Causal Imitation Learning with Unobserved Confounders

Junzhe Zhang
Columbia University
junzhez@cs.columbia.edu

Daniel Kumor
Purdue University
dkumor@purdue.edu

Elias Bareinboim
Columbia University
eb@cs.columbia.edu

Abstract

One of the common ways children learn is by mimicking adults. Imitation learning focuses on learning policies with suitable performance from demonstrations generated by an expert, with an unspecified performance measure, and unobserved reward signal. Popular methods for imitation learning start by either directly mimicking the behavior policy of an expert (*behavior cloning*) or by learning a reward function that prioritizes observed expert trajectories (*inverse reinforcement learning*). However, these methods rely on the assumption that covariates used by the expert to determine her/his actions are fully observed. In this paper, we relax this assumption and study imitation learning when sensory inputs of the learner and the expert differ. First, we provide a non-parametric, graphical criterion that is complete (both necessary and sufficient) for determining the feasibility of imitation from the combinations of demonstration data and qualitative assumptions about the underlying environment, represented in the form of a causal model. We then show that when such a criterion does not hold, imitation could still be feasible by exploiting quantitative knowledge of the expert trajectories. Finally, we develop an efficient procedure for learning the imitating policy from experts' trajectories.

1 Introduction

A unifying theme of Artificial Intelligence is to learn a policy from observations in an unknown environment such that a suitable level of performance is achieved [31, Ch. 1.1]. Operationally, a policy is a decision rule that determines an action based on a certain set of covariates; observations are possibly generated by a human demonstrator following a different *behavior policy*. The task of evaluating policies from a combination of observational data and assumptions about the underlying environment has been studied in the literature of causal inference [27] and reinforcement learning [35]. Several criteria, algorithms, and estimation methods have been developed to solve this problem [27, 34, 3, 6, 33, 30, 42]. In many applications, unfortunately, it is not entirely clear which performance measure the demonstrator (possibly subconsciously) is optimizing. That is, the reward signal is not labeled and accessible in the observed expert's trajectories. In such settings, the performance of candidate policies is not uniquely discernible from the observational data due to latent outcomes, even when infinitely many samples are gathered, let alone learning a policy with satisfactory performance.

An alternative approach used to circumvent this issue is to find a policy that mimics a demonstrator's behavior, which leads to the *imitation learning* paradigm [2, 4, 14, 26]; that is, how one should undertake imitation. The expectation (in reality hope) is that if the demonstrations are generated by an expert with near-optimal reward, the performance of the imitator would also be satisfactory. Current methods of imitation learning can be categorized into the *behavior cloning* [43, 29, 21, 22, 20] and the *inverse reinforcement learning* [23, 1, 36, 44]. The former focuses on learning a nominal expert policy that approximates the conditional distribution mapping observed input covariates of the behavior policy to the action domain. The latter attempts to learn a reward function that prioritizes observed behaviors of the expert; reinforcement learning methods are then applied using the learned

Preprint. Under review.

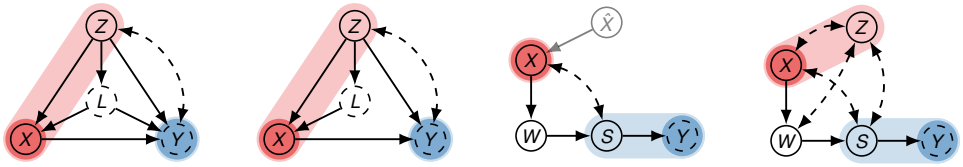


Figure 1: Causal diagrams where X represents an action (shaded red) and Y represents a latent reward (shaded blue). Input covariates of the policy space Π are shaded in light red and minimal imitation surrogates relative to action X and reward Y are shaded in light blue.

reward function to obtain a nominal policy. However, both families of methods rely on the assumption that the input covariates of the behavior policy generating demonstrations are fully observed. When unobserved covariates exist, however, naively imitating the nominal expert policy does not necessarily lead to a satisfactory performance, even when the expert him or herself behaves optimally.

For concreteness, consider a learning scenario depicted in Fig. 2 concerning with trajectories of human-driven cars collected by drones flying over highways [17, 8]. Using such data, we want to learn a policy $\pi(x|z)$ deciding the acceleration (action) X of the demonstrator car based on the velocity and locations of both the demonstrator and front cars, summarized as covariates Z. In reality, the human demonstrator also uses the tail light L of the front car to coordinate his/her actions. The demonstrator performance is evaluated with a latent reward function Y taking X, Z, L as input. However, only observations of X, Z are collected by the drone, summarized as probabilities $P(x, z)$. Fig. 1a describes the graphical representation of this environment. A naive approach would estimate the conditional distribution $P(x|z)$ and use it as policy π. A preliminary analysis reveals that this naive "cloning" approach leads to sub-optimal performance. Consider an instance where variables $X, Y, Z, L, U \in \{0, 1\}$; their values are decided by functions: $L \leftarrow Z \oplus U$, $X \leftarrow Z \oplus \neg L$, $Y \leftarrow X \oplus Z \oplus L$; Z, U are independent variables drawn uniformly over $\{0, 1\}$; $\oplus$ represents the *exclusive-or* operator. The expected reward $\mathbb{E}[Y|\text{do}(\pi)]$ induced by $\pi(x|z) = P(x|z)$ is equal to 0.5, which is quite far from the optimal demonstrator's performance, $\mathbb{E}[Y] = 1$.

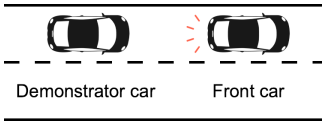


Figure 2: The tail light of the front car is unobserved in highway (aerial) drone data.

This may be surprising since even when one is able to perfectly mimic an optimal demonstrator, the learned policy can still be suboptimal. In this paper, we try to explicate this phenomenon and, more broadly, understand *imitability* through a causal lens[1]. Our task is to learn an imitating policy that achieves the expert's performance from demonstration data in a *structural causal model* [27, Ch. 7], allowing for unobserved confounders affecting both action and outcome variables. Specifically, our contributions are as follows. (1) We introduce a complete graphical criterion for determining the feasibility of imitation from demonstration data and qualitative knowledge about the data-generating process represented as a causal graph. (2) We develop a sufficient algorithm for identifying an imitating policy when such criterion does not hold, by leveraging the quantitative knowledge in the observational distribution. (3) We provide an efficient and practical procedure for finding an imitating policy through explicit parametrization of the causal model. Our results are validated on high-dimensional, synthetic datasets. Due to space constraints, proofs are provided in Appendix A.

1.1 Preliminaries

In this section, we introduce the basic notations and definitions used throughout the paper. We use capital letters to denote variables (X) and small letters for their values (x). Let $\mathscr{D}_X$ represent the domain of X and $\mathscr{P}_X$ the space of probabilistic distribution over $\mathscr{D}_X$. For a set $\boldsymbol{X}$, let $|\boldsymbol{X}|$ denote its dimension. We consistently use the abbreviation $P(x)$ to represent the probabilities $P(X = x)$. Finally, $I_{\{\boldsymbol{Z}=\boldsymbol{z}\}}$ is an indicator function that returns 1 if $\boldsymbol{Z} = \boldsymbol{z}$ holds true; otherwise 0.

[1]Some recent progress in the field of causal imitation has been reported, albeit oblivious to the phenomenon described above and our contributions. Some work considered settings in which the input to the expert policy is fully observed [7], while another assumed that the primary outcome is observed (e.g., Y in Fig. 1a) [8].

