 2) ways in which correlation and causation fail to coincide. Taking estimation (Heckman task 3) into account adds coincidental/spurious correlation and a *non*-trivial measure of unfaithfulness to the menagerie. Finally, outside of this taxonomy entirely, the existing term 'reverse causation' refers to those scenarios in which the causal effect of $X$ on $Y$ is mistaken for $Y$ on $X$ or vise-versa.

Tufte's maxim, "Correlation is not causation, but it sure is a hint" is tempting. It is true in the sense that a random model will likely have $\rho \approx \gamma$. However, there remains a nontrivial possibility of encountering other correlation/causation relationships such as inverse causation, a problem that no amount of additional data sampling will mitigate. Although it does not directly advance any methods to solve causal inference in practice, the $\gamma\rho$ plot serves the purpose of acting as a visual description of the following principle: data is simply no substitute for accurate causal assumptions.

# Chapter 4: Causal programming (theory)

> The purpose of abstraction is not to be vague, but to create a new semantic level in which one can be absolutely precise.
>
> —Edsger W. Dijkstra

## 4.1 Introduction

Computer science, in general, and programming languages, in particular, are built on abstractions. The essence of a good abstraction is one that isolates the user from the irrelevant, while preserving access to the relevant details. *Declarative* programming abstractions permit expressing the logic of computation without describing the control flow, separating the 'what' of computation from the 'how'. This is programming in the more 'mathematical' sense (e.g. logic programming, linear programming). One indicator as to how successful such a language is in this regard is how well the language corresponds to the mathematical theory it is based on, without requiring any additional statements from the user about how the computation is to be performed.

A related problem, but somewhat in opposition to the goal of declarative programming, is providing performance and completeness guarantees. In general, it be-

comes more difficult for the implementation of more powerful declarative languages to guarantee an efficient runtime, or even guarantee that the underlying computation will terminate. For example, Prolog, despite being based on a restricted subset of first-order logic, is still powerful enough to express arbitrary computation and is thus unable to provide termination guarantees.

This chapter considers the issue of designing effective causal programming abstractions, independent of implementation. One of the key problems is how to group the different mathematical objects used in causal inference into distinct concepts that are compatible with human intuition. This chapter presents baseline abstractions of: model, data/distribution, query and formula, and considers causal inference to be the problem of finding instances of a logical *relation* that satisfy given criteria. This can be viewed as an axis of abstraction that builds on, but is conceptually distinct from, the Heckman Hierarchy. The tasks in Heckman Hierarchy work 'forward' — starting with the definition of models and known probabilities and identifying model parameters. By casting causal inference as a logical relation, it is possible to consider inference in 'any direction', with the goal of unifying different causal inference problems in the same theoretical framework. In this sense, causal programming is a *meta*-theory; this chapter introduces high-level abstractions, which can be instantiated to develop a theory of causal inference that can be implemented in practice.

## 4.2 Causal inference as a logical relation

The main contribution of this chapter is to introduce the *causal inference relation*:

$$\langle M, D, Q, F \rangle_V$$

where:

- $M$ is a set of structural causal *model*s
- $D$ is a set of *distributions*; specifically, a set of known probability functions
- $Q$ is a *query* from the causal hierarchy
- $F$ is a *formula* that computes $Q$ as a function of $D$, for every model in $M$
- $V$ is the set of endogenous variables under consideration

The causal inference relation is indexed by $V$; there is a relation for each set of endogenous variables. $V$ can be thought of as the set of all variables under consideration that can be potentially manipulated and/or measured.

$M$ is a (possibly infinite) set of structural causal models, described by some finite set of model assumptions. The main focus of this dissertation is on the use of causal diagrams, typically denoted $G$, to denote such sets of models. For example, a researcher might consider the set of all structural causal models of the form:

$$X = f_X(\epsilon_X)$$

$$Y = f_Y(Z, \epsilon_Y)$$

$$Z = f_Z(X, \epsilon_Z)$$

where $\epsilon_X \not\perp\!\!\!\perp \epsilon_Y$, that is to say, $\epsilon_X$ and $\epsilon_Y$ are *not* independent. This can represent a variant of the smoking/cancer example, where $X$ is smoking, $Z$ is tar, and $Y$ is cancer, i.e. tar mediates the effect of smoking on cancer, but there may be a latent factor that causally affects both smoking and cancer. This set of structural causal models is represented by the causal diagram in Figure 4.1.

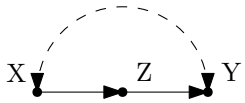



Figure 4.1: A causal diagram where the effect of $X$ on $Y$ is mediated by $Z$. In addition, $X$ and $Y$ share a latent common cause; equivalently the background variables for $X$ and $Y$ are not independent.

In principle, $M$ could be described by any restriction on the set of all recursive structural causal models; this includes such assumptions as conditional independence assumptions between potential outcome variables (e.g. $Y_x \perp\!\!\!\perp X$), or restrictions on the functional relationships, such as the assumption that all $f_i$'s are linear. However, in this dissertation, the set of all causal diagrams over $V$ will serve as the main example of the domain of $M$.

$D$ is a set of known population probability functions, e.g. the joint 'observational' probability function $P(v)$ over all of the endogenous variables. For the model in Figure 4.1, $P(v)$ would be the joint probability function $P(x, y, z)$. Unless otherwise stated, it will be assumed that all probability distributions are *strictly* positive, i.e. $P(v) \neq 0, \forall v$, as this is required by many theorems in causal inference.

$D$ can be extended to include other types of population probabilities. For example, it may be the case that, in addition to the observational probability function, the interventional distributions for a limited subset of variables is available. Such a $D$ can be represented as $P(v \mid do(z')), \forall Z' \subseteq Z$, where $Z$ is some subset of $V$ that can be directly manipulated. Note that this includes the observational probability distribution, as $P(v \mid do(z')) = P(v)$ when $Z'$ is the empty set.

Note that by defining $D$ as a set of known probability *functions*, $D$ also includes information about conditional independences between variables (e.g. knowledge of the probability function $P(x, y)$ includes information about whether $X \perp\!\!\!\perp Y$). When necessary to distinguish between symbolic information about the distribution (e.g. $P(v)$ is known) and the numerical probabilities, this dissertation refers to the former as the *signature* and the latter as the distribution.

$Q$ can be any query from the causal hierarchy: statistical/associational (e.g. $P(y \mid x)$), interventional/causal (e.g. $P(y \mid do(x))$) or counterfactual (e.g. $P(Y_x \mid Z_w)$). The main focus in this dissertation is on causal effect queries, queries of the form $P(y \mid do(x))$, where $y$ and $x$ are disjoint subsets of $V$.

Finally, the formula, $F$, computes $Q$ as a function of $D$, in all models entailed by $M$. In principle, $F$ could be extended to include bounds on probabilities, but the focus of this dissertation is on exact results.

The simplest problem involving the causal inference relation, hereby dubbed "causal checking", is determining whether a given tuple $\langle M, D, Q, F \rangle$ is an instance of the causal inference relation. For example, $M =$ (Figure 4.1), $D = P(x, y, z)$, $Q = P(y \mid do(x))$, and $F = \sum_z P(y \mid x, z)P(z)$ is a valid instance of the relation.

The same tuple, but with $F = P(y \mid x)$ instead is *not* an instance the relation, since this $F$ does not correctly compute the query in all models entailed by $M$.

The causal inference relation provides a general framework for analyzing problems in causal inference. Many problems in causal inference can be seen as finding an instance, or enumerating all the instances, of the causal inference relation that satisfy given criteria. These problems can be broadly categorized by which of $M, D, Q$ are given:

- $M, D, Q$ - Identification: the problem of finding a formula to compute a causal query
- $D, Q$ - Causal discovery: the problem of enumerating the models that are compatible with given population probabilities distributions
- $M, Q$ - Research design: the problem determining the observational and/or experimental data that must be collected to answer a given query
- $M, D$ - Query generation: the problem of enumerating identifiable queries

Note that problems where $F$ is given are not considered, as they represent methodologically suspect practices. For example, searching for $M$, given $D$, $Q$ and $F$ is an attempt to find a *post hoc* rationalization for a calculation of a causal effect that has already been performed.

## 4.3 Identification

Consider the variant of the smoking/cancer model described in Figure 4.1. Furthermore, suppose that an analyst knows the joint pre-intervention distribution, $P(x, y, z)$, and wishes to compute the causal effect of $X$ on $Y$, $P(y \mid do(x))$.

This corresponds to the following problem: find one instance of the causal inference relation such that $M = (\text{Figure } 4.1), D = P(x, y, z), Q = P(y \mid do(x))$. An appropriate $F$ can be identified using the rules of the causal calculus (Theorem 2.8.1). A full solution to this problem is $\langle M, D, Q, F \rangle$ where $M, D$ and $Q$ are as given, and $F$ is:

$$\sum_{z} P(z \mid x) \sum_{x'} P(y \mid x', z) P(x')$$

It is possible for there to be several instances of the causal inference relation that satisfy given criteria. For example, the instances of the causal inference relation that satisfy $M = (\text{Figure } 4.2)$, $D = P(x, y, z)$, $Q = P(y \mid do(x))$ includes solutions $\langle M, D, Q, F_1 \rangle$ and $\langle M, D, Q, F_2 \rangle$, where $F_1$, again, is $\sum_z P(z \mid x) \sum_{x'} P(y \mid x', z) P(x')$, and $F_2$ is simply $P(y \mid x)$. In an identification problem, the solutions are equivalent in the sense that for a given $D$, they will all compute the same value for the query. However, there are other causal inference problems where finding multiple solutions is of interest.

Conversely, there may be no instances of the causal inference relation that satisfy given criteria. Attempting to find an instance of the causal inference relation


Figure 4.2: A causal diagram without latent confounding. Unlike the causal diagram in Figure 4.1, there is no latent common cause of $X$ and $Y$. The causal effect $P(y \mid do(x))$ is equal to the conditional probability $P(y \mid x)$.

that satisfies $M =$ (Figure 4.3), $D = P(x, y, z)$, $Q = P(y \mid do(x))$ will fail; $Q$ cannot be uniquely computed in all models entailed by $M$.

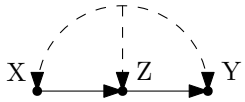



Figure 4.3: A causal diagram where $P(y \mid do(x))$ is not identifiable. Intuitively, there is a latent common cause that affects $X$, $Y$ and $Z$, so it is not possible to determine if any observed covariation is due to the effect of $X$ on $Z$ on $Y$, or if the common cause is responsible.

Treating the full tuple $\langle M, D, Q, F \rangle$ as the solution, as opposed to just the formula, $F$, may seem redundant for identification problems. The utility of this approach becomes more apparent for less restrictive search criteria.

## 4.4 Causal discovery

If a causal diagram is not specified, then causal inference becomes a problem of causal discovery. As a simple example, consider an analyst that is studying a system with just two endogenous variables, $X$, and $Y$. Suppose the analyst knows that the variables are dependent, knows the joint observational probability function, i.e. $D = P(x, y)$, where $X \not\perp\!\!\!\perp Y$, and wishes to infer the causal effect of $X$ on $Y$, i.e. $Q = P(y \mid do(x))$.

A causal diagram and probability function are said to be Markov compatible if the probability function respects the conditional independences implied by

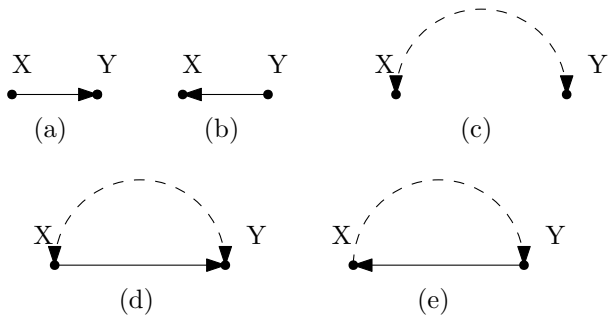



Figure 4.4: Causal diagrams that are Markov compatible with $X \not\perp\!\!\!\perp Y$.

the Markov condition (Definition 2.6.1). There are three causal diagrams that are Markov compatible with $D$ that also permit identification of $Q$: $M_1 =$ (Figure 4.4a), $M_2 =$ (Figure 4.4b), $M_3 =$ (Figure 4.4c). This corresponds to the following instances of the causal relation: $\langle M_1, D, Q, F_1 \rangle$, $\langle M_2, D, Q, F_2 \rangle$, $\langle M_3, D, Q, F_3 \rangle$ where:

$$F_1 = P(y \mid x)$$

$$F_2 = F_3 = P(y)$$

If the domain of $M$ is limited to the space of *Markovian* causal diagrams, then this set of solutions is also complete, in the sense that every causal diagram that is Markov compatible with $D$ is contained in one of the enumerated instances of causal inference relation ($\langle M_1, D, Q, F_1 \rangle$ and $\langle M_2, D, Q, F_2 \rangle$). However, if the domain of $M$ also includes semi-Markovian causal diagrams, then there are several causal diagrams that are compatible with $D$ that do not permit identification of $Q$.