We will use calligraphic letters, e.g., $\mathcal{G}$, to represent directed acyclic graphs (DAGs) (e.g., Fig. 1). We denote by $\mathcal{G}_{\overline{\boldsymbol{X}}}$ an edge-induced subgraph obtained from $\mathcal{G}$ by removing arrows coming into nodes in $\boldsymbol{X}$; $\mathcal{G}_{\underline{\boldsymbol{X}}}$ is a subgraph of $\mathcal{G}$ by removing arrows going out of $\boldsymbol{X}$. We will use standard family conventions for graphical relationships such as parents, children, descendants, and ancestors. For example, the set of parents of $\boldsymbol{X}$ in $\mathcal{G}$ is denoted by $pa(\boldsymbol{X})_{\mathcal{G}} = \cup_{X\in\boldsymbol{X}} pa(X)_{\mathcal{G}}$. ch, de and an are similarly defined, We write Pa, Ch, De, An if arguments are included as well, e.g. $De(\boldsymbol{X})_{\mathcal{G}} = de(\boldsymbol{X})_{\mathcal{G}} \cup \boldsymbol{X}$. A path from a node X to a node Y in $\mathcal{G}$ is a sequence of edges which does not include a particular node more than once. Two sets of nodes $\boldsymbol{X}, \boldsymbol{Y}$ are said to be d-separated by a third set $\boldsymbol{Z}$ in a DAG $\mathcal{G}$, denoted by $(\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z})_{\mathcal{G}}$, if every edge path from nodes in one set to nodes in another are "blocked". The criterion of blockage follows [27, Def. 1.2.3].

The basic semantical framework of our analysis rests on *structural causal models* (SCMs) [27, Ch. 7]. An SCM M is a tuple $\langle \boldsymbol{U}, \boldsymbol{V}, \mathcal{F}, P(\boldsymbol{u})\rangle$ where $\boldsymbol{V}$ is a set of endogenous variables and $\boldsymbol{U}$ is a set of exogenous variables. $\mathcal{F}$ is a set of structural functions where $f_V \in \mathcal{F}$ decides values of an endogenous variable $V \in \boldsymbol{V}$ taking as argument a combination of other variables. That is, $V \leftarrow f_V(Pa_V, U_V), Pa_V \subseteq \boldsymbol{V}, U_V \subseteq \boldsymbol{U}$. Values of $\boldsymbol{U}$ are drawn from an exogenous distribution $P(\boldsymbol{u})$. Naturally, each SCM M induces a distribution $P(\boldsymbol{v})$ over endogenous variables $\boldsymbol{V}$. An intervention on a subset $\boldsymbol{X} \subseteq \boldsymbol{V}$, denoted by do$(\boldsymbol{x})$, is an operation where values of $\boldsymbol{X}$ are set to constants $\boldsymbol{x}$, regardless of how they are ordinarily determined through functions $\{f_X : \forall X \in \boldsymbol{X}\}$. For an SCM M, let $M_{\boldsymbol{x}}$ be a submodel of M induced by intervention do$(\boldsymbol{x})$. For a set $\boldsymbol{S} \subseteq \boldsymbol{V}$, the interventional distribution $P(\boldsymbol{s}|\text{do}(\boldsymbol{x}))$ induced by do$(\boldsymbol{x})$ is defined as the distribution over $\boldsymbol{S}$ in the submodel $M_{\boldsymbol{x}}$, i.e., $P(\boldsymbol{s}|\text{do}(\boldsymbol{x}); M) \triangleq P(\boldsymbol{s}; M_{\boldsymbol{x}})$. We leave M implicit when it is obvious from the context. For a detailed survey on SCMs, we refer readers to [27, Ch. 7].

2 Imitation Learning in Structural Causal Models

In this section, we formalize and study the imitation learning problem in causal language. We first define a special type of SCM that explicitly allows one to model the unobserved nature of some endogenous variables, which is called the partially observable structural causal model (POSCM).[2]

Definition 1 (Partially Observable SCM)**.** A POSCM is a tuple $\langle M, \boldsymbol{O}, \boldsymbol{L}\rangle$, where M is a SCM $\langle \boldsymbol{U}, \boldsymbol{V}, \mathcal{F}, P(\boldsymbol{u})\rangle$ and $\langle \boldsymbol{O}, \boldsymbol{L}\rangle$ is a pair of subsets forming a partition over $\boldsymbol{V}$ (i.e., $\boldsymbol{V} = \boldsymbol{O} \cup \boldsymbol{L}$ and $\boldsymbol{O} \cap \boldsymbol{L} = \emptyset$); $\boldsymbol{O}$ and $\boldsymbol{L}$ are called observed and latent endogenous variables, respectively.

Each POSCM M induces a probability distribution over the observed variables $\boldsymbol{O}$, $P(\boldsymbol{o})$, usually called the *observational* distribution. M is associated with a *causal diagram* $\mathcal{G}$ (e.g., see Fig. 1) where solid nodes represent observed variables $\boldsymbol{O}$, dashed nodes represent latent variables $\boldsymbol{L}$, and arrows represent the arguments Pa_V of each functional relationship f_V. Exogenous variables $\boldsymbol{U}$ are not explicitly shown; a bi-directed arrow between nodes V_i and V_j indicates the presence of an unobserved confounder (UC) affecting both V_i and V_j, i.e., $U_{V_i} \cap U_{V_j} \neq \emptyset$.

Consider an POSCM $\langle M, \boldsymbol{O}, \boldsymbol{L}\rangle$ with $M = \langle \boldsymbol{U}, \boldsymbol{V}, \mathcal{F}, P(\boldsymbol{u})\rangle$. Our goal is to learn an efficient policy to decide the value of an action variable $X \in \boldsymbol{O}$. The performance of the policy is evaluated using the expected value of a reward variable Y. Throughout this paper, we assume that reward Y is latent and X affects Y (i.e., $Y \in \boldsymbol{L} \cap De(X)_{\mathcal{G}}$). A *policy* π is a function mapping from values of covariates $Pa^* \subseteq \boldsymbol{O} \setminus De(X)_{\mathcal{G}_{\overline{X}}}$ to a probability distribution over X, which we denote by $\pi(x|pa^*)$. An intervention following a policy π, denoted by do(π), is an operation that draws values of X independently following π, regardless of its original (natural) function f_X. Let M_π denote the manipulated SCM of M induced by do(π). Similar to atomic settings, the interventional distribution $P(\boldsymbol{v}|\text{do}(\pi))$ is defined as the distribution over $\boldsymbol{V}$ in the manipulated model M_π, given by,

$$P(\boldsymbol{v}|\text{do}(\pi)) = \sum_{\boldsymbol{u}} P(\boldsymbol{u}) \prod_{V\in\boldsymbol{V}\setminus\{X\}} P(v|pa_V, u_V)\pi(x|pa^*). \tag{1}$$

The expected reward of a policy π is thus given by the causal effect $\mathbb{E}[Y|\text{do}(\pi)]$. The collection of all possible policies π defines a *policy space*, denoted by $\Pi = \{\pi : \mathscr{D}_{Pa^*} \mapsto \mathscr{P}_X\}$ (if $Pa^* = \emptyset$, $\Pi = \{\pi : \mathscr{P}_X\}$). For convenience, we define function $Pa(\Pi) = Pa^*$. A policy space Π' is a subspace of Π if $Pa(\Pi') \subseteq Pa(\Pi)$. We will consistently highlight action X in dark red, reward Y in

[2] This definition will facilitate the more explicitly articulation of which endogenous variables are available to the demonstrator and corresponding policy at each point in time.

dark blue and covariates $Pa(\Pi)$ in light red. For instance, in Fig. 1a, the policy space over action X is given by $\Pi = \{\pi : \mathscr{D}_Z \mapsto \mathscr{P}_X\}$; Y represents the reward; $\Pi' = \{\pi : \mathscr{P}_X\}$ is a subspace of Π.