Note that any causal diagram where all endogenous variables share a common, latent cause is Markov compatible with every joint observational probability function $P(v)$. This has consequences for interpreting the results of causal discov-

ery. It is generally incorrect to treat causal discovery as definitively determining the causes of variables in a system. Instead, a discovered model can be viewed as a set of additional, compatible assumptions that will permit answering a given query. Causal discovery will usually be incomplete, since non-identifiable models remain a possibility, unless explicitly ruled out by domain knowledge.

Causal discovery algorithms generally rely on the assumption that $P$ is faithful to $G$ (this condition is also called 'stability' [57]), which is the assumption that every conditional independence relationship that is true in $P$ is entailed by the Markov condition [79]. For example, if $I = P(x, y)$, where $X \perp\!\!\!\perp Y$, then $P$ is Markov compatible with every diagram in figure 4.4. However, $P$ is not faithful to any of these diagrams; intuitively, the edges between $X$ and $Y$ suggest a dependency between the variables that is not present.

## 4.5 Research design

If the data/distribution is not specified, then causal inference becomes a problem of *research design*. As an example, consider a scenario where an analyst wishes to calculate $P(y \mid do(x))$ with respect to the causal diagram in figure 4.5.

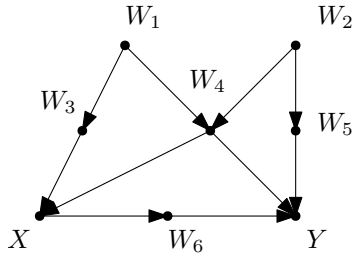



Figure 4.5: A causal diagram, adapted from [56], that permits identifying $P(y \mid do(x))$ in multiple ways.

The complete joint observational probability function $P(v)$ is sufficient, but unnecessary. In particular, an analyst may be interested in calculating causal effect from less information when it is expensive, or otherwise difficult to obtain the complete joint observational probability function. $P(y \mid do(x))$ can be computed as either:

$$F_1 = \sum_{w_3, w_4} P(y \mid w_3, w_4, x) P(w_3, w_4)$$

$$F_2 = \sum_{w_4, w_5} P(y \mid w_4, w_5, x) P(w_4, w_5)$$

Solutions are sensitive to the domain and representation of $D$. One possible representation of $D_1$ is $P(y \mid w_3, w_4, x), P(w_3, w_4)$. However, this implies a somewhat cumbersome domain for $D$ and can make it difficult to determine equivalent distributions. For example, the probability functions $P(y \mid x), P(x)$ are semantically, but not syntactically, equivalent to $P(x, y)$. A less expressive, but simpler domain for $D$ is the set of joint observational probability functions over subsets of $V$. This domain has a natural partial order: $P(v_1)$, is included in a more general distribution, $P(v_2)$, if $V_1 \subset V_2$. In this context, *minimal* distributions to calculate $P(y \mid do(x))$ are $D_1 = P(x, y, w_3, w_4)$ and $D_2 = P(x, y, w_4, w_5)$.

## 4.6 Query generation

If the query is not specified, then causal inference becomes a problem of query generation. Note that the number of identifiable queries has the potential to be very large. For example, if $D = P(v)$, and $M$ is a Markovian causal diagram, then all queries of the form $P(y_1, \ldots, y_m \mid do(x_1, \ldots, x_n))$ are identifiable, which is exponential in $|V|$. Tractable query generation will generally require some restriction on the space of queries or a willingness to accept an incomplete set of solutions.

As a simple example of query generation, consider the problem of generating all queries that either involve the causal effect on $Y$, i.e. $P(y \mid do(\ldots))$ or involve manipulating $x$, i.e. $P(\ldots \mid do(x)))$, with $M =$ (figure 4.6) and $D = P(v)$. Two such queries are identifiable: $Q_1 = P(y \mid do(x))$, and $Q_2 = P(z \mid do(x))$.

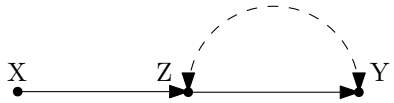



Figure 4.6: A semi-Markovian causal diagram that permits identifying $P(y \mid do(x))$ and $P(z \mid do(x))$ but not $P(y \mid do(z))$

Query generation can be combined with the other causal inference tasks. For example, starting with a known joint probability function, causal discovery can enumerate Markov compatible models, with query generation to enumerate identifiable queries for each Markov compatible model.

## 4.7 The Causal and Heckman Hierarchies revisited

Much like how the Causal Hierarchy outlines the possible questions that may be asked in the course of causal inference, and the Heckman Hierarchy outlines distinct tasks arising from the analysis of causal models, causal programming outlines the different types of different causal inference *goals*. Causal programming acts as a separate axis of abstraction, one that builds upon, unifies and extends existing problems and concepts in causal modeling and inference.

The causal programming concept of a model corresponds precisely to the first causal inference task in the Heckman Hierarchy and the causal programming concept of a query corresponds precisely to the Causal Hierarchy. Identification (Heckman task 2) problems are 'forward' inference in causal programming: starting with the definition of a model, the signature of a distribution, and a query to infer a formula. Evaluating the formula for a given empirical distribution corresponds to estimation (Heckman task 3).

Causal programming's formulation of causal discovery is named after the existing literature on causal discovery algorithms [66]. This roughly corresponds to two 'backwards' steps in the Heckman Hierarchy. First, using real data (part of Heckman task 3) to determine conditional independences (i.e. properties of population distributions, part of Heckman task 2). Then, given these conditional independence assumptions, generating a set of compatible causal models (i.e. Heckman task 1). The traditional definition of causal discovery is simply the problem of finding compatible causal models, given probability distributions. Causal programming

further assumes that the ultimate goal is to actually answer some given query. Each $\langle M_i, D, Q, F_i \rangle$ tuple provides *context* to the solution; each compatible model $M_i$ is a model compatible with the given distribution, and the corresponding $F_i$ identifies the query for *that* set of compatible assumptions.

Causal programming's formulation of research design is inspired by the existing problem of experimental design, which is usually considered part of the third causal inference task: considering problems of sensitivity and statistical power. Causal programming's research design is the rough equivalent, but for Heckman's second task. Instead of experimental design's consideration of how to most effectively use existing data, research design is the question of what data a researcher should try to obtain.

Finally, query generation can be seen as an extension of determining the statistical implications of a model, e.g. reading conditional independence properties from a graph via the d-separation criterion. Causal programming's formulation of query generation extends this to the problem of generating identifiable queries, in general.

## 4.8 Restricted causal inference relation

The causal inference relation can be useful as a conceptual framework, but it is not a practical way of analyzing causal inference problems unless the domains of $M$, $D$ and $Q$ are appropriately restricted. Note that if $M$ can include arbitrary model assumptions, then conducting causal inference may require invoking arbitrary

mathematical theorems!

Several previously studied problems can be cleanly expressed as special cases of finding instances of the causal inference relation. In particular, identification has several subproblems that permit complete algorithms, in the sense that if it is possible to identify $Q$ from $M$ and $D$, then the algorithm is guaranteed to find an appropriate $F$. This is illustrated as follows: Let $G$ be a Markovian or semi-Markovian causal diagram, $P(v)$ be the joint observational probability function, and $W$, $X$, $Y$, and $Z$ each be subsets of $V$:

- Causal effect identification (ID) [39, 76]: $M = G$, $D = P(v)$, $Q = P(y \mid do(x))$
- Conditional causal effect identification (IDC) [75]: $M = G$, $D = P(v)$, $Q = P(y \mid w, do(x))$
- Causal effect identification via surrogate experiments (zID) [5]: $M = G$, $D = P(v \mid do(z')), \forall Z' \subseteq Z$, $Q = P(y \mid do(x))$

A zIDC algorithm, combining the capabilities of IDC and zID, would correspond to $M = G$, $D = P(v \mid do(z')), \forall Z' \subseteq Z$, $Q = P(y \mid w, do(x))$. Finding a complete algorithm for zIDC appears to be an open problem.

Causal discovery can be performed with Inductive Causation (IC) [62]. Given a probability distribution $P$ and assuming faithfulness, IC outputs a *pattern*, which denotes an equivalence class of causal diagrams. If the underlying model is known to be Markovian, then IC is also complete, in that the resulting pattern will correspond to the complete set of causal diagrams that are Markov compatible with $P$.

Otherwise, IC will produce a pattern that includes many, but not all, compatible semi-Markovian models.

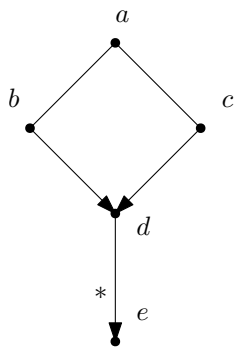



Figure 4.7: A marked pattern [57]. Marked edges denoted by an asterisk, e.g. $d \to e$, signify a directed edge in the underlying model. Directed edges, e.g. $b \to d$, represent either $b \to d$ or a latent common cause of both $b$ and $d$. Undirected edges, e.g. $a$ — $b$, represent either $a \leftarrow b$, $a \to b$, or a latent common cause.

Inductive Causation leaves the details of some its steps unspecified. In particular, IC requires searching for a set $S_{ab}$ such that $(a \perp\!\!\!\perp b \mid S_{ab})$ for every pair of variables $a$ and $b$ in $V$, but does not specify how such sets should be found. The PC algorithm [78] is a refinement of IC that runs in polynomial time on fixed-degree graphs. The combination of IC-based algorithms and identification algorithms permit finding instances of the causal inference relation that correspond to known $D$ and $Q$. For example, the combination of PC and ID would permit finding instances of the causal inference relation that correspond to $D = P(v), Q = P(y \mid do(x))$

Problems related to research design have been discussed in the structural causal model literature; for example, Pearl notes that the front-door and back-door criteria permit an analyst a degree of freedom in selecting which set of covariates to adjust for when calculating causal effect [57]. However, the more general problem of finding instances of the causal inference relation corresponding to given $M$ and

$Q$ does not appear to have an existing, standard formulation. Similarly, query generation is implicitly considered in the analysis of identification, but does not appear to have been formulated as a problem in its own right.

All of these problems can be unified as special cases of finding instances of the causal inference relation, with the following domains for $M$, $D$ and $Q$:

- $M$: Markovian and semi-Markovian causal diagrams over $V$. Causal diagrams can be represented as $G = (V, E, C)$ where $(V, E)$ forms a directed acyclic graph and $C$ is a *confounding family* of $V$, corresponding to the dashed edges that represent latent confounding. A Sperner family is a collection of subsets of a given set, such that none of the subsets contain any of the others [87]. This dissertation defines a confounding family of $V$ to be a Sperner family of $V$, with the further requirement that none of the subsets of $V$ are singleton.

- $D$: Distributions that can be represented as $P(w \mid do(z')), \forall Z' \subseteq Z$, for a given $Z \subseteq W$ and $W \subseteq V$. Note that this is simply $P(v)$ when $W = V$ and $Z = \emptyset$. This representation has a natural partial order: a distribution $D_1$ is said to be contained in another distribution $D_2$ if $W_1 \subseteq W_2$ and $Z_1 \subseteq Z_2$.

- $Q$: Queries that can be represented as $P(y \mid w, do(x))$ for a given $w \subseteq V, x \subseteq V, y \subseteq V$, such that $w$, $x$ and $y$ are disjoint sets.

With respect to these domains, this dissertation suggests the following as canonical formulations of causal inference problems:

- Identification (zIDC): Given $M, D, Q$, find one instance of the causal inference

relation $\langle M, D, Q, F \rangle$.

- Causal discovery: Given $D, Q$, enumerate $\langle M_i, D, Q, F_i \rangle$ for distinct $M_i$, i.e. for any two enumerated instances of the causal inference relation, $\langle M_1, D, Q, F_1 \rangle$ and $\langle M_2, D, Q, F_2 \rangle$, $M_1 \neq M_2$

- Research design: Given $M, Q$, enumerate $\langle M, D_i, Q, F_i \rangle$, for distinct, *minimal* $D_i$, i.e. for any enumerated instance of the causal inference relation, $\langle M, D_i, Q, F_i \rangle$, there does not exist another instance $\langle M, D_j, Q, F_j \rangle$ such that $D_i$ is contained in $D_j$

- Query generation: Given $M, D$, enumerate $\langle M, D, Q_i, F_i \rangle$ for distinct $Q_i$

The combination of the axioms of probability theory and the inference rules of Pearl's causal calculus [56] are known to be complete for the ID, IDC and zID problems [39, 75, 5]. I conjecture that they are complete for all of the problems above as well. Furthermore, if a complete zIDC algorithm exists, it would constitute a complete — albeit intractable, for larger $|V|$ — solution for all of these problems. Causal discovery, research design and query generation problems can be reduced to identification problems by instantiating all possible Markov compatible causal diagrams, distributions, or queries, respectively, and running an identification algorithm for each instantiation.

## 4.9 Other domains for the causal inference relation

Other problems in causal inference can be represented in the causal inference relation framework by modifying the domain of the relation appropriately. In particular, the problems of identification of counterfactuals and recovery from selection bias require only minor extensions to $Q$ and $D$, respectively.

Identification of counterfactuals can be represented by extending the domain of possible queries. Let $G$ be a causal diagram, $\alpha$ and $\beta$ be conjunctions of counterfactual events, e.g. $Y_x, Z_w$, in the potential outcomes notation, and $P_*$ be the set of all experiments, i.e. $P(v \mid do(z')), \forall Z' \subseteq V$. With respect to these domains, the following problems are known to have complete algorithms [74]:

- Counterfactual identification (ID*): $M = G$, $D = P_*$, $Q = P(\alpha)$
- Conditional counterfactual identification (IDC*): $M = G$, $D = P_*$, $Q = P(\alpha \mid \beta)$

Selection bias can be represented by extending distributions to include "s-biased" data [6], i.e. $P(v \mid S = 1)$, where S represents a binary indicator of entry into the data pool. For example [3], in studying the effect of a training program on earnings, subjects achieving higher incomes may tend to report their earnings more frequently than those who earn less. Recovery from selection bias is the problem of answering queries about the general population, despite the data being collected under selection bias. This data may be accompanied by unbiased data, $P(t)$, over some subset $T \subset V$. Bareinboim [3] outlines several problems related to selection

bias:

- Selection without external data: $M = G$, $D = P(v \mid S = 1)$, $Q = P(y \mid x)$
- Selection with external data: $M = G$, $D = P(v \mid S = 1), P(t)$, $Q = P(y \mid x)$
- Selection in causal inferences: $M = G$, $D = P(v \mid S = 1), P(t)$, $Q = P(y \mid do(x))$

Complete identification criteria exist for selection without external data. Sufficient criteria and a valid algorithm for selection with external data and selection in causal inferences exist, but are not known to be complete. In particular, identification in the presence of both selection bias and latent confounding (i.e. in semi-Markovian models) is particularly difficult [3].

This formulation of the causal inference relation does not act as an exhaustive survey of existing causal inference methods and algorithms for SCMs. There is no notion of providing bounds for query, when an exact result cannot be computed [12]. Causal diagrams are the only form of model assumptions considered, which excludes parametric assumptions and nonrecursive (i.e. cyclic) systems. And the problem of external validity, i.e. generalizing results to a different environment from which the original data was collected, is not considered [7].

In principle, the relation could be modified to represent these problems, but this would add considerable complexity. Introducing problems into the framework requires careful selection of the domains of $M$, $D$, $Q$ and $F$ to represent the problem of interest, while still permitting a small set of complete inference rules.

## 4.10 Causal programming (optimization)

The causal inference relation casts problems in causal inference as the problem of finding instances of a logical relation. A further generalization is casting causal inference as an optimization problem. The generalized problem is to find optimal instances of the causal inference relation with respect to a cost function:

$$\begin{array}{ll} \text{minimize} & g(M, D, Q) \\ \text{subject to} & \exists F : \langle M, D, Q, F \rangle \\ \text{and} & M \in M^*, D \in D^*, Q \in Q^* \end{array}$$

Where $g$ is a cost function, $\exists F : \langle M, D, Q, F \rangle$ is the statement that there exists a formula such that $\langle M, D, Q, F \rangle$ constitutes an instance of the causal inference relation, and $M^*$, $D^*$, and $Q^*$ are the given domains for models, distributions and queries under consideration.

A natural problem to consider in this framework is the problem of *optimal* research design. For example, consider a scenario where an analyst wishes to calculate $P(y \mid do(x))$ with respect to the causal diagram in figure 4.5. Since $M$ and $Q$ are given, the only degree of freedom is in the domain of distributions; $M^*$ is just a single causal diagram, i.e. $M^* =$ (figure 4.5) and $Q^*$ is just a single query, i.e. $Q^* = P(y \mid do(x))$.

Let the domain of distributions be all joint probability functions over subsets of $V$, i.e. $D^* = P(w), \forall W \subseteq V$, and the cost function, $g(D)$, be a linear cost

function where including each $w_i$ in the joint probability function costs $i$, e.g. the cost of $P(v) = P(x, y, w_1, w_2, \ldots w_6)$ is 21. In this example, the solution would be the instance of the causal inference relation: $\langle M, D, Q, F \rangle$, where $M$ and $Q$ are as given, $D = P(x, y, w_3, w_4)$, and $F = \sum_{x_3, x_4} P(y \mid x, w_3, w_4) P(w_3, w_4)$, with a cost of 7.

As a function of $D$, $g$ can be interpreted as the cost of performing observational and/or experimental research, with the optimal instance of the causal inference relation representing the least expensive way to answer the original query. As a function of $M$, $g$ can be interpreted as the complexity of a model, with the optimal solution representing the simplest set of additional assumptions that permit answering the original query — a formalization of Occam's razor. Finally, as a function of $Q$, $g$ can be interpreted as the (inverse, when minimizing $g$) value of being able to identify a particular query, which can be combined with other causal inference tasks. For example, given a causal model, but not an distribution or query, finding an optimal instance of the causal inference relation would represent the finding the most valuable, identifiable query, and the distribution required to compute it.

## 4.11 Relationship to the scientific method

Consider the steps involved in an idealized, simplified scientific method: Observe. Hypothesize. Predict. Experiment. (Repeat.) This corresponds well to tasks associated with the causal inference relation. Observation corresponds to obtaining a set of observational probabilities functions ($D$). Hypothesizing corresponds to

causal discovery of compatible models ($M$). Prediction requires generating identifiable queries ($Q$). Finally, experimentation corresponds to obtaining interventional probabilities that confirm or deny the prediction. This process can be repeated with the interventional probabilities included in $D$.

In this sense, causal programming is framework for formalizing (part of) the scientific method. At a high level, the abstractions of model, distribution, query and formula are a guide to grouping different mathematical objects used in causal inference. Defining precise domains for each of these abstractions makes it possible rigorously consider questions regarding the soundness and completeness of corresponding causal inference algorithms. What the causal programming framework introduces is a *unified* way of considering a large class of different problems, a foundation for building automated causal inference systems.

It is worth noting that causal programming does not — and is not designed to — consider questions of ontology, i.e. the question of which factors, and *how* these factors should be entered into formal analysis as model variables. The causal inference relation assumes a known, fixed set of endogenous variables $V$. The question of whether or not these variables adequately capture the relevant aspects of a system being studied is beyond the scope of causal programming. In this sense, causal programming is clearly not a complete framework for formalizing the scientific process.

This limitation suggests an additional desideratum for any implementation of causal programming: the resulting system should be interactive, to make it easy for users to iteratively analyze and refine their models. The implementation of the

identification portion of causal programming is the subject of the next chapter.

# Chapter 5: Causal programming (implementation)

> When someone says: 'I want a programming language in which I need only say what I wish done', give him a lollipop.
>
> —Alan J. Perlis

## 5.1 Whittemore

This chapter introduces Whittemore,[1] an implementation of causal programming, focusing on the identification and estimation of interventional queries. Whittemore is implemented as an embedded, domain-specific language in Clojure, a dialect of Lisp. The main significance of Whittemore is that it provides a declarative programming language and interactive system for the full 'pipeline' of causal modeling and inference. A user can start with 'raw' data, declare how it is to be interpreted as a probability distribution, declare their model assumptions and calculate estimates

[1] The Yale shooting problem [31] is a scenario that is difficult to correctly formalize in first-order logic. In the problem, Fred (later identified as a turkey) is initially alive and a gun is initially unloaded. Loading the gun, waiting for a moment, and then shooting the gun is expected to kill Fred. In one solution, Fred indeed dies. In another — also logically correct — solution, the gun counterintuitively becomes unloaded and Fred survives. Similar 'shooting' problems are have been used as examples when describing theories of causality [57] [30].

Samuel Whittemore was an early American farmer and soldier. A monument in Massachusetts is inscribed: "Near this spot, Samuel Whittemore, then 80 years old, killed three British soldiers, April 19, 1775. He was shot, bayoneted, beaten and left for dead, but recovered and lived to be 98 years of age."

of causal effect, all within the same system.

Several of the components that comprise the implementation of Whittemore have been previously implemented elsewhere. I argue Whittemore is unique in its strong emphasis on declarative programming: a user can specify what it is they want computed, without having to specify details of program control flow. In addition, the syntax of the programming language was designed to mimic the corresponding mathematical syntax as closely as possible. The ultimate goal is to make conducting causal inference no more difficult than writing down the corresponding mathematical statements.

Whittemore is similar to other declarative programming languages that are closely based on a mathematical theory. For example, logic/relational programming languages (e.g. Prolog) are largely based on defining formulas in first-order logic [9]. Probabilistic programming can be seen as a language for defining probability distributions, with an operator that implements conditional sampling [49]. As an implementation of the theory of causal programming, Whittemore can be seen as being based on the theory of structural causal models, and comprised of two operators: identification, which finds formulas that compute a causal query of interest, and estimation which applies formulas to transform probability distributions to other probability distributions. The goal of this chapter is to demonstrate the viability of causal programming as a paradigm.

In particular, this required the (new) implementation of a purely functional version of Shpitser's ID algorithm [73] and designing a protocol/interface for probability distributions to enable seamless transitions between identification and esti-

mation tasks. Since Whittemore is based on the ID algorithm, it is complete for the class of queries it supports. If it fails to identify a causal effect, it is because it is impossible to uniquely calculate the causal effect for the given model assumptions.

## 5.2 A motivating example

Before describing the design and implementation of the language, this section presents a motivating example. The implementation of Whittemore has built-in support for use in a Jupyter notebook [44]. The notebook interface automatically renders certain objects as rich output, e.g. as tables or graphs instead of plain text. This provides a user much faster feedback than a typical write-compile-run cycle, or even a read-eval-print-loop interface; a user can type code and immediately be presented with rich output. *All code examples in this section are shown with their automatically rendered outputs.*

These examples use some additional functions that are not part of 'core' Whittemore. To load data, `read-csv` parses and processes a comma-separated values file; `head` returns the first $n$ samples for inspection. To visualize a probability distribution, `plot-univariate` returns a plot of a marginal distribution.

The example in this section is an analysis of the treatment of renal calculi (i.e. kidney stones), using data from a real study [11]. This is the same example as discussed in Chapter 3; `renal-calculi.csv` is a comma-separated value file that contains the relevant data. The first step of analysis is to parse and process the dataset:

```
(define kidney-dataset
  (read-csv "data/renal-calculi.csv"))

(head kidney-dataset 5)
```

| :size | :success | :treatment |
|---|---|---|
| "small" | "yes" | "surgery" |
| "large" | "yes" | "nephrolithotomy" |
| "small" | "yes" | "surgery" |
| "small" | "yes" | "surgery" |
| "large" | "yes" | "nephrolithotomy" |

The `kidney-dataset` is a dataset of 3 variables, each with two possible values. The possible treatments were either open surgery or nephrolithotomy, the possible kidney stone sizes were either large or small, and the final result of treatment was either determined to be success, or failure. This dataset naturally lends to being modeled as a joint categorical distribution.