Our goal is to learn an efficient policy $\pi \in \Pi$ that achieves satisfactory performance, e.g., larger than a certain threshold $\mathbb{E}[Y|\text{do}(\pi)] \geq \tau$, without knowledge of underlying system dynamics, i.e., the actual, true POSCM M. A possible approach is to identify the expected reward $\mathbb{E}[Y|\text{do}(\pi)]$ for each policy $\pi \in \Pi$ from the combinations of the observed data $P(\boldsymbol{o})$ and the causal diagram $\mathcal{G}$. Optimization procedures are applicable to find a satisfactory policy π. Let $\mathscr{M}(\mathcal{G})$ denote a hypothesis class of POSCMs that are compatible with a causal diagram $\mathcal{G}$. We define next the non-parametric notion of identifiability in the context of POSCMs and conditional policies, adapted from [27, Def. 3.2.4].

Definition 2 (Identifiability)**.** Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{V}$. $P(\boldsymbol{y}|\text{do}(\pi))$ is said to be identifiable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if $P(\boldsymbol{y}|\text{do}(\pi); M)$ is uniquely computable from $P(\boldsymbol{o}; M)$ and π for any POSCM $M \in \mathscr{M}(\mathcal{G})$ and any $\pi \in \Pi$.

In imitation learning settings, one of the main challenges is that reward Y is not specified and remains latent, which precludes the identifiability of the performance measure $\mathbb{E}[Y|\text{do}(\pi)]$, as shown next.

Corollary 1. *Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{V}$. If not all variables in $\boldsymbol{Y}$ are observed (i.e., $\boldsymbol{Y} \cap \boldsymbol{L} \neq \emptyset$), $P(\boldsymbol{y}|do(\pi))$ is not identifiable.*

In words, Corol. 1 shows that when the reward Y is latent, it is infeasible to uniquely determine values of $\mathbb{E}[Y|\text{do}(\pi)]$ from $P(\boldsymbol{o})$, which suggests the need to explore learning through other modalities.

2.1 Causal Imitation Learning

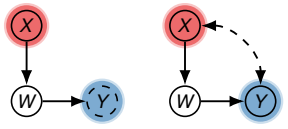
To circumvent issues of non-identifiability, a common solution is to assume that the observed trajectories are generated by an "expert" demonstrator with satisfactory performance $\mathbb{E}[Y]$, e.g., no less than a certain threshold ($\mathbb{E}[Y] \geq \tau$). If we could find a policy π that perfectly imitates the expert, $\mathbb{E}[Y|\text{do}(\pi)] = \mathbb{E}[Y]$, the performance of the learner is also guaranteed to be satisfactory. Formally,

Definition 3 (Imitability)**.** Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{V}$. $P(\boldsymbol{y})$ is said to be imitable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if there exists a policy $\pi \in \Pi$ uniquely computable from $P(\boldsymbol{o}; M)$ such that $P(\boldsymbol{y}|\text{do}(\pi); M) = P(\boldsymbol{y}; M)$ for any POSCM $M \in \mathscr{M}(\mathcal{G})$.

Our task is to determine the imitability of the expert performance. More specifically, we want to learn an *imitating policy* $\pi \in \Pi$ from $P(\boldsymbol{o})$ that mimics the expert reward, $P(y|\text{do}(\pi)) = P(y)$[3], in any POSCM M associated with the causal diagram $\mathcal{G}$. Consider Fig. 3a as an example. $P(y)$ is imitable with policy $\pi(x) = P(x)$ since by Eq. (1) and marginalization, $P(y|\text{do}(\pi)) = \sum_{x,w} P(y|w)P(w|x)\pi(x) = \sum_{x,w} P(y|w)P(w|x)P(x) = P(y)$. In practice, unfortunately, the expert's performance cannot always be imitated. To understand this setting, we first write, more explicitly, the conditions under which this is not the case:

Lemma 1. *Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{V}$. $P(\boldsymbol{y})$ is not imitable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if there exists two POSCMs $M_1, M_2 \in \mathscr{M}(\mathcal{G})$ satisfying $P(\boldsymbol{o}; M_1) = P(\boldsymbol{o}; M_2)$ while there exists no policy $\pi \in \Pi$ such that for $i = 1, 2$, $P(\boldsymbol{y}|do(\pi); M_i) = P(\boldsymbol{y}; M_i)$.*

It follows as a corollary that $P(\boldsymbol{y})$ is not imitable if there exists a POSCM M compatible with $\mathcal{G}$ such that no policy $\pi \in \Pi$ could ensure $P(\boldsymbol{y}|\text{do}(\pi); M) = P(\boldsymbol{y}; M)$. For instance, consider the causal diagram $\mathcal{G}$ and policy space Π in Fig. 3b. Here, the expert's reward $P(y)$ is not imitable: consider a POSCM with functions $X \leftarrow U, W \leftarrow X, Y \leftarrow U \oplus \neg W$; values U are drawn uniformly over $\{0, 1\}$. In this model, $P(Y = 1|\text{do}(\pi)) = 0.5$ for any policy π, which is far from the optimal expert reward, $P(Y = 1) = 1$.



(a) (b)

Figure 3: Imitability v. Identifiability.

An interesting observation from the above example of Fig. 3b is that the effect $P(y|\text{do}(\pi))$ could be identifiable, following the front-door criterion in [27, Thm. 3.3.4], but no policy imitates the corresponding $P(y)$. However, in some settings, the expert's reward $P(y)$ is imitable but the imitator's reward $P(y|\text{do}(\pi))$ cannot be uniquely determined. To witness, consider

[3]The imitation is trivial if $Y \notin De(X)_{\mathcal{G}}$: by Rule 3 of [27, Thm. 3.4.1] (or [6, Thm. 1]), $P(y|\text{do}(\pi)) = P(y)$ for any policy π. This paper aims to find a *specific* π satisfying $P(y|\text{do}(\pi)) = P(y)$ even when $Y \in De(X)_{\mathcal{G}}$.

again the example in Fig. 3a. The imitability of $P(y)$ has been previously shown; while $P(y|\text{do}(\pi))$ is not identifiable due to latent reward Y (Corol. 1).

In general, the problem of imitability is orthogonal to identifiability, and requires separate consideration. Since imitability does not always hold, we introduce a useful graphical criterion for determining whether imitating the expert's performance is feasible, and if so, how.

Theorem 1 (Imitation by Direct Parents)**.** *Given a causal diagram $\mathcal{G}$ and a policy space Π, $P(y)$ is imitable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if $pa(X)_{\mathcal{G}} \subseteq Pa(\Pi)$ and there is no bi-directed arrow pointing to X in $\mathcal{G}$. Moreoever, the imitating policy $\pi \in \Pi$ is given by $\pi(x|pa(\Pi)) = P(x|pa(X)_{\mathcal{G}})$.*

In words, Thm. 1 says that if the expert and learner share the same policy space, then the policy is always imitable. In fact, this result can be seen as a causal justification for when the method of "behavior cloning", widely used in practice, is valid, leading to proper imitation. When the original behavior policy f_X is contained in the policy space Π, the leaner could imitate the expert's reward $P(y)$ by learning a policy $\pi \in \Pi$ that matches the distribution $P(x|pa(\Pi))$ [43, 29]. We now consider the more challenging setting when policy spaces of the expert and learner disagree (i.e., $f_X \notin \Pi$). We will leverage a graphical condition adopted from the celebrated *backdoor criterion* [27, Def. 3.3.1].

Definition 4 (Imitation Backdoor)**.** Given a causal diagram $\mathcal{G}$ and a policy space Π, a set $\boldsymbol{Z}$ is said to satisfy the *imitation backdoor* criterion (i-backdoor) w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if and only if $\boldsymbol{Z} \subseteq Pa(\Pi)$ and $(Y \perp\!\!\!\perp X|\boldsymbol{Z})_{\mathcal{G}_{\underline{X}}}$, which is called the *i-backdoor admissible set*.