The `kidney-distribution` (Figure 5.1) is the empirical probability distribution associated with this study. This distribution exhibits Simpson's paradox, which despite being well known in the statistical literature, continues to "trap the unwary" [15] [58]. In this distribution, the probability of success given surgery, i.e. $P(\text{success} = \text{“yes”} \mid \text{treatment} = \text{“surgery”})$, is clearly less than the probability of success given nephrolithotomy[2]:

[2]Note that `(q ...)` is used to represent $P(...)$; `q` stands for "query".

```
(define kidney-distribution
  (categorical kidney-dataset))

(plot-univariate
  kidney-distribution :success)
```

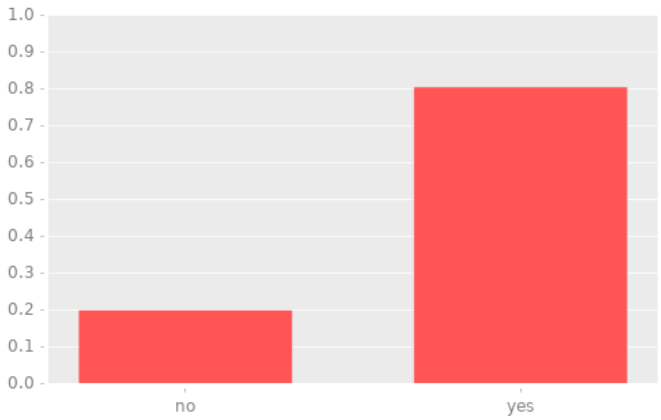



Figure 5.1: The definition of the joint categorical `kidney-distribution` and plot of the marginal distribution of the `success` variable.

```
(estimate kidney-distribution
  (q {:success "yes"} :given {:treatment "surgery"}))
```

0.78

```
(estimate kidney-distribution
  (q {:success "yes"} :given {:treatment "nephrolithotomy"}))
```

0.8257142857142857

However, a reversal appears when conditioning on subgroups. When restricted to observing patients with small kidney stones, surgery appears to be the superior treatment:

```
(estimate kidney-distribution
  (q {:success "yes"} :given {:size "small"
                              :treatment "surgery"}))
```

0.9310344827586207

```
(estimate kidney-distribution
  (q {:success "yes"} :given {:size "small"
                              :treatment "nephrolithotomy"}))
```

0.8666666666666667

When restricted to observing patients with large kidney stones, surgery, again, appears to be the superior treatment:

```
(estimate kidney-distribution
  (q {:success "yes"} :given {:size "large"
                              :treatment "surgery"}))
```

0.7300380228136882

```
(estimate kidney-distribution
  (q {:success "yes"} :given {:size "large"
                              :treatment "nephrolithotomy"}))
```

0.6875

In other words, when looking at small or large kidney stones, surgery appears to be the superior treatment, but when looking at the overall distribution, nephrolithotomy appears to be the superior treatment. Resolving the paradox relies on recognizing that deciding on a superior treatment involves answering interven-

tional (level 2 of the Causal Hierarchy) queries, not associational queries (level 1). The data in this study was drawn from an observational study, not a randomized controlled trial. Specifically, the data was collected under circumstances where doctors were more likely to send patients with larger kidney stones to surgery; in other words, treatment was a *function of* kidney stone size. Success, in turn, was a function of both treatment and size. Given these circumstances, this distribution[3] can be viewed as being generated by the following model:

$$\texttt{size} = f_1(\epsilon_1)$$

$$\texttt{treatment} = f_2(\texttt{size}, \epsilon_2)$$

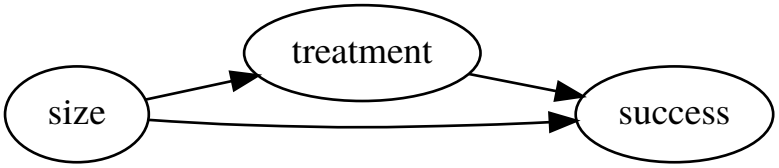
$$\texttt{success} = f_3(\texttt{treatment}, \texttt{size}, \epsilon_3)$$

where each $f_i$ is some (unknown) function and each $\epsilon_i$ is an independent arbitrarily distributed background variable, representing factors outside of the model. For example, success is not solely determined by treatment and kidney stone size. Other factors (e.g. the skill of the surgeon) determine the final outcome, all of which are agglomerated and represented by a single background variable.

Declaring these model assumptions as the `charig1986` model in Whittemore is straightforward. Note that Whittemore automatically generates and renders causal diagram that corresponds to the given model assumptions:

[3]There is a subtlety: It is more accurate to say that the original kidney dataset was drawn from some unknown (and unknowable) population distribution of medical outcomes, i.e. the `kidney-dataset` is a collection of samples. The `kidney-distribution` is the *empirical* distribution associated with these samples and an approximation of the original distribution.

```
(define charig1986
  (model
    {:size []
     :treatment [:size]
     :success [:treatment :size]}))
```




Determining the superior treatment is an interventional query, $P(\text{success} \mid do(\text{treatment}))$. Given the model assumptions in `charig1986`, this can be determined as a function of the `kidney-distribution`, that is to say, a function of the joint probability distribution over `size`, `success`, `treatment`. Whittemore readily identifies the causal effect:

```
(identify charig1986
  (q [:success] :do [:treatment]))
```

$$\sum_{\text{size}} P(\text{size})P(\text{success} \mid \text{size}, \text{treatment})$$

The output of `identify` is a `Formula` object. (Note that Whittemore automatically generates and renders LaTeX math.) The formula correctly calculates $P(\text{success} \mid do(\text{treatment}))$, for every structural causal model compatible with the given causal diagram. In other words, it solves the corresponding identification problem. An identification problem may be of interest in and of itself, however, the goal in this example is not merely find such a formula. The goal is to determine

which treatment is superior. This requires determining the actual interventional probabilities. This is an identification task, followed by an estimation task.

The `infer` function is 'syntactic sugar', combining identification and estimation into a single step:

```
(infer charig1986 kidney-distribution
  (q {:success "yes"} :do {:treatment "surgery"}))
```

0.8325462173856037

```
(infer charig1986 kidney-distribution
  (q {:success "yes"} :do {:treatment "nephrolithotomy"}))
```

0.778875

Despite the literature on resolving Simpson's paradox, it continues to invoke confusion or even outright disbelief [58]. Causal programming offers an alternative solution: a user does not even have to be *aware* of the paradox to calculate the correct causal effect. As seen in this example, causal programming, in general, and Whittemore, specifically, encapsulates the underlying inference algorithms. It provides a declarative language that only exposes those details that are necessary to perform causal modeling and inference.

The rest of this chapter describes Whittemore's syntax, semantics and implementation and includes additional usage examples.

## 5.3 Syntax and semantics

Whittemore is defined as a total (i.e. always terminating), purely functional subset of its host language. The reference implementation of Whittemore is in Clojure, a dialect of Lisp [35]. Like Lisp, there are no statements; a causal program is a sequence of expressions.

An expression in Whittemore is a constant, symbol, or `(`*`op expr`*`*)` where *op* is a causal programming operator, and *expr* is an expression. Operators are described using regular expression syntax: `?` (optional), `*` (0 or more), `+` (1 or more), with non-terminals denoted by *italics*.

⟨*expr*⟩ ::= ⟨*constant*⟩ | ⟨*symbol*⟩ | (⟨*op*⟩ ⟨*expr*⟩*)

⟨*op*⟩ ::= define | model | data | q | identify
| estimate | measure | signature | ⟨*distribution*⟩

Figure 5.2: Whittemore grammar

### 5.3.1 Constants

Constants in Whittemore are same as the host (Clojure) language. Constants include standard atomic data types (e.g. integer and floating point numbers, strings, booleans) as well as keywords, which are symbolic identifiers that evaluate to themselves. Keywords begin with a colon and can contain alphanumeric characters and special characters that are not reserved by the host language, e.g. `:x`, `:x'`, `:treatment`, `:z_1` are all valid keywords.

In addition to the atomic data types, constants include the following collection types, with literal syntax:

- Vectors are ordered collections of values, e.g. `[:x :y]`
- Maps are unordered collections that maps unique keys to values, e.g. `{:x 0, :y 1}`
- Sets are unordered collection of unique values, e.g. `#{:x :y}`.

Keywords and sets are optional data types, in that strings can generally be used in place of keywords, and vectors can be used in place of sets, without significantly affecting the semantics of the program. This opens up the possibility of porting Whittemore to other host languages that do not have the same built-in data types. However, keywords and set notation are preferred in some cases where it is useful to have a visual distinction.

### 5.3.2 Symbols

`(define` *`symbol docstring? value`*`)`

Symbols are identifiers that normally refer to another value. The `define` operator binds a symbol to a value, and returns the value. 'Pure' Whittemore cannot rebind symbols. This restriction is necessary for Whittemore to be a purely functional language.

The implementation of Whittemore slightly relaxes this restriction — rebinding a symbol is a warning rather than an error. Although this makes Whittemore

impure, in practice, it is convenient to be able to redefine symbols, especially for interactive usage.

### 5.3.3 Model

`(model` *dag confounding**`)`

A model corresponds precisely to the concept of a semi-Markovian causal diagram, representing a class of structural causal models. The `model` operator returns a new `Model` where *dag* is a map of variables to their parents, and each *confounding* is a set of endogenous variables whose background variables are not independent (Figure 5.3).

```
(define front-door
  (model
    {:x []
     :z [:x]
     :y [:z]}
    #{:x :y}))
```

$$X = f_X(\epsilon_X)$$
$$Z = f_Z(x, \epsilon_Z)$$
$$Y = f_Y(z, \epsilon_Y)$$
$$\epsilon_X \not\perp\!\!\!\perp \epsilon_Y$$

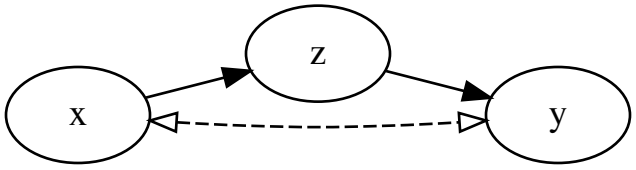



Figure 5.3: An expression defining a model, the equivalent structural causal model written as a system of equations, and the corresponding causal diagram. This set of model assumptions corresponds to $Z$ being a function of $X$ and $Y$ being a function of $Z$. In addition, $\epsilon_X$ $\epsilon_Y$ are not independent; equivalently, $X$ and $Y$ share some latent common cause.

### 5.3.4 Data

`(data` *joint*`)`

`data` produces a *signature* of a probability function, i.e. the symbolic information about a population probability distribution (Section 4.2). Whittemore currently only supports representing knowledge of joint probability functions. For example, knowledge of the joint probability function, $P(x, y, z)$, is represented as `(data [:x :y :z])`.

Note that this is not a representation of a particular probability distribution or a particular dataset — it's the symbolic representation of the joint probability function a researcher expects to be able to obtain. Using Whittemore to conduct inference with an explicit `(data ...)` can only yield symbolic formulas, not numerical estimates.

### 5.3.5 Query

`(q` *effect* `:do` *do*`?` `:given` *given*`?)`

A Query is a statistical or causal query, such that the resulting value is a probability distribution. For example, `(q [:y] :given {:x 1})` corresponds to $P(y \mid X = 1)$, a statistical query. `(q [:y_1 :y_2] :do {:x 0})` corresponds to $P(y_1, y_2 \mid do(X = 0))$, a causal query. Whittemore does not currently support counterfactual queries, although support is planned for a future release.

Note that since *do* and *given* are both optional, they are implemented as keyword arguments in the host language. Their default values are the empty map.

### 5.3.6 Formula

`(identify` *`model data? query`*`)`

The `identify` operator returns a `Formula` that computes *query*, as a function of *data*, in every SCM entailed by *model*, or a `Fail`, if such a Formula does not exist. If unspecified, *data* defaults to the joint observational probability function over all endogenous variables in *model*. For example, `(identify front-door (q [:y] :do {:x 0}))`, with implicit `(data [:x :y :z])`, returns the Formula:

$$\sum_{z}\left[\sum_{x} P(y \mid x, z)P(x)\right] P(z \mid x)$$

where: $x = 0$

Note that `Formulas` follow lexical scoping rules, e.g. only the 'outer' $x$ is bound to 0. The implementation of Formulas is discussed in the "Implementation" section.

The same `identify` expression, but with `(data [:x :y])` returns a `Fail` describing the hedge [74] that renders identification impossible. Since `identify` is based on the ID algorithm [76], it is complete; a Fail will be returned if and only if no appropriate Formula exists.

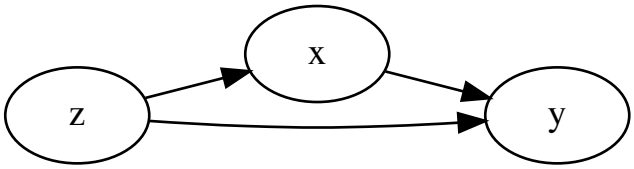
## 5.4 Identification examples

Identification is a purely symbolic task; the resulting formulas map population distributions to causal effects, but, by themselves, do not perform any numerical

calculations. This can still be useful by itself. Identification can inform a researcher if the model assumptions they are willing to make and the data they plan to collect will be enough to infer the queries of interest.

Identification in the kidney stone example is a case of a back-door[4] adjustment [56]; note that the model is isomorphic to the following:

```
(define back-door
  (model
    {:z []
     :x [:z]
     :y [:x :z]}))
```

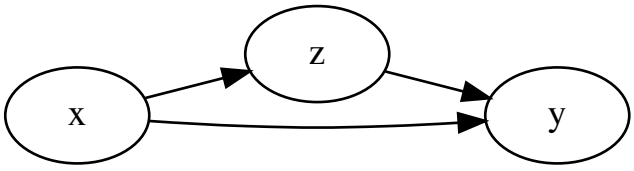



```
(identify back-door
  (q [:y] :do [:x]))
```

$$P(y \mid do(x)) = \sum_{z} P(y \mid x, z)P(z)$$

As expected, Whittemore does *not* try to adjust for variables when doing so would bias the estimate of causal effect:

```
(define mediation
  (model
    {:z [:x]
     :x []
     :y [:x :z]}))
```




[4]The back-door criterion is sufficient graphical criterion for adjustment; a set of variables $Z$ satisfies the back-door criterion relative to $(X, Y)$ if no node in $Z$ is a descendant of $X$ and $Z$ blocks (d-separates) every path between $X$ and $Y$ that contains an arrow into $Y$.

```
(identify mediation
  (q [:y] :do [:x]))
```

$$P(y \mid x)$$

By default, Whittemore assumes that the joint probability distribution function over all endogenous variables will be available, e.g. for endogenous variables $X, Y, Z$, `(data [:x :y :z])`. If this is not the case, then Whittemore performs a *latent projection* [79], converting the causal diagram to one that replaces the unknown variables with latent variables (e.g. dashed, bidirected edges). For example, if only the joint distribution of $P(X, Y)$ is available for the `back-door` model, then identifying the causal effect of $X$ on $Y$ becomes impossible:

```
(identify back-door
  (data [:x :y])
  (q [:y] :do [:x]))
```

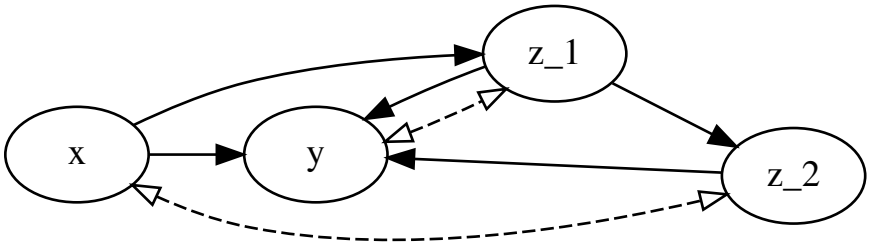
```
#whittemore.core.Fail{}
```

Identification fails (i.e. a `Fail` object is returned) in this case because it is impossible to identify the causal effect $P(y \mid do(x))$ from just the joint probability distribution $P(y \mid do(x))$.

Note that Whittemore is by no means limited to the special cases of back door and front door adjustment [56]. Causal programming easily identifies formulas for computing causal effect that involve non-standard adjustments:

```
(define concomitant-example
  "Figure 1 (f) from (Shpitser 2008)"
  (model
    {:y [:x :z_1 :z_2]
     :z_2 [:z_1]
     :z_1 [:x]
```

```
 :x []}
#{:y :z_1}
#{:x :z_2}))
```




```
(identify concomitant-example
  (q [:y] :do [:x]))
```

$$\sum_{z_1,z_2} \left[ \sum_{x} P(x)P(z_2 \mid x, z_1) \right] P(z_1 \mid x)P(y \mid x, z_1, z_2)$$

This is example is notable in that identifying $P(y \mid do(x)$ *requires* summing over *post* treatment variables — identification will fail if $Z_1$ or $Z_2$ is not available. This is in stark contrast to the recommendation that that one should avoid summing over post-treatment variables to avoid introducing bias [53].

## 5.5 Implementation (identification)

The syntax of probability theory is weakly typed, and heavily overloaded. For example, even when restricted to associational/statistical queries, the symbol $P()$ has several meanings in mathematics. $P(y \mid x)$ is a function from values of $X$ to probability distributions of $Y$; $P(y \mid X = x)$ is a particular probability distribution; $P(Y = y \mid X = x)$ is a particular probability value, i.e. a real number between 0 and 1. In each case, the syntax is identical, but each expression has a different type.

In addition to supporting this syntax, for an implementation of causal programming to be fully declarative, it has to be purely functional — any portion

of code that manipulates state is complected with every other piece of code that manipulates the same state.

Clojure is well suited to implementing causal programming. The language has literal syntax for (immutable) vectors and maps and is dynamically typed which makes it possible for Whittemore to closely match the syntax of the corresponding mathematics. Lisp dialects, in general, blur the line between an API/library and a language. Whitemore is implemented as a library, but acts as a sublanguage — it's possible to code in 'pure' Whittemore, with strong termination and completeness guarantees, while still being able to write more general code in the host language, as necessary.

Currently, the `identify` operator is an implementation of Shpitser's ID algorithm. The ID algorithm and several related algorithms have been previously implemented in the R programming language [81]. Unlike this previous implementation,[5] Whittemore's `identify` is purely functional. In addition, Whittemore natively supports estimation of causal effects (described later in this chapter) and has a strong emphasis on interactive 'notebook' usage.

The Model, Data, Query and Formula types are all implemented as persistent (immutable) hash array mapped tries (HAMT) [2] which support lookup and 'modification' — associating a key and value creates a new data structure (Figure 5.4) — in $\log_{32} N$ time.[6] This provides good performance while remaining free of side

[5] Whittemore was designed independently; I became aware of Tikka and Karven's implementation after beginning work on Whittemore.

[6] In asymptotic analysis, this is no different than the $O(\log N)$ performance of a binary tree. Empirically, HAMTs enjoy performance that rivals other (mutable) implementations of the associative array abstract data type.

effects. A considerable advantage is that the data structures can be freely shared with any other part of a program; it is impossible to corrupt a data structure since none of them can be changed.

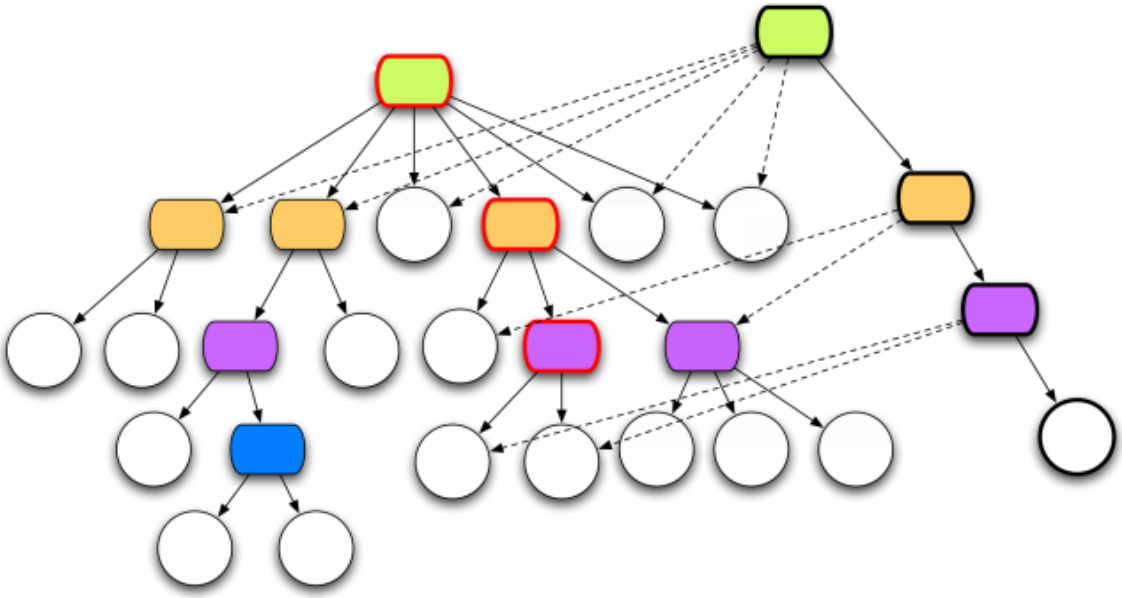


Figure 5.4: The HAMT is effectively an 32-tree, which can be used to implement a map (associative array) data type. 'Modifying' (e.g. changing a value for one of the key-value pairs) an existing map leaves the original map unchanged. A new map is created, one that mostly shares structure with the original for efficiency [35].

Models, i.e. causal diagrams, are represented in a modified adjacency-list format; models are a map, whose keys are `:pa` and `:bi`, and whose values are another map, and a set of sets, respectively (Figure 5.5). These correspond to the two type types of edges that must be kept track of: directed edges and bidirected edges. Directed edges (`:pa`) are represented as a map of vertices to the set of their parents. Bidirected edges (key `:bi`) are implemented as a set of sets representing pairs of edges. When used in a Jupyter notebook, these are automatically rendered as causal diagrams via the Graphviz [24] graph visualization software.

The ID algorithm relies on 'modified' graphs, e.g. $G_{\overline{X}}$, the graph where all of

```
{:pa {:x #{}, :y #{:z}, :z #{:x}},
 :bi #{#{:x :y}}}
```

Figure 5.5: The `front-door` model as a map. $X$ has no parents, $Y$ has the parent $Z$, and $Z$ has the parent $X$. In addition, there exists a bidirected edge between $X$ and $Y$. During interactive 'notebook' usage, models are automatically rendered as causal diagrams (Figure 5.3).

the incoming edges to nodes $X$ are removed. Since all Whittemore data structures are immutable, the 'modified' graph can be safely created without affecting the original.

Formulas are defined as a map of bindings of variables to values, and a *form*, which is defined recursively:

- {:p #{*vars*} :given #{*vars*}}
- {:sum *form* :sub #{*vars*}}
- {:prod #{*forms*}}
- {:numer *form* :denom *form*}

These forms correspond to a probability expression, summation, product, and fraction, respectively. Formulas follow lexical scoping rules, which obviates the need to rename variables — variable bindings are determined the first surrounding `:sum` that contains the variable as a subscript.

Additional keys can be added to the Model, Data, Query, and Formula types, without changing the semantics of a program, permitting considerable future extensibility.

## 5.6 'Nanopass' simplification of formulas

The ID algorithm produces formulas that identify causal effects. However, there is no guarantee that the resulting formulas are particularly understandable or efficient for estimation. Essentially, the ID algorithm is 'unaware' of the rules of probability theory.

One approach to simplifying formulas is to add steps to the ID algorithm to simplify formulas during identification. The ID algorithm is recursive; by interleaving simplification and identification steps, it is possible to dramatically reduce the complexity of the final formula [80].

The disadvantage of this approach is that it requires specific changes to the implementation of the ID algorithm and intertwines identification and simplification. This raises difficulties for future extensions. Instead, Whittemore's approach is to borrow the idea of a 'nanopass' compiler [43]. The 'core' inference performed by the implementation of ID produces unsimplified, but valid formulas. These formulas can then be sent through a pipeline of simplification steps, repeatedly applying pattern-matching rules to reduce the complex formulas into simpler, but still valid formulas. The requirement is that each 'pass' produces a simpler, but still valid formula, preserving correctness, while leaving this process open to customization and extension.

Whittemore's implementation is especially amenable to this approach since Formulas are 'ordinary' Clojure data structures and immutable. It is safe to 'change' a formula because it is not a true change — it produces a logically new formula data

structure. The implementation shares structure for efficiency, but this is safe to do so because the interface does not permit in-place modification. It is also easy to take advantage of parallelism. There is no risk to sharing the same formula among separate threads, because there is no state that can be corrupted via race conditions.

As an example, the formula $\frac{P(x,y,z)}{\sum_y P(x,y,z)}$ is represented by the following map:

```
{:numer {:p #{:x :y :z}},
 :denom {:sub #{:y},
         :sum {:p #{:x :y :z}}}}
```

This formula can be reduced to `{:p #{:y} :given #{:x :z}}` by applying a marginalization rule on the `:denom` form, resulting in `{:numer #{:y :z :x} :denom :p #{:z :x}}`. This, in turn, can be reduced to `{:p #{:y} :given #{:x :z}}` by applying a conditional probability rule. This final formula is a representation of $P(y \mid x, z)$.

The 'nanopass' approach respects separation of concerns in the implementation: the tasks of identification and simplification are kept entirely separate. An advantage of this approach is that future extensions to causal programming that implement additional identification algorithms will get any improvements on the compilation pipeline for 'free'. For example, to add support for for conditional causal effect queries (e.g. $P(y \mid z, do(x))$) and/or counterfactual queries (e.g. $P(Y_x \mid x')$), the implementation of `identify` could be updated to the IDC or IDC* [74] algorithm. The output of an updated `identify` can be sent through the same simplification pipeline, requiring no additional changes to Whittemore.

## 5.7 Estimation and the distribution protocol

Causal diagrams represent completely nonparametric model assumptions. Accordingly, the result of running an identification algorithm, given only the assumptions in a causal diagram, will also be nonparametric. A `Formula` calculates a causal effect query as a function of probability functions. For example, in the `front-door` model example, $P(y \mid do(x))$ is identified by the formula $\sum_z P(y \mid x, z)P(z)$, which is a function of the conditional probability function $P(y \mid x, z)$ and the marginal probability function $P(z)$.

The structure of $P(y \mid x, z)$ and $P(z)$ depends on the type of probability distribution being represented. Estimating causal effects, i.e. calculating numerical probabilities, requires specific knowledge of this representation. This is at odds with the nonparametric nature of a `Formula`.

To solve this 'impedance mismatch' between identification and estimation, Whittemore treats the problem of evaluating a formula to be part of the definition of a probability distribution. There is no single piece of code describing how a `Formula` evaluates probabilities, that would limit Whittemore to a single method of estimation. The design relies on the definition and implementation of a probability distribution to describe how it applies a `Formula` to itself. `Formula` code remains distribution-agnostic and the implementation of estimation methods and the representation of probability distributions can be open to user extension.