For concreteness, consider again the highway driving example in Fig. 1a. There exists no i-backdoor admissible set due to the path $X \leftarrow L \rightarrow Y$. Now consider a modified graph in Fig. 1b where arrow $L \rightarrow Y$ is removed. $\{Z\}$ is i-backdoor admissible since $Z \in Pa(\Pi)$ and $(Y \perp\!\!\!\perp X|Z)_{\mathcal{G}_{\underline{X}}}$. Leveraging the imitation backdoor condition, our next theorem provides a full characterization for when imitating expert's performance is achievable, despite the fact that the reward Y is latent.

Theorem 2 (Imitation by Backdoor)**.** *Given a causal diagram $\mathcal{G}$ and a policy space Π, $P(y)$ is imitable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if and only if there exists an i-backdoor admissible set $\boldsymbol{Z}$ w.r.t. $\langle \mathcal{G}, \Pi \rangle$. Moreover, the imitating policy $\pi \in \Pi$ is given by $\pi(x|pa(\Pi)) = P(x|\boldsymbol{z})$.*

That is, one can learn an imitating policy from a policy space $\Pi' = \{\pi : \mathscr{D}_{\boldsymbol{Z}} \mapsto \mathscr{P}_X\}$ that mimics the conditional probabilities $P(x|\boldsymbol{z})$ if and only if $\boldsymbol{Z}$ is i-backdoor admissible. If that is the case, such a policy can be learned from data through standard density estimation methods. For instance, Thm. 2 ascertains that $P(y)$ in Fig. 1a is indeed non-imitable. On the other hand, $P(y)$ in Fig. 1b is imitable, guaranteed by the i-backdoor admissible set $\{Z\}$; the imitating policy is given by $\pi(x|z) = P(x|z)$.

3 Causal Imitation Learning with Data Dependency

One may surmise that the imitation boundary established by Thm. 2 suggests that when there exists no i-backdoor admissible set, it is infeasible to imitate the expert performance from observed trajectories of demonstrations. In this section, we will show that this is not the case. Consider again the definition of imitability. While it assumes access to the observational distribution $P(\boldsymbol{o})$, Def. 3 is defined with respect to a family of candidate models $\mathscr{M}(\mathcal{G})$ characterized with only the qualitative knowledge represented in a causal diagram $\mathcal{G}$. Let $\mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$ denote a subset of POSCMs in the family $\mathscr{M}(\mathcal{G})$ that induce the observational distribution $P(\boldsymbol{o})$, i.e., $\mathscr{M}(\mathcal{G}, P(\boldsymbol{o})) = \{\forall M \in \mathscr{M}(\mathcal{G}) : P(\boldsymbol{o}; M) = P(\boldsymbol{o})\}$. We introduce a refined notion of imitability that it will explore the quantitative knowledge of observed trajectories $P(\boldsymbol{o})$ (to be exemplified). Formally,

Definition 5 (Data-dependent Imitability)**.** Given a causal diagram $\mathcal{G}$, a policy space Π, and an observational distribution $P(\boldsymbol{o})$, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{V}$. $P(\boldsymbol{y})$ is said to be *practically imitable* (for short, p-imitable) w.r.t. $\langle \mathcal{G}, \Pi, P(\boldsymbol{o}) \rangle$ if there exists a policy $\pi \in \Pi$ uniquely computable from $P(\boldsymbol{o})$ such that $P(\boldsymbol{y}|\text{do}(\pi); M) = P(\boldsymbol{y})$ for any POSCM $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$.

Instead of the entire hypothesis class $\mathscr{M}(\mathcal{G})$, Def. 5 seeks to find an imitating policy only for the subset of POSCMs $\mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$ compatible with the observed data $P(\boldsymbol{o})$. This implies that given a causal diagram $\mathcal{G}$ and a policy space Π, any imitable $P(y)$ is also p-imitable, but not vice versa. In other words, for a non-imitable $P(y)$ w.r.t. $\langle \mathcal{G}, \Pi \rangle$, it could still be p-imitable after analyzing the distribution $P(\boldsymbol{o})$. For concreteness, consider again $P(y)$ in Fig. 3b which is not imitable due to the bi-directed arrow $X \leftrightarrow Y$. However, new imitation opportunities arise when observational

probabilities $P(x, w, y)$ are provided. Suppose the underlying POSCM is given by: $X \leftarrow U_X \oplus U_Y$, $W \leftarrow X \oplus U_W$, $Y \leftarrow W \oplus U_Y$ where U_X, U_Y, U_W are independent binary variables drawn from $P(U_X = 1) = P(U_Y = 1) = P(U_W = 0) = 0.9$. Here, the causal effect $P(y|\text{do}(x))$ is identifiable from $P(x, w, y)$ following the front-door formula $P(y|\text{do}(x)) = \sum_w P(w|x) \sum_{x'} P(y|w, x')P(x')$ [27, Thm. 3.3.4]. We thus have $P(Y = 1|\text{do}(X = 0)) = 0.82$ which coincides with $P(Y = 1) = 0.82$, i.e., $P(y)$ is p-imitable with atomic intervention $\text{do}(X = 0)$. In the most practical settings, the expert reward $P(y)$ rarely equates to $P(y|\text{do}(x))$; stochastic policies $\pi(x)$ are then applicable to imitate $P(y)$ by re-weighting $P(y|\text{do}(x))$ induced by the corresponding atomic interventions.[4]

To tackle this situation in a general way, we proceed by defining a set of observed variables that serve as a surrogate of the unobserved Y with respect to interventions on X. Formally,

Definition 6 (Imitation Surrogate)**.** Given a causal diagram $\mathcal{G}$, a policy space Π, let $\boldsymbol{S}$ be an arbitrary subset of $\boldsymbol{O}$. $\boldsymbol{S}$ is an *imitation surrogate* (i-surrogate) w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if $(Y \perp\!\!\!\perp \hat{X}|\boldsymbol{S})_{\mathcal{G} \cup \Pi}$ where $G \cup \Pi$ is a supergraph of $\mathcal{G}$ by adding arrows from $Pa(\Pi)$ to X; $\hat{X}$ is a new parent to X.

An i-surrogate $\boldsymbol{S}$ is said to be minimal if there exists no subset $\boldsymbol{S}' \subset \boldsymbol{S}$ such that $\boldsymbol{S}'$ is also an i-surrogate w.r.t. $\langle \mathcal{G}, \Pi \rangle$. Consider as an example Fig. 1c where the supergraph $\mathcal{G} \cup \Pi$ coincides with the causal diagram $\mathcal{G}$. By Def. 6, both $\{W, S\}$ and $\{S\}$ are valid i-surrogate relative to $\langle X, Y \rangle$ with $\{S\}$ being the minimal one. By conditioning on S, the decomposition of Eq. (1) implies $P(y|\text{do}(\pi)) = \sum_{s,w,u} P(y|s)P(s|w, u)P(w|x)\pi(x)P(u) = \sum_s P(y|s)P(s|\text{do}(\pi))$. That is, the surrogate S mediates all influence of inventions on action X to reward Y. It is thus sufficient to find an imitating policy π such that $P(s|\text{do}(\pi)) = P(s)$ for any POSCM M associated with Fig. 1c. The resultant policy is guaranteed to imitate the expert's reward $P(y)$.

When an i-surrogate $\boldsymbol{S}$ is found and $P(\boldsymbol{s}|\text{do}(\pi))$ is identifiable, one could compute $P(\boldsymbol{s}|\text{do}(\pi))$ for each policy π from the observational distribution $P(\boldsymbol{o})$ and check if it matches $P(\boldsymbol{s})$. In many settings, however, $P(\boldsymbol{s}|\text{do}(\pi))$ is not identifiable w.r.t. $\langle \mathcal{G}, \Pi \rangle$. For example, in Fig. 1d, S is an i-surrogate w.r.t. $\langle \mathcal{G}, \Pi \rangle$, but $P(s|\text{do}(\pi))$ is not identifiable due to collider Z (π uses non-descendants as input by default). Fortunately, identifying $P(\boldsymbol{s}|\text{do}(\pi))$ may still be feasible in some subspaces of Π:

Definition 7 (Identifiable Subspace)**.** Given a causal diagram $\mathcal{G}$, a policy space Π, and a subset $\boldsymbol{S} \subseteq \boldsymbol{O}$, let Π' be a policy subspace of Π. Π' is said to be an *identifiable subspace* (i-subspace) w.r.t. $\langle \mathcal{G}, \Pi, \boldsymbol{S} \rangle$ if $P(\boldsymbol{s}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi' \rangle$.

Consider a policy subspace $\Pi' = \{\pi : \mathscr{P}_X\}$ in Fig. 1d (i.e. π that does not exploit Z). $P(s|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi' \rangle$ following the front-door adjustment on W [27, Thm. 3.3.4]. We could then evaluate the effects $P(s|\text{do}(\pi))$ for each policy $\pi \in \Pi'$ from the observed $P(x, w, s, z)$ and obtain the imitating policy (if exists). When X, W, S are categorical, the imitating policy is a solution of linear equations, for any s, $\sum_x \pi(x) \sum_w P(w|x) \sum_{x'} P(s|w, x')P(x') = P(s)$. In other words, $\{S\}$ and Π' forms an instrument that allows one to solve the imitation learning problem in Fig. 1d.

Definition 8 (Imitation Instrument)**.** Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{S}$ be a subset of $\boldsymbol{O}$ and Π' be a subspace of Π. $\langle \boldsymbol{S}, \Pi' \rangle$ is said to be an *imitation instrument* (i-instrument) if $\boldsymbol{S}$ is an i-surrogate w.r.t. $\langle \mathcal{G}, \Pi' \rangle$ and Π' is an i-subspace w.r.t. $\langle \mathcal{G}, \Pi, \boldsymbol{S} \rangle$.

The presence of an imitation instrument $\langle \boldsymbol{S}, \Pi' \rangle$ reduces the imitation learning on a latent reward Y to a p-imitability problem over observed variables $\boldsymbol{S}$ using policies in a subspace Π':

Lemma 2. *Given a causal diagram $\mathcal{G}$, a policy space Π, and an observational distribution $P(\boldsymbol{o})$, let $\langle \mathcal{G}, \Pi' \rangle$ be an i-instrument w.r.t. $\langle \mathcal{G}, \Pi \rangle$. If for an arbitrary $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$, there exists a policy $\pi \in \Pi'$ such that $P(\boldsymbol{s}|do(\pi); M) = P(\boldsymbol{s})$, $P(y)$ is p-imitable w.r.t. $\langle \mathcal{G}, \Pi, P(\boldsymbol{o}) \rangle$. Moreover, π is an imitating policy for $P(y)$ w.r.t. $\langle \mathcal{G}, \Pi, P(\boldsymbol{o}) \rangle$.*

When an i-instrument $\langle \boldsymbol{S}, \Pi' \rangle$ is obtained, evaluating the causal effect $P(\boldsymbol{s}|\text{do}(\pi))$ from observed trajectories $P(\boldsymbol{o})$ could still be computational challenging if some observed variables in $\boldsymbol{O}$ are high-dimensional (e.g., W in Fig. 1d). Lem. 2 suggests a practical approach to address this issue. If $P(\boldsymbol{s}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi' \rangle$, by Def. 2, it remains invariant over the family $\mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$. Thus, given an i-instrument, we can compute $P(\boldsymbol{s}|\text{do}(\pi); \hat{M})$ in an arbitrary $\hat{M} \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$; such an evaluation will always coincide with the actual, true causal effect $P(\boldsymbol{s}|\text{do}(\pi))$. This allows one to obtain an imitating policy through direct parametrization of POSCMs [19]. Let $\mathscr{M}_\theta(\mathcal{G})$ be a family

[4] Consider a variation of the model where $P(U_W = 1) = 0.7$. $P(y)$ is p-imitable with $\pi(X = 0) = 0.75$.

of POSCMs in $\mathscr{M}(\mathcal{G})$ such that it is relatively simple to compute (sample from) $P(\boldsymbol{s}|\text{do}(\pi); M)$ for every $M \in \mathscr{M}_\theta(\mathcal{G})$. We obtain an POSCM $\hat{M} \in \mathscr{M}_\theta(\mathcal{G})$ such that $P(\boldsymbol{o}; \hat{M}) = P(\boldsymbol{o})$; an imitating policy π is computed in $\hat{M}$. Lem. 2 guarantees that such a policy π imitates expert's reward $P(y)$.

3.1 Confounding Robust Imitation

Our task in this section is to introduce a general algorithm that finds i-instruments, and learns a p-imitating policy given $P(\boldsymbol{o})$. A naive approach is to enumerate all pairs of subset $\boldsymbol{S}$ and subspace Π' and check whether they form an i-instrument; if so, we then compute an imitating policy for $P(\boldsymbol{s})$ w.r.t. $\langle \mathcal{G}, \Pi', P(\boldsymbol{o}) \rangle$. However, the challenge is that the number of all possible subspaces Π' (or subsets $\boldsymbol{S}$) could be exponentially large. Fortunately, such a search space could be much restricted. Let $\mathcal{G} \cup \{Y\}$ denote a causal diagram obtained from $\mathcal{G}$ by making reward Y observed. The following proposition suggests that it suffices to consider only identifiable subspaces w.r.t. $\langle \mathcal{G} \cup \{Y\}, \Pi, Y \rangle$.

Lemma 3. *Given a causal diagram $\mathcal{G}$, a policy space Π, let a subspace $\Pi' \subseteq \Pi$. If there exists $\boldsymbol{S} \subseteq \boldsymbol{O}$ such that $\langle \boldsymbol{S}, \Pi' \rangle$ is an i-instrument w.r.t. $\langle \mathcal{G}, \Pi \rangle$, Π' is an i-subspace w.r.t. $\langle \mathcal{G} \cup \{Y\}, \Pi, Y \rangle$.*

Our algorithm IMITATE is described in Alg. 1. We assume access to an IDENTIFY oracle [39, 32, 6] that takes as input a causal diagram $\mathcal{G}$, a policy space Π and a set of observed variables $\boldsymbol{S}$. If $P(\boldsymbol{s}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi \rangle$, IDENTIFY returns "YES"; otherwise, it returns "NO". For details about the IDENTIFY oracle, we refer readers to Appendix B. More specifically, IMITATE takes as input a causal diagram $\mathcal{G}$, a policy space Π and an observational distribution $P(\boldsymbol{o})$. At Step 2, IMITATE applies a subroutine LISTIDSPACE to list identifiable subspaces Π' w.r.t. $\langle \mathcal{G} \cup \{Y\}, \Pi, Y \rangle$, following the observation made in Lem. 3. The implementation details of LISTIDSPACE are provided in Appendix C. When an identifiable subspace Π' is found, IMITATE tries to obtain an i-surrogate $\boldsymbol{S}$ w.r.t the diagram $\mathcal{G}$ and subspace Π'. While there could exist multiple such i-surrogates, the following proposition shows that it is sufficient to consider only minimal ones.

Algorithm 1: IMITATE

1: **Input:** $\mathcal{G}, \Pi, P(\boldsymbol{o})$.
2: **while** LISTIDSPACE$(\mathcal{G} \cup \{Y\}, \Pi, Y)$ outputs a policy subspace Π' **do**
3: **while** LISTMINSEP$(\mathcal{G} \cup \Pi', \hat{X}, Y, \{\}, \boldsymbol{O})$ outputs a surrogate set $\boldsymbol{S}$ **do**
4: **if** IDENTIFY$(\mathcal{G}, \Pi', \boldsymbol{S}) =$ YES **then**
5: Solve for a policy $\pi \in \Pi'$ such that
$$P(\boldsymbol{s}|\text{do}(\pi); M) = P(\boldsymbol{s}; M)$$
for any POSCM $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$.
6: Return π if it exists; continue otherwise.
7: **end if**
8: **end while**
9: **end while** Return FAIL.