Conceptually, formulas are treated as transformations from probability distributions to probability distributions, *not* as a means of calculating the probability of

an individual event. As a result, there's no restrictions on how a probability distribution has to be represented in a program. Whittemore merely defines a probability distribution *protocol*. In Clojure, a protocol is a named set of methods and their signatures; the resulting functions are polymorphic in their first argument.[7] The `Distribution` protocol is defined as the following methods:

- `(estimate` *`this formula`*`)`

  Returns the result of applying a *formula* to *this* distribution, yielding a new distribution.

- `(measure` *`this event`*`)`

  Returns the probability of *event*, i.e. `measure` implements the mathematical concept of a probability measure. An *event* is expected to be a map of keywords to values.

- `(signature` *`this`*`)`

  Returns the Data 'signature' of the distribution.

Whittemore includes an implementation of a categorical distribution. Constructing a categorical distribution is done with the `categorical` function which accepts a vector of samples (events) as it's argument and automatically infers the support of the joint distribution. As a simple example:

```
(define example-distribution
  (categorical
    [{:x 0, :y 0}
     {:x 0, :y 1}
```

[7] Clojure protocols are analogous to a Java interface.

```
{:x 1, :y 0}
{:x 1, :y 1}
{:x 1, :y 1}]))
```

creates a joint categorical distribution from a collection of five samples. The resulting probability distribution has $P(x = 1, y = 1) = 2/5$ and $P(x = 0, y = 1) = P(x = 1, y = 0) = P(x = 0, y = 0) = 1/5$ and is represented as a map of the distribution's probability mass function. For categorical distributions, `measure` can be implemented as a simple map lookup, e.g. `(measure example-distribution {:x 0, :y 0})` returns 2/5. The `signature` of this distribution is `{:joint #{:x :y}}`, i.e. a representation that this is a joint distribution over the variables $X$ and $Y$. Neither `signature`, nor `measure` is typically called from user code — Whittemore provides higher-level functions and 'syntactic sugar'.

The Distribution protocol is user extensible; other probability distributions can be implemented in the host language without modification to Whittemore's implementation. The implementation of `estimate` for a new distribution only has to specify how a formula transforms a probability distribution into another probability distribution — the other use cases are also 'syntactic sugar'.

## 5.8 Infer and 'syntactic sugar'

The reference implementation of Whittemore provides some 'syntactic sugar' to make causal programming easier. In particular, the `q` operator has three versions that mimic common usage of $P()$ in probability theory:

- 'Unbound' query, e.g. `(q [:y] :do [:x])`, a query where *do* and *given* are

vectors. An unbound query can still be provided as an argument to `identify`, but the resulting formula cannot be used as an argument to `estimate` without first providing the necessary variable bindings.

- 'Bound' query, e.g. `(q [:y] :do {:x 0})`, corresponding to a conditional or interventional distribution (this is considered the canonical version of a Query).

- 'Event' query, e.g. `(q {:y 1} :do {:x 0})`, corresponding to a specific probability, i.e. *effect* is an event.

Providing an event query to `estimate` implies `measure`. For example, assuming that an appropriate probability distribution is bound to the symbol `smoking`[8]:

```
(estimate smoking
  (q {:y 1} :given {:x 1}))
```

Returns the probability 0.8525.

In addition, Whittemore provides the `infer` operator, which combines the functionality of `identify`, `estimate` and `measure`. For example:

```
(infer front-door smoking
  (q {:y 1} :do {:x 1}))
```

Returns the probability 0.4975.

[8]These examples assume that `smoking` follows the probability distribution in [57, Table 3.1]

# Chapter 6: The computational power of dynamic Bayesian networks

> Decidable problems are priceless; for everything else, there's pattern-matching.
>
> —Meredith Patterson

## 6.1 Cyclic causal models

The theory of causal programming and the implementation in Whittemore were designed to respect somewhat competing principles: a researcher should be able to express a very general class of models, queries and data/distributions, while still allowing implementations to guarantee completeness in inference. In particular, the class of models that is supported is precisely that of acyclic causal diagrams, essentially, the space of 'fully' nonparametric structural causal model assumptions.

Nonparametricity in model specification is a 'liberating' assumption — it allows a researcher to enter the assumption that $X$ causes $Y$, while making no additional claims as to the nature of that relationship. In contrast, requiring acyclic causal diagrams is restrictive, removing a potentially interesting class of models from analysis. This suggests the following related analysis: in what sense the restriction to acyclic graphs is a 'natural' restriction?

The definition of structural causal models (Definition 2.4.2) requires that each $f_i$ equation forms a mapping from $U_i \cup PA_i$ to $V_i$; each equation assigns a value to its corresponding endogenous variable as a function of its direct causes and the entire set of equations has a unique solution. The nonparametric nature of the equations is essentially the source of problem. A system of nonlinear equations may have zero, one, or multiple solutions — this last case is particularly problematic. In the case of zero solutions, it is reasonable to claim that the model is simply inconsistent. In the case of multiple solutions, a probability over the background variables no longer uniquely induces a probability distribution over the endogenous variables.

It is illuminating to consider what cyclic models are generally designed to analyze: the *equilibrium* distribution of variables evolving over time. Cases of mutual causation, e.g. $X$ causing $Y$ and $Y$ causing $X$ can be broken down into cases of variables affecting their next time-step: $X_t$ causing $Y_{t+1}$ and $Y_t$ causing $X_{t+1}$. A cyclic causal model is 'shorthand' for a model unrolled over time.

This chapter argues that extending causal analysis to cyclic causal diagrams is *fundamentally* difficult. In particular, it presents a proof that the equilibrium distribution of dynamic Bayesian networks is uncomputable by showing that such networks can simulate arbitrary computation.

This has consequences for any attempt to extend causal programming to support cyclic models: if the equilibrium distribution of such models in uncomputable, then it is impossible to design complete algorithms for conducting inference. Informally, the causal programming abstraction 'breaks'. With nonparametric, acyclic models, it is possible to implement a programming language that is guaranteed to

find a formula that computes a given causal query, whenever such a formula exists. When this process fails, it is because such a formula *can not exist* — the model assumptions are provably too weak.

When the causal models, themselves, are capable of performing arbitrary computation, it is impossible to design causal inference algorithms with such a property. It may be the case that, for given model assumptions, a query has a definitive answer. It will not be possible, in general, to design an algorithm that is guaranteed to answer a query, because doing so is equivalent to solving the halting problem. In other words, inference may fail, and it will be unknowable if the failure is because the query has no answer, or if because the execution of the algorithm was simply unable to find the answer. It may be the case that inference needs to be run longer, but it may be the case that it will never succeed.

## 6.2 Dynamic Bayesian Networks

Dynamic Bayesian networks are the time-generalization of Bayesian networks and relate variables to each other over adjacent time steps. Dynamic Bayesian networks unify and extend a number of state-space models including hidden Markov models, hierarchical hidden Markov models and Kalman filters. Dynamic Bayesian networks (DBN) extend Bayesian networks to model a probability distribution over a semi-infinite collection of random variables, with each collection of random variables modeling the system at a point in time [18]. Following the conventions in [54], the collections are denoted $Z_1, Z_2, \ldots$ and variables are partitioned $Z_t = (U_t, X_t, Y_t)$ to

represent input, hidden and output variables of a state space model. Such a network is "dynamic" in the sense that it can model a dynamic system, not that the network topology changes over time.

A DBN is defined as a pair $(B_1, B_\rightarrow)$, where $B_1$ is a Bayesian network that defines the prior $P(Z_1)$ and $B_\rightarrow$ is a two-slice temporal Bayes net (2TBN) that defines $P(Z_t|Z_{t-1})$ via a directed acyclic graph:

$$P(Z_t|Z_{t-1}) = \prod_{i=1}^{N} P(Z_t^i|pa(Z_t^i)) \tag{6.1}$$

where $Z_t^i$ is the $i^{\text{th}}$ node at time $t$, and $pa(Z_t^i)$ are the parents of $Z_t^i$ in the graph. A 2TBN is simply a Bayesian network where the nodes are partitioned into vertices at time $t$ and time $t+1$. The parents of a node can either be in the same time slice or in the previous time slice (i.e. the model is first-order Markov).

The semantics of a DBN can be defined by "unrolling in time" the 2TBN until there are $T$ time-slices; the joint distribution is then given by:

$$P(Z_{1:T}) = \prod_{t=1}^{T} \prod_{i=1}^{N} P(Z_T^i|pa(Z_t^i)) \tag{6.2}$$

Analyzing the computational power of a DBN requires defining what it means for a DBN to accept (and halt) or reject an input. Define an input sequence, $\{U_t\}$ of Bernoulli random variables to model the binary input. Similarly, define an output sequence $\{Y_t\}$ ($Y_t \in \{run, halt_0, halt_1\}$) to represent whether the machine has halted and the answer that it gives. Given an input, $in_1, in_2, \ldots, in_t$, to a decision problem, the machine modeled by the DBN has halted and accepted at time $t$, if and only if

$P(Y_t = halt_1 | U_1 = in_1, \ldots, U_n = in_t) > 0.5$ and halted and rejected if and only if $P(Y_t = halt_0 | U_1 = in_1, \ldots, U_n = in_t) > 0.5$.

## 6.3 Discrete Dynamic Bayesian Networks Are Not Turing-complete

"Discrete" Bayesian networks are Bayesian networks where all random variables have some finite number of outcomes, i.e. Bernoulli or categorical random variables. If dynamic Bayesian networks are permitted to increase the number of random variables in the network over time, then simulating a Turing-machine becomes trivial: simply add a new variable each time step to model a newly reachable cell on the Turing machine's tape. However, this requires some 'first-order' features in the language used to specify the network and the computational effort required at *each step* of the simulation will grow without bound.

With a fixed number of random variables at each time step and the property that DBNs are first-order Markov, the computational effort per step remains constant. However, discrete DBNs have sub-Turing computational power. Intuitively, a discrete DBN cannot possibly simulate a Turing machine since there is no way to store the contents of the machine's tape.

More formally, any discrete Bayesian network can be converted into a hidden Markov model [54]. This is done by 'collapsing' the hidden variables ($X_t$) of the DBN into a single random variable by taking the Cartesian product of their sample space. The 'collapsed' DBN models a probability distribution over an exponentially larger, but still finite sample space. Hidden Markov models are equivalent to probabilistic

finite automata [19] which recognize the stochastic languages. Stochastic languages are in the RP-complexity class and thus discrete DBNs are not Turing complete.

## 6.4 A Dynamic Bayesian Network with Continuous and Discrete Variables

I present a construction for a 2TBN that can simulate the transitions of a two stack push-down automaton (PDA), which is equivalent to the standard one tape Turing machine. A two stack PDA consists of a finite control, two unbounded binary stacks and an input tape. At each step of computation, the machine reads and advances the input tape, reads the top element of each stack and can either push a new element, pop the top element or leave each stack unchanged. The state of the control can change as function of previous state and the read symbols. When the control reaches one of two possible halt states ($\{halt_0, halt_1\}$), the machine stops and its output to the decision problem it was computing is defined which of the halt states it stops on.

A key part of the construction is using a Dirac distribution to simulate a stack. A Dirac distribution centered at $\mu$ can be defined as the limit of normal distributions:

$$\delta(\mu) \equiv \lim_{\sigma \to 0+} \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-x^2}{2\sigma^2}} \tag{6.3}$$

A single Dirac distributed random variable is sufficient to simulate a stack. The stack construction adapted from [77] encodes a binary string $\omega = \omega_1\omega_2 \ldots \omega_n$

into the number:

$$q = \sum_{i=1}^{n} \frac{2\omega_i + 1}{4^i} \tag{6.4}$$

Note that if the string begins with the value 1, then $q$ has a value of at least 3/4 and if the string begins with 0, then $q$ is less than 1/2 - there is never a need to distinguish among two very close numbers to read the most significant digit. In addition, the empty string is encoded as $q = 0$, but any non-empty string has value at least 1/4.

All random variables, except for the stack random variables, are categorically distributed - thus, the conditional probabilities densities between them can be represented using standard conditional probability tables.

Extracting the top value from a stack requires a conditional probability distribution for a Bernoulli random variable ($Top \in \{0, 1\}$), given a Dirac ($Stack \in \mathbb{R}$) distributed parent. The Heavyside step function meets this requirement and is defined as the limit of logistic functions (or, more generally, softmax functions), centered at 1/2:

$$H(x) \equiv \lim_{k \to \infty} \frac{1}{1 + e^{-k(x-1/2)}} \tag{6.5}$$

The linear operation $4q - 2$ transfers the range of $q$ to at least 1 when the top element of the stack is 1 and no more than 0 when the top element of the stack is 0. Then, the conditional probability density function:

$$P(Top|Stack = q) = H(4q - 2) \tag{6.6}$$

yields $P(Top) = 1$ whenever the top element of the stack is 1 and $P(Top) = 0$ whenever the top element of the stack is 0.

Similarly, a conditional probability distribution can be defined for Bernoulli random variable $Empty \in \{0, 1\}$, as:

$$P(Empty|Stack = q) = 1 - H(4q) \tag{6.7}$$

to check if a stack is empty.