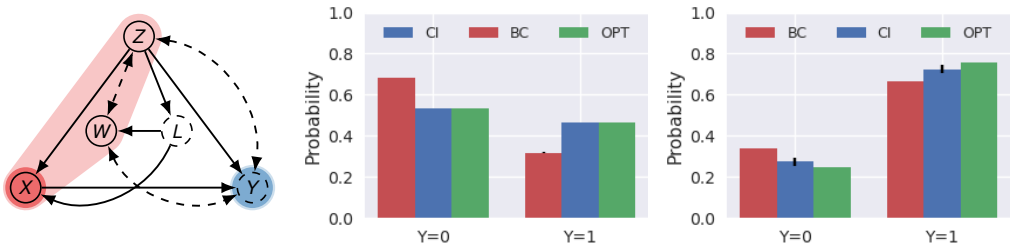
Lemma 4. *Given a causal diagram $\mathcal{G}$, a policy space Π, an observational distribution $P(\boldsymbol{o})$ and a subset $\boldsymbol{S} \subseteq \boldsymbol{O}$. $P(\boldsymbol{s})$ is p-imitable only if for any $\boldsymbol{S}' \subseteq \boldsymbol{S}$, $P(\boldsymbol{s}')$ is p-imitable w.r.t. $\langle \mathcal{G}, \Pi, P(\boldsymbol{o}) \rangle$.*

We apply a subroutine LISTMINSEP in [41] to enumerate minimal i-surrogates in $\boldsymbol{O}$ that d-separate $\hat{X}$ and Y in the supergraph $\mathcal{G} \cup \Pi'$. When a minimal i-surrogate $\boldsymbol{S}$ is found, IMITATE uses IDENTIFY oracle to validate if $P(\boldsymbol{s}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi' \rangle$, i.e., $\langle \boldsymbol{S}, \Pi' \rangle$ form an i-instrument. Consider Fig. 1d as an example. While $P(y|\text{do}(\pi))$ is not identifiable for every policy in Π had Y been observed, Π contains an i-subspace $\{\pi : \mathscr{P}_X\}$ w.r.t. $\langle \mathcal{G} \cup \{Y\}, \Pi, Y \rangle$, which is associated with a minimal i-surrogate $\{S\}$. Applying IDENTIFY confirms that $\langle \{S\}, \{\pi : \mathscr{P}_X\} \rangle$ is an i-instrument.

At Step 5, IMITATE solves for a policy π in the subspace Π' that imitates $P(\boldsymbol{s})$ for all instance in the hypothesis class $\mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$. If such a policy exists, IMITATE returns π; otherwise, the algorithm continues. Since $\langle \boldsymbol{S}, \Pi' \rangle$ is an i-instrument, Lem. 2 implies that the learned policy π, if it exists, is ensured to imitate the expert reward $P(y)$ for any POSCM $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$.

Theorem 3. *Given a causal diagram $\mathcal{G}$, a policy space Π, and an observational distribution $P(\boldsymbol{o})$, if* IMITATE *returns a policy $\pi \in \Pi$, $P(y)$ is p-imitable w.r.t. $\langle \mathcal{G}, \Pi, P(\boldsymbol{o}) \rangle$.*

We solve for the imitating policy at Step 5 using the direct parametrization of POSCMs introduced previously, following the discussion of Lem. 2. In practical experiments, we consider a family of POSCMs $\mathscr{M}_\theta(\mathcal{G})$ where functions of each variable in $\boldsymbol{O}$ are parametrized with a family of neural networks, similar to [19]. We then obtain a model $\hat{M} \in \mathscr{M}_\theta(\mathcal{G})$ such that $P(\boldsymbol{o}; \hat{M}) = P(\boldsymbol{o})$ using *Generative Adversarial Networks* (GANs) [9, 25]. The imitating policy is trained through explicit intervention in the learned $\hat{M}$, and optimized by using a GAN to make policy π imitate $P(\boldsymbol{s})$.



(a) Highway Driving (b) Highway Driving (c) MNIST Digits

Figure 4: (a) Causal diagram for highway driving example where a left-side car exists; (b,c) $P(y|\text{do}(\pi))$ induced by the causal imitation method (*ci*) and the naive behavior cloning (*bc*) compared with the actual distribution $P(y)$ over the expert's reward (*opt*).

4 Experiments

We demonstrate our algorithms on several synthetic datasets, including `highD` [17] consisting of natural trajectories of human driven vehicles, and on `MNIST` digits. In all experiments, we test our causal imitation method (*ci*): we apply Thm. 2 when there exists an i-backdoor admissible set; otherwise, Alg. 1 is used to leverage the observational distribution. As a baseline, we also include the naive behavior cloning (*bc*) that mimics the observed conditional distribution $P(x|pa(\Pi))$ and the actual reward distribution generated by the expert (*opt*). We found that our algorithms consistently imitate distributions over the expert's reward in imitable (p-imitable) cases; and p-imitable instances commonly exist. We refer readers to Appendix D for more experiments, details, and analysis.

Highway Driving We consider a modified example of the drone recordings of human-driven cars in Sec. 1 where the driver's braking action W of the left-side car is also observed. Fig. 4a shows the causal diagram of this environment; Z represent the velocity of the front-car; action X represents the velocity of the driving car; W and the reward signal Y are both affected by an unobserved confounder U, representing the weather condition. In Fig. 4a, $\{Z\}$ is i-backdoor admissible while $\{Z, W\}$ is not due to active path $X \leftarrow L \rightarrow W \leftrightarrow Y$. We obtain policies for the causal and naive imitators training two separate GANs. Distributions $P(y|\text{do}(\pi))$ induced by all algorithms are reported in Fig. 4b. We also measure the L1 distance between $P(y|\text{do}(\pi))$ and the expert's reward $P(y)$. We find that the causal approach (*ci*), using input set $\{Z\}$, successfully imitates $P(y)$ (L1 $= 0.0018$). As expected, the naive approach (*bc*) utilizing all covariates $\{Z, W\}$ is unable to imitate the expert (L1 = 0.2937).

MNIST Digits We consider an instance of Fig. 1c where X, S, Y are binary variables; binary values of W are replaced with corresponding images of `MNIST` digits (pictures of 1 or 0), determined based on the action X. For the causal imitator (*ci*), we learn a POSCM $\hat{M}$ such that $P(x, w, s; \hat{M}) = P(x, w, s)$. To obtain $\hat{M}$, we train a GAN to imitate the observational distribution $P(x, w, s)$, with a separate generator for each X, W, S. We then train a separate discriminator measuring the distance between observed trajectories $P(s)$ and interventional distribution $P(s|\text{do}(\pi); \hat{M})$ over the i-surrogate $\{S\}$. The imitating policy is obtained by minimizing such a distance. Distributions $P(y|\text{do}(\pi))$ induced by all algorithms are reported in Fig. 4c. We find that the causal approach (*ci*) successfully imitates $P(y)$ (L1 $= 0.0634$). As expected, the naive approach (*bc*) mimicking distribution $P(x)$ is unable to imitate the expert (L1 = 0.1900).

5 Conclusion

We introduce a graphical criterion that is complete (i.e., sufficient and necessary) for determining the feasibility of learning an imitating policy that mimics the expert's performance from combinations of demonstration data and qualitative knowledge about the data-generating process represented as a causal diagram. We also study a data-dependent notion of imitability depending on the observational distribution. We develop an algorithm that finds an imitating policy, by exploiting quantitative knowledge contained in the observational data and the presence of surrogate endpoints, accompanied by an efficient procedure for estimating such imitating policy from observed expert's trajectories.

References

[1] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1, 2004.

[2] B. D. Argall, S. Chernova, M. Veloso, and B. Browning. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483, 2009.

[3] E. Bareinboim and J. Pearl. Causal inference and the data-fusion problem. *Proceedings of the National Academy of Sciences*, 113:7345–7352, 2016.