Finally, the linear operations $\frac{q}{4} + \frac{2b+1}{4}$ and $4q - (2b + 1)$ push and pop $b$, respectively, from a stack. The conditional probability density for a stack at time $t + 1$, given a stack at time $t$, the top of the stack at time $t$, and action to be performed on the stack ($Action_t \in \{push_0, push_1, pop, noop\}$) is fully described as follows:

$$\begin{aligned}
P(Stack_{t+1}|Top_t = p, Stack_t = q, Action_t = push_0) &= \delta(q/4 + 1/4) \\
P(Stack_{t+1}|Top_t = p, Stack_t = q, Action_t = push_1) &= \delta(q/4 + 3/4) \\
P(Stack_{t+1}|Top_t = p, Stack_t = q, Action_t = pop) &= \delta(4q - (2p + 1)) \\
P(Stack_{t+1}|Top_t = p, Stack_t = q, Action_t = noop) &= \delta(q)
\end{aligned} \tag{6.8}$$

Since there are two stacks in the full construction, they are labeled, at time

$t$, as $Stack_{a,t}$ and $Stack_{b,t}$. The rest of the construction is straightforward. $State_t$, $Action_a$ and $Action_b$ are functions of $State_{t-1}, Top_{a,t}, Empty_{a,t}, Top_{b,t}, Empty_{b,t}$ and $in_t$. Since all of these are discrete random variables, the conditional probability densities is simply the transition function of the PDA, written as a (0, 1) stochastic matrix. As expected $P(Y = halt_i|State) = 1$ if $State$ is that halt state, and 0 otherwise.

Finally, the priors for the dynamic Bayesian network are simply $P(Stack_{a,1}) = P(Stack_{b,1}) = \delta(0)$, $P(State_1 = q_0) = 1$, where $q_0$ is the initial state.

As described, this construction is somewhat of an abuse of the term 'probabilistic graphical model' - all probability mass is concentrated into a single event for every random variable in the system, for every time step. However, it is easy to see this construction faithfully simulates a two stack machine, as each random variable in the construction corresponds exactly to a component of the simulated automaton.

## 6.5 Exact Inference in Continuous-discrete Bayesian Networks

This construction requires continuous random variables, which raise concerns as to whether the marginal posterior probabilities can be effectively computed. The original junction tree algorithm [46] and cut-set conditioning [59] approaches to belief propagation compute exact marginals for arbitrary DAGs, but require discrete random variables. Lauritzen's algorithm [45] conducts inference in mixed graphical models, but is limited to conditional linear Gaussian (CLG) continuous random

variables. In a CLG model, let $X$ be a continuous node, $\mathbf{A}$ be its discrete parents, and $Y_1, \ldots, Y_k$ be continuous parents. Then

$$p(X|\mathbf{a}, \mathbf{y}) = N(\mathbf{w}_{\mathbf{a},0} + \sum_{i=1}^{k} w_{\mathbf{a},i} y_i; \sigma^2_{\mathbf{a}}) \tag{6.9}$$

Lauritzen's algorithm can only conduct approximate inference, since the true posterior marginals may be some multimodal mix of Gaussians, while the algorithm itself only supports CLG random variables. However, the algorithm is exact in the sense that it computes exact first and second moments for the posterior marginals which is sufficient for the Turing machine simulation.

Laurientz's algorithm does not permit discrete random variables to be children of continuous random variables. Lerner's algorithm [47] extends Lauritzen's algorithm to support softmax conditional probability densities for discrete children of continuous parents. Let A be a discrete node with the possible values $a_1, \ldots, a_m$ and let $Y_1, \ldots, Y_k$ be its parents. Then:

$$P(A = a_i|y_1, \ldots, y_k) = \frac{exp(b^i + \sum_{l=1}^{n} w_l^i y_l)}{\sum_{j=1}^{m} exp(b^j + \sum_{l=1}^{n} w_l^i y_l)} \tag{6.10}$$

Like Lauritzen's algorithm, Lerner's algorithm computes approximate posterior marginals - relying on the observation that the product of a softmax and a Gaussian is approximately Gaussian - but exact first and second moments, up to errors in the numerical integration used to compute the best Gaussian approximation of the product of a Gaussian and a softmax. This calculation is actually simpler in the case where the softmax is replaced with a Heavyside and the Lerner algorithm

can run essentially unmodified with a mixture of Heavyside and softmax conditional probability densities. In the case of Dirac-distributed parents, with Heavyside conditional probability densities, numeric integration is unnecessary and no errors are introduced in computing the first and second moments of the posterior distribution.

Any non-zero variance for the continuous variables will 'leak' probability to other values for the 'stack' random variables in the Turing machine simulation, eventually leading to errors. Lauritzen's original algorithm assumes positive-definite covariance matrices for the continuous random variables, but can be extend to handle degenerate Gaussians [67]. In summary: posterior marginals for the Turing machine simulation can be computed exactly, using a modified version of the Lerner algorithm when restricted to Dirac distributed continuous random variables with Heavside conditional probability densities. If Gaussian random variables and softmax conditional probability densities are also introduced, then the first and second moments of the posterior marginals can be computed 'exactly', up to errors in numerical integration, although this will slowly degrade the quality of the Turing machine simulation in later time steps.

Inference in Bayesian networks is NP-hard [13]. However, assuming that arithmetic operations can be computed in unit time over arbitrary-precision numbers (e.g. the real RAM model), the work necessary at each time step is constant. Thus, dynamic Bayesian networks can simulate Turing-machines with only a constant time overhead in the real RAM model, and slowdown proportional to the time complexity of arbitrary precision arithmetic otherwise.

## 6.6 Aside: comparison to neural networks

This result for dynamic Bayesian networks is analogous to Siegelmann and Sontag's proof that a recurrent neural network can simulate a Turing machine in real time [77]. In fact, neural networks and Bayesian networks turn out to have very similar expressive power:

1. Single perceptron ≈ Gaussian naive Bayes (Logistic regression) [55]
2. Multilayer perceptron ≈ Full Bayesian network (Universal function approximation) [14] [85]
3. Recurrent neural network ≈ Dynamic Bayesian network (Turing complete)

There is an interesting gap in decidability - it takes very little to turn a sub-Turing framework for modeling into a Turing-complete one. In the case of neural networks, a single recurrent layer, with arbitrary-precision rational weights and a saturating linear transfer function is sufficient. With dynamic Bayesian networks, two time-slices, continuous-valued random variables with a combination of linear and step function conditional probability densities is sufficient.

Although such a simple recurrent neural network is theoretically capable of performing arbitrary computations, practical extensions include higher-order connections [64], 'gates' in long short-term memory [36], and even connections to an 'external' Turing machine [28]. These additions enrich the capabilities of standard neural networks and make it easier to train them for complex algorithmic tasks.

An interesting open question is to what degree dynamic Bayesian networks can be similarly extended and how the 'core' dynamic Bayesian network being capable of Turing-complete computation affects the overall performance of such networks.

## 6.7 Consequences for causal modeling

The main consequence of the uncomputability of dynamic Bayesian networks for causal modeling can be summed up as "Nonparametric, cyclic, complete: pick at most two". The vast majority of this dissertation focused on nonparametric models with completeness guarantees in inference. In comparison, linear structural equation modeling is decidedly parametric, but guarantees a unique equilibrium, even in cyclic models.

Unfortunately, the main result in this chapter suggests that preserving completeness for other classes of parametric models is fundamentally difficult: simulating a Turing machine can be done with a combination of discrete and normally distributed random variables, even if the only permitted conditional density function between continuous parents and discrete children is the logistic function. This strongly suggests that semi-parametric classes of cyclic models will often be similarly lacking in completeness guarantees, e.g. restricting the $f_i$s to monotonic functions will be insufficient. Informally, it is too easy to render a system 'accidentally' Turing-complete.

This is not to suggest that cyclic causal modeling and inference is hopeless in practice — many *particular* sets of model assumptions permit computing the equi-

librium distribution of the system. What is unobtainable is the guarantee that the query will either be identifiable or provably unidentifiable from the information at hand. In a sense, this is no worse than most mathematical and scientific research — the solution may be provable with more work, or forever unprovable. Nonparametric, acyclic models are profoundly *unusual* in the strong inference guarantees that can be provided.

# Chapter 7: Conclusions and future work

> The really challenging problems are still ahead.
>
> —Judea Pearl

## 7.1 Summary

The underlying goal of this dissertation was to reduce the ambiguity and difficulty of rigorous causal modeling and inference. To this end, this dissertation explored the development of new abstractions, taxonomies and related analyses based on structural causal models. The causation coefficient, taxonomy of correlation/causation relationships and related analysis clarify how correlation and causation can fail to coincide and provide an argument for the necessity of formal causal analysis. The causal programming abstractions of model, distribution, query and formula unify a large number of different causal inference problems into a single theoretical framework. The implementation of causal programming demonstrates that it is possible to provide a declarative programming language and interactive system for the identification and estimation of interventional queries. Finally, the analysis of the equilibrium distribution of cyclic models suggests that recursive, nonparametric models are, in some sense, maximally powerful.

The 'core' of the work described in this dissertation is the design and implementation of causal programming. The analysis of the causation coefficient supports the necessity of causal programming, and the analysis of cyclic models suggests that further generalization is fundamentally difficult. Causal programming itself shows that it is possible to abstract over problems of causality — making it easy to declare and understand causal assumptions, while automating away the task of conducting inference.

Ideally, a researcher should only need to formally declare what they know and what they wish to know, and be guaranteed to either get the correct answer, or rest assured that that reaching such a conclusion is impossible with the available information. The implementation of causal programming is a first, concrete step towards this goal.

## 7.2 Contributions

The specific contributions of the dissertation are as follows:

- The causation coefficient and related analyses introduce a taxonomy of correlation / causation relationships and a new method for visualizing distributions of causal models. I argue it is insufficient to merely say 'correlation is not causation'; no single epigram will suffice to convey the nature of the possible interactions. The taxonomy outlines how correlation and causation may fail to coincide. Introducing the $\gamma\rho$ plot makes it possible to visualize where in this taxonomy a distribution of models lie. This provides new possible intuition for

understanding why, despite warnings, it is easy to fall into the trap of thinking correlation and causation are the same. For random models, drawn from a distribution of simple causal models, correlation and causation mostly do coincide. However, for models where this is not true, no amount of additional data sampling will suffice to correct the misconception. There is simply no substitute for proper causal analysis.

- The causal programming abstractions group the mathematical objects associated with structural causal modeling into: model, data, query and formula. This permits unifying a large number of (previously separate) problems in causal inference and acts as a guide for formalizing new problems of interest.

- The implementation of causal programming demonstrates that the abstractions are amenable to automated inference. The chief significance is that this demonstrates that it is possible to implement the identification and estimation of interventional queries as a declarative (purely functional) programming language. As an embedded, domain specific language, it is straightforward to extend the language and embed it into a notebook interface for interactive computing. Lisp syntax permits the syntax of the language to closely match the underlying mathematics. Since the implementation is based on a new, purely functional implementation of Shpitser's ID algorithm, inference is complete for identifying causal effect queries from joint observational probability distributions.

- The analysis of the equilibrium distribution of cyclic models is centered around

a proof that the equilibrium distribution of dynamic Bayesian networks is noncomputable, given mild assumptions about the distribution of the random variables in the model. This strongly suggests a fundamental limitation to causal modeling and inference. Methods to analyze *cyclic*, nonparametric models will be necessarily incomplete.

## 7.3 Future work

The current implementation of causal programming is limited in the types of inference that can be performed. It should be relatively straightforward to implement support for counterfactual queries, surrogate experiments and transportability problems — there exists efficient complete algorithms for these problems in the structural causal model literature. In addition, sound, but incomplete support for recovery from selection bias, causal discovery, research design and query generation could be added.

The implementation of causal programming does not directly try to solve the problem of estimation, effectively 'offloading' it to a protocol, to be implemented by user code. This opens up the possibility to combine causal programming with probabilistic programming. Causal programming generates formulas that transform probability distribution to other probability distribution, but exact inference is expensive in the general case. One of probabilistic programming's key insights is that intractable exact inference problems can be solved approximately by sampling. Instead of computing the distribution directly, a large number of samples can be

generated, from which measures of interest (e.g. expected value) can be efficiently calculated. Probabilistic programming is an active area of research, and by combing causal programming with probabilistic programming, advances in one will benefit the other.

A more ambitious goal of automating scientific discovery likely remains far off. Hopefully, causal programming represents one step towards that goal: as a guide to designing languages and software systems that make it easy for researchers to formalize and understand what their assumptions and data and enabling a virtuous cycle between computer inference and human judgement.

# Bibliography


[1] Richard P Bagozzi and Youjae Yi. "Specification, evaluation, and interpretation of structural equation models". In: *Journal of the academy of marketing science* 40.1 (2012), pp. 8–34.

[2] Phil Bagwell. *Ideal hash trees*. Tech. rep. Es Grands Champs, 2001.