[4] A. Billard, S. Calinon, R. Dillmann, and S. Schaal. Survey: Robot programming by demonstration. *Handbook of robotics*, 59(BOOK_CHAP), 2008.

[5] J. Correa and E. Bareinboim. From statistical transportability to estimating the effect of stochastic interventions. In S. Kraus, editor, *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 1661–1667, Macao, China, 2019. International Joint Conferences on Artificial Intelligence Organization.

[6] J. Correa and E. Bareinboim. A calculus for stochastic interventions: Causal effect identification and surrogate experiments. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, New York, NY, 2020. AAAI Press.

[7] P. de Haan, D. Jayaraman, and S. Levine. Causal confusion in imitation learning. In *Advances in Neural Information Processing Systems*, pages 11693–11704, 2019.

[8] J. Etesami and P. Geiger. Causal transfer for imitation learning and decision making under sensor-shift. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, New York, NY, 2020. AAAI Press.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[10] O. Goudet, D. Kalainathan, P. Caillou, I. Guyon, D. Lopez-Paz, and M. Sebag. Causal Generative Neural Networks. *arXiv:1711.08936 [stat]*, Nov. 2017.

[11] O. Goudet, D. Kalainathan, P. Caillou, D. Lopez-Paz, I. Guyon, M. Sebag, A. Tritas, and P. Tubaro. Learning Functional Causal Models with Generative Neural Networks. *arXiv:1709.05321 [stat]*, Sept. 2017.

[12] R. D. Hjelm, A. P. Jacob, T. Che, A. Trischler, K. Cho, and Y. Bengio. Boundary-Seeking Generative Adversarial Networks. *arXiv:1702.08431 [cs, stat]*, Feb. 2018.

[13] Y. Huang and M. Valtorta. Pearl's calculus of intervention is complete. In R. Dechter and T. Richardson, editors, *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, pages 217–224. AUAI Press, Corvallis, OR, 2006.

[14] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017.

[15] M. Kocaoglu, C. Snyder, A. G. Dimakis, and S. Vishwanath. CausalGAN: Learning Causal Implicit Generative Models with Adversarial Training. *arXiv:1709.02023 [cs, math, stat]*, Sept. 2017.

[16] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[17] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2118–2125, 2018.

[18] S. Lee and E. Bareinboim. Structural causal bandits with non-manipulable variables. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, pages 4164–4172, Honolulu, Hawaii, 2019. AAAI Press.

[19] C. Louizos, U. Shalit, J. Mooij, D. Sontag, R. Zemel, and M. Welling. Causal Effect Inference with Deep Latent-Variable Models. *arXiv:1705.08821 [cs, stat]*, May 2017.

[20] J. Mahler and K. Goldberg. Learning deep policies for robot bin picking by simulating robust grasping sequences. In *Conference on robot learning*, pages 515–524, 2017.

[21] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. L. Cun. Off-road obstacle avoidance through end-to-end learning. In *Advances in neural information processing systems*, pages 739–746, 2006.

[22] K. Mülling, J. Kober, O. Kroemer, and J. Peters. Learning to select and generalize striking movements in robot table tennis. *The International Journal of Robotics Research*, 32(3):263–279, 2013.

[23] A. Y. Ng, S. J. Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, pages 663–670, 2000.

[24] X. Nguyen, M. J. Wainwright, and M. I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, Nov. 2010.

[25] S. Nowozin, B. Cseke, and R. Tomioka. F-GAN: Training Generative Neural Samplers using Variational Divergence Minimization. *arXiv:1606.00709 [cs, stat]*, June 2016.

[26] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, J. Peters, et al. An algorithmic perspective on imitation learning. *Foundations and Trends in Robotics*, 7(1-2):1–179, 2018.

[27] J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, 2000. 2nd edition, 2009.

[28] J. Pearl. Remarks on the method of propensity scores. *Statistics in Medicine*, 28:1415–1416, 2009. <http://ftp.cs.ucla.edu/pub/stat_ser/r345-sim.pdf>.

[29] D. A. Pomerleau. Alvinn: An autonomous land vehicle in a neural network. In *Advances in neural information processing systems*, pages 305–313, 1989.

[30] P. Rosenbaum and D. Rubin. The central role of propensity score in observational studies for causal effects. *Biometrika*, 70:41–55, 1983.

[31] S. Russell and P. Norvig. Artificial intelligence: a modern approach. 2002.

[32] I. Shpitser and J. Pearl. Identification of joint interventional distributions in recursive semi-Markovian causal models. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, pages 1219–1226. 2006.

[33] I. Shpitser and E. Sherman. Identification of personalized effects associated with causal pathways. In *UAI*, 2018.

[34] P. Spirtes, C. N. Glymour, and R. Scheines. *Causation, prediction, and search*. MIT press, 2000.

[35] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 1998.

[36] U. Syed and R. E. Schapire. A game-theoretic approach to apprenticeship learning. In *Advances in neural information processing systems*, pages 1449–1456, 2008.

[37] J. Tian. *Studies in Causal Reasoning and Learning*. PhD thesis, Computer Science Department, University of California, Los Angeles, CA, November 2002.

[38] J. Tian. Identifying dynamic sequential plans. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, pages 554–561, 2008.

[39] J. Tian and J. Pearl. A general identification condition for causal effects. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence*, pages 567–573, Menlo Park, CA, 2002. AAAI Press/The MIT Press.

[40] J. Tian and J. Pearl. A general identification condition for causal effects. Technical Report R-290-A, Department of Computer Science, University of California, Los Angeles, CA, 2003.

[41] B. van der Zander, M. Liśkiewicz, and J. Textor. Constructing separators and adjustment sets in ancestral graphs. In *Proceedings of the UAI 2014 Conference on Causal Inference: Learning and Prediction-Volume 1274*, pages 11–24, 2014.

[42] C. J. Watkins and P. Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

[43] B. WIDROW. Pattern-recognizing control systems. *Compurter and Information Sciences*, 1964.

[44] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

A Proofs

Corollary 1. *Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{V}$. If not all variables in $\boldsymbol{Y}$ are observed (i.e., $\boldsymbol{Y} \cap \boldsymbol{L} \neq \emptyset$), $P(\boldsymbol{y}|do(\pi))$ is not identifiable.*

Proof. Let $Y \in \boldsymbol{Y} \cap \boldsymbol{L}$. For any SCM M_1 that induces $\mathcal{G}$, we could obtain an SCM M_2 by replacing f_Y and $P(u_Y)$ associated with Y. Since there is no restriction on the parametrization of f_Y and $P(u_Y)$, we could always ensure $P(y|\text{do}(\pi); M_1) \neq P(y|\text{do}(\pi); M_2)$. For example, in both M_1, M_2, let $Y \leftarrow U_Y$; we define $P(U_Y = 0; M_1) = 0.1$ and $P(U_Y = 0; M_2) = 0.9$. It is immediate to see that $P(\boldsymbol{y}|\text{do}(\pi); M_1) \neq P(\boldsymbol{y}|\text{do}(\pi); M_2)$, i.e., $P(\boldsymbol{y}|\text{do}(\pi))$ is not identifiable. □

Lemma 1. *Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{V}$. $P(\boldsymbol{y})$ is not imitable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if there exists two POSCMs $M_1, M_2 \in \mathscr{M}(\mathcal{G})$ satisfying $P(\boldsymbol{o}; M_1) = P(\boldsymbol{o}; M_2)$ while there exists no policy $\pi \in \Pi$ such that for $i = 1, 2$, $P(\boldsymbol{y}|do(\pi); M_i) = P(\boldsymbol{y}; M_i)$.*

Proof. The lack of existence of a shared imitating policy between M_1, M_2 eliminates the possibility of the existence of a function from $P(\boldsymbol{o})$ to a policy π that imitates $P(\boldsymbol{y})$ in any POSCM compatible with the causal diagram $\mathcal{G}$. □