[3] Elias Bareinboim. "Generalizability in Causal Inference". PhD thesis. UCLA, 2014.

[4] Elias Bareinboim, Carlos Brito, and Judea Pearl. "Local characterizations of causal Bayesian networks". In: *Graph Structures for Knowledge Representation and Reasoning*. Springer, 2012, pp. 1–17.

[5] Elias Bareinboim and Judea Pearl. "Causal Inference by Surrogate Experiments". In: *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence*. 2012.

[6] Elias Bareinboim and Judea Pearl. "Controlling selection bias in causal inference". In: *Artificial Intelligence and Statistics*. 2012, pp. 100–108.

[7] Elias Bareinboim and Judea Pearl. "Transportability from Multiple Environments with Limited Experiments: Completeness Results". In: *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*. 2014.

[8] Roger Brown and Deborah Fish. "The psychological causality implicit in language". In: *Cognition* 14.3 (1983), pp. 237–273.

[9] William E Byrd. "Relational programming in minikanren: Techniques, applications, and implementations". PhD thesis. Indiana University, 2009.

[10] Nancy Cartwright. "Nature's Capacities and their Measurement". In: *OUP Catalogue* (1994).

[11] Clive R Charig et al. "Comparison of treatment of renal calculi by open surgery, percutaneous nephrolithotomy, and extracorporeal shockwave lithotripsy." In: *Br Med J (Clin Res Ed)* 292.6524 (1986), pp. 879–882.

[12] David Maxwell Chickering and Judea Pearl. "A clinician's tool for analyzing non-compliance". In: *Proceedings of the National Conference on Artificial Intelligence*. 1996, pp. 1269–1276.

[13] Gregory F Cooper. "The computational complexity of probabilistic inference using Bayesian belief networks". In: *Artificial intelligence* 42.2 (1990), pp. 393–405.

[14] George Cybenko. “Approximation by superpositions of a sigmoidal function”. In: *Mathematics of control, signals and systems* 2.4 (1989), pp. 303–314.

[15] A. P. Dawid. “Conditional Independence in Statistical Theory”. In: *Journal of the Royal Statistical Society* (1979).

[16] A. Philip Dawid. “Causal inference without counterfactuals”. In: *Journal of the American Statistical Association* (2000).

[17] Bruno De Finetti. “Foresight: Its logical laws, its subjective sources”. In: *Studies in subjective probability* 1 (1964), pp. 93–158.

[18] Thomas Dean and Keiji Kanazawa. “A Model for Reasoning About Persistence and Causation”. In: *Comput. Intell.* 5.3 (Dec. 1989), pp. 142–150. ISSN: 0824-7935. DOI: 10.1111/j.1467-8640.1989.tb00324.x.

[19] P. Dupont, F. Denis, and Y. Esposito. “Links between probabilistic automata and hidden Markov models: probability distributions, learning models and induction algorithms”. In: *Pattern Recognition* 38.9 (2005). Grammatical Inference, pp. 1349–1371. ISSN: 0031-3203.

[20] Robert F Engle, David F Hendry, and Jean-Francois Richard. “Exogeneity”. In: *Econometrica: Journal of the Econometric Society* (1983), pp. 277–304.

[21] Kit Fine. “Critical Notice”. In: *Mind* 84.355 (1975), pp. 451–458.

[22] Milton Friedman. “The Fed’s Thermostat”. In: *The Wall Street Journal* (2003).

[23] David Galles and Judea Pearl. “An axiomatic characterization of causal counterfactuals”. In: *Foundations of Science* 3.1 (1998), pp. 151–182.

[24] Emden R. Gansner and Stephen C. North. “An open graph visualization system and its applications to software engineering”. In: *Software - Practice and Experience* 30.11 (2000), pp. 1203–1233.

[25] Martin Gardner. *Aha! A Two Volume Collection*. MAA, 2006.

[26] Dan Geiger, Thomas Verma, and Judea Pearl. “Identifying independence in Bayesian networks”. In: *Networks* 20.5 (1990), pp. 507–534.

[27] Allan Gibbard and William L Harper. “Counterfactuals and two kinds of expected utility”. In: *Foundations and applications of decision theory*. Springer, 1978, pp. 125–162.

[28] Alex Graves, Greg Wayne, and Ivo Danihelka. “Neural turing machines”. In: *arXiv preprint arXiv:1410.5401* (2014).

[29] Trygve Haavelmo. “The Statistical Implications of a System of Simultaneous Equations”. In: *Econometrica* 11.1 (1943), pp. 1–12.

[30] Joseph Y Halpern. *Actual causality*. MiT Press, 2016.

[31] Steve Hanks and Drew McDermott. “Nonmonotonic logic and temporal projection”. In: *Artificial intelligence* 33.3 (1987), pp. 379–412.

[32] Megan Head et al. “The Extent and Consequences of P-Hacking in Science”. In: *PLOS biology* (2015).

[33] James J Heckman. "The Scientific Model of Causality". In: *Sociological Methodology* 35.1 (2005), pp. 1–97.

[34] James Heckman and Rodrigo Pinto. "Causal analysis after Haavelmo". In: *Econometric Theory* 31.1 (2015), pp. 115–151.

[35] Rich Hickey. "The Clojure programming language". In: *Dynamic Languages Symposium*. 2008.

[36] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[37] Paul Holland. "Causal Inference, Path Analysis, and Recursive Structural Equation Models". In: *Sociological Methodology* (1988).

[38] Paul Holland. "Statistics and Causal Inference". In: *Journal of the American Statistical Association* (1986).

[39] Yimin Huang and Marco Valtorta. "Identifiability in causal Bayesian networks: A sound and complete algorithm". In: *Proceedings of the national conference on artificial intelligence*. Vol. 21. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. 2006, p. 1149.

[40] David Hume. *An Enquiry Concerning Human Understanding*. 1748.

[41] Edwin T Jaynes. "Information theory and statistical mechanics". In: *Physical review* 106.4 (1957), p. 620.

[42] Edwin T Jaynes. *Probability theory: the logic of science*. Cambridge University Press, 2003.

[43] Andrew W. Keep. "A Nanopass framework for Commercial Compiler Development". PhD thesis. School of Informatics and Computing, Indiana University, 2012.

[44] et al. Kluyver Thomas. "Jupyter Notebooks-a publishing format for reproducible computational workflows". In: *ELPUB*. IOS Press, 2016, pp. 87–90.

[45] Steffen L Lauritzen. "Propagation of probabilities, means, and variances in mixed graphical association models". In: *Journal of the American Statistical Association* 87.420 (1992), pp. 1098–1108.

[46] Steffen L Lauritzen and David J Spiegelhalter. "Local computations with probabilities on graphical structures and their application to expert systems". In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1988), pp. 157–224.

[47] Uri Lerner, Eran Segal, and Daphne Koller. "Exact inference in networks with discrete children of continuous parents". In: *Proceedings of the seventeenth conference on uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc. 2001, pp. 319–328.

[48] David Lewis. "Causation". In: *The Journal of Philosophy* 70.17 (1973), pp. 556–567.

[49] Vikash Kumar Mansinghka. "Natively probabilistic computation". PhD thesis. Massachusetts Institute of Technology, 2009.

[50] Jacob Marschak. *Economic Measurements for Policy and Prediction*. Tech. rep. Cowles Commission, 1953.

[51] Alfred Marshall. *Principles of Economics*. Macmillan, 1890.

[52] John Stuart Mill. *Principles of Political Economy*. John W. Parker, 1848.

[53] Jacob M Montgomery, Brendan Nyhan, and Michelle Torres. "How conditioning on posttreatment variables can ruin your experiment and what to do about it". In: *American Journal of Political Science* 62.3 (2018), pp. 760–775.

[54] Kevin Patrick Murphy. "Dynamic bayesian networks: representation, inference and learning". PhD thesis. University of California, Berkeley, 2002.

[55] Andrew Ng and Michael Jordan. "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes". In: *Advances in neural information processing systems* 14 (2002), p. 841.

[56] Judea Pearl. "Causal diagrams for empirical research". In: *Biometrika* 82.4 (1995), pp. 669–688.

[57] Judea Pearl. *Causality*. Cambridge University Press, 2009.

[58] Judea Pearl. "Comment: Understanding Simpson's Paradox". In: *The American Statistician* 68.1 (2014), pp. 8–13.

[59] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., 1988.

[60] Judea Pearl. *Why there is no statistical test for confounding, why many think there is, and why they are almost right*. Tech. rep. UCLA, 1998.

[61] Judea Pearl and Dana Mackenzie. *The Book of Why: The New Science of Cause and Effect*. Basic Books, 2018.

[62] Judea Pearl and Thomas Verma. "A theory of inferred causation". In: *Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference* (1991), pp. 441–452.

[63] Karl Pearson. "Mathematical Contributions to the Theory of Evolution". In: *Proceedings of the Royal Society of London* (1896).

[64] Fernando J Pineda. "Generalization of back propagation to recurrent and higher order neural networks". In: *Neural information processing systems*. 1988, pp. 602–611.

[65] Graham E Quinn et al. "Myopia and ambient lighting at night". In: *Nature* 399.6732 (1999), pp. 113–114.

[66] Vineet K Raghu et al. "Comparison of strategies for scalable causal discovery of latent variable models from mixed data". In: *International journal of data science and analytics* (2018).

[67] Christopher Raphael. "Bayesian networks with degenerate Gaussian distributions". In: *Methodology and Computing in Applied Probability* 5.2 (2003), pp. 235–263.

[68] Hans Reichenbach. *The Direction of Time*. University of Los Angeles Press, 1956.

[69] Paul R Rosenbaum and Donald B Rubin. "The central role of the propensity score in observational studies for causal effects". In: *Biometrika* 70.1 (1983), pp. 41–55.

[70] Nicholas Rowe. *Milton Friedman's Thermostat*. 2010. URL: `https://www.econlib.org/archives/2015/10/a_theory_of_hou.html`.

[71] Donald B Rubin. "Comment: The design and analysis of gold standard randomized experiments". In: *Journal of the American Statistical Association* 103.484 (2008), pp. 1350–1353.

[72] Donald B Rubin. "Estimating causal effects of treatments in randomized and nonrandomized studies". In: *Journal of educational Psychology* 66.5 (1974), p. 688.

[73] Ilya Shpitser. "Complete Identification Methods for Causal Inference". PhD thesis. University of California Los Angeles, CA 90095-1596, USA, 2008.

[74] Ilya Shpitser and Judea Pearl. "Complete identification methods for the causal hierarchy". In: *Journal of Machine Learning Research* 9.Sep (2008), pp. 1941–1979.

[75] Ilya Shpitser and Judea Pearl. "Identification of conditional interventional distributions". In: *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*. 2006, pp. 437–444.

[76] Ilya Shpitser and Judea Pearl. "Identification of joint interventional distributions in recursive semi-Markovian causal models". In: *21st National Conference on Artificial Intelligence and the 18th Innovative Applications of Artificial Intelligence Conference, AAAI-06/IAAI-06*. 2006, pp. 1219–1226.

[77] Hava T Siegelmann and Eduardo D Sontag. "On the computational power of neural nets". In: *Journal of computer and system sciences* 50.1 (1995), pp. 132–150.

[78] Peter Spirtes and Clark Glymour. "An algorithm for fast recovery of sparse causal graphs". In: *Social science computer review* 9.1 (1991), pp. 62–72.

[79] Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, Prediction, and Search*. 2000.

[80] Santtu Tikka and Juha Karvanen. "Simplifying probabilistic expressions in causal inference". In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 1203–1232.

[81] Santtu Tikka and Juhan Karvanen. "Identifying Causal Effects with the R Package causaleffect". In: *Journal of Statistical Software* (2017).

[82] Edward R Tufte. *The Cognitive Style of PowerPoint: Pitching Out Corrupts Within*. Graphics Press, 2006.

[83] Caroline Uhler et al. "Geometry of the faithfulness assumption in causal inference". In: *The Annals of Statistics* 41.2 (2013), pp. 436–463.

[84] Kevin S Van Horn. "Constructing a logic of plausible inference: a guide to Cox's theorem". In: *International Journal of Approximate Reasoning* 34.1 (2003), pp. 3–24.

[85] Gherardo Varando, Concha Bielza, and Pedro Larrañaga. "Expressive power of binary relevance and chain classifiers based on Bayesian networks for multi-label classification". In: *Probabilistic Graphical Models*. Springer, 2014, pp. 519–534.

[86] Howard Wainer. "Eelworms, bullet holes, and Geraldine Ferraro: Some problems with statistical adjustment and some solutions". In: *Journal of Educational and Behavioral Statistics* 14.2 (1989), pp. 121–140.

[87] Eric Weisstein. *Antichain*. MathWorld. 2005.

[88] Eric Weisstein. *Correlation Coefficient*. MathWorld. 2006.

[89] Sewall Wright. "Correlation and causation". In: *Journal of agricultural research* 20.7 (1921), pp. 557–585.