Theorem 1 (Imitation by Direct Parents)**.** *Given a causal diagram $\mathcal{G}$ and a policy space Π, $P(y)$ is imitable w.t.r. $\langle \mathcal{G}, \Pi \rangle$ if $pa(X)_{\mathcal{G}} \subseteq Pa(\Pi)$ and there is no bi-directed arrow pointing to X in $\mathcal{G}$. Moreoever, the imitating policy $\pi \in \Pi$ is given by $\pi(x|pa(\Pi)) = P(x|pa(X)_{\mathcal{G}})$.*

Proof. Since there is not bi-directed arrow pointing into X, it is variables that $pa(X)_{\mathcal{G}}$ is i-backdoor admissible relative to $\langle X, Y \rangle$ in $\mathcal{G}$. By Rule 2 of do-calculus, we have

$$P(y) = \sum_{x, pa(X)_{\mathcal{G}}} P(pa(X)_{\mathcal{G}}) P(x|pa(X)_{\mathcal{G}}) P(y|\text{do}(x), pa(X)_{\mathcal{G}}).$$

Since $pa(X)_{\mathcal{G}} \subseteq Pa(\Pi)$, the conditional distribution $P(x|pa(X)_{\mathcal{G}})$ can be represented as a policy in Π. Let $\pi(x|Pa(\Pi)) = \pi(x|pa(X)_{\mathcal{G}}) = P(x|pa(X)_{\mathcal{G}})$. We must have

$$P(y) = \sum_{x, pa(X)_{\mathcal{G}}} P(pa(X)_{\mathcal{G}}) \pi(x|pa(X)_{\mathcal{G}}) P(y|\text{do}(x), pa(X)_{\mathcal{G}}) = P(y|\text{do}(\pi)).$$ □

Lemma 5. *Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Z} = An(Y)_{\mathcal{G}} \cap Pa(\Pi)$. If $P(y)$ is imitable relative to $\langle \mathcal{G}, \Pi \rangle$, then $(Y \perp\!\!\!\perp X|\boldsymbol{Z})_{\mathcal{G}_{\underline{X}}}$.*

Proof. We consider first a simplified causal diagram $\mathcal{H}$ where all endogenous variables are observed, i.e., $\boldsymbol{V} = \boldsymbol{O}$; and $Pa(\Pi) = \boldsymbol{O} \setminus \{X, Y\}$. In this diagram $\mathcal{G}$, $(Y \not\perp\!\!\!\perp X|\boldsymbol{Z})_{\mathcal{G}_{\underline{X}}}$ implies that there exists a bi-directed path l of the form $X \leftrightarrow Z_1 \leftrightarrow Z_2 \leftrightarrow \cdots \leftrightarrow Z_n \leftrightarrow Y$ such that $Z_i \in \boldsymbol{Z}$ for any $i = 1, \ldots, n$. We denote by $\boldsymbol{Z}^* = \{Z_1, \ldots, Z_n\}$. Recall that $X \in An(Y)_{\mathcal{H}}$. We could thus obtain a subgraph $\mathcal{H}'$ of $\mathcal{H}$ that satisfies the following condition:

- $\mathcal{H}'$ contains the bi-directed path l.
- All nodes in $\mathcal{H}'$ are descendants of $\boldsymbol{Z}^* \cup \{X\}$.
- Y is a descendant for all nodes in $\mathcal{H}'$.
- Every endogenous node V in $\mathcal{H}'$ has at most one child.
- All bi-directed arrows in $\mathcal{H}'$ are contained in l.

A mental image for depicting $\mathcal{H}'$ is to think of a tree rooted in node Y; $X, Z_1, \ldots, Z_n$ are leaf nodes; nodes $X, Z_1, \ldots, Z_2, Y$ are connected by the bi-directed path l; $\mathcal{H}'$ contain no bi-directed arrow except path l. We now construct a POSCM M' compatible with $\mathcal{G}'$. More specifically, values of each exogenous confounder U_i residing on l are drawn uniformly over a binary domain $\{0, 1\}$. For each endogenous variable V_i in $\mathcal{H}'$, its values is equal to the parity sum of its parents in $\mathcal{H}'$, i.e., $V_i \leftarrow \oplus_{V_j \in pa(V_i)_{\mathcal{H}'}} V_j$. By construction, in the subgraph $\mathcal{H}'$, each exogenous U_i has exactly two

directed paths going to Y. This means that in the constructed model M', observed values of Y is always equal to 0, i.e., the reward distribution $P(Y = 0; M') = 1$.

Consider a policy space Π' of the form $\{\pi : \mathscr{D}_{\boldsymbol{Z}^*} \mapsto \mathscr{P}_X\}$. Let U_n denote the exogenous confounder residing on l that is closest to Y, i.e., $X \leftrightarrow Z_1 \leftrightarrow Z_2 \leftrightarrow \cdots \leftrightarrow Z_n \leftarrow U_n \rightarrow Y$. By the definition of M', values of each variable $Z_i \in \boldsymbol{Z}^*$ is the parity sum of at least two exogenous variables U_j, U_k. Since values of each U_i are drawn uniformly over $\{0, 1\}$, we must have $P(u_n, \boldsymbol{z}^*) = P(u_n)P(\boldsymbol{z}^*)$, or equivalently, $P(u_n|\boldsymbol{z}^*) = P(u_n)$. By definition, we could obtain from l a directed path going from U_n to Y that is not intercepted by $\boldsymbol{Z}^*$. That is, given any value $\boldsymbol{Z}^* = \boldsymbol{z}^*$, values of Y is decided by a parity function taking U_n as an input. Since $P(u_n|\boldsymbol{z}^*) = P(u_n)$ and U_n is drawn uniformly over $\{0, 1\}$, it is verifiable that in M', given any $\boldsymbol{Z}^* = \boldsymbol{z}^*$, the conditional distribution $P(Y = 0|\text{do}(x), \boldsymbol{z}^*; M') = 0.5$. This means that for any policy $\pi \in \Pi'$, the interventional distribution $P(Y = 0|\text{do}(\pi); M') = 0.5$, which is far from the observational distribution $P(Y = 0; M') = 1$. That is, $P(y)$ is not imitable w.r.t. $\langle \mathcal{H}', \Pi' \rangle$.

We will next show the non-imitability of $P(y)$ w.r.t. $\langle \mathcal{H}, \Pi \rangle$. For any node V that is not included in $\mathcal{H}'$, let its values be decided by an independent noise U_V drawn uniformly over $\{0, 1\}$. We denote this extended POSCM by M. Obviously, M is compatible with the causal diagram $\mathcal{H}$. In this model, for any covariate $V \in Pa(\Pi) \setminus \boldsymbol{Z}^*$, it is either (1) a random variable disconnected to any other endogenous variable in M; or (2) decided by a function taking $\boldsymbol{Z}^*$ as input. That is, $V \in Pa(\Pi) \setminus \boldsymbol{Z}^*$ contains no value of information with regard to the reward Y when intervening on action X. By [16, Ch. 23.6], this means that for any policy $\pi \in \Pi$, there exists a policy π' in the subspace Π' such that $P(y|\text{do}(\pi); M) = P(y|\text{do}(\pi'); M)$. Recall that there exists no policy π in Π' that could ensure $P(y|\text{do}(\pi); M') = P(y; M')$ in POSCM M'. By definition of M and M', we must have $P(y; M) = P(y; M')$ and for any policy $\pi \in \Pi$, $P(y|\text{do}(\pi); M) = P(y|\text{do}(\pi); M')$. It is immediate to see that there exists no policy $\pi \in \Pi$ that could induce $P(y|\text{do}(\pi); M) = P(y; M)$ in the extended POSCM M, i.e., $P(y)$ is not imitable w.r.t. $\langle \mathcal{H}, \Pi \rangle$.

We now consider a general causal diagram $\mathcal{G}$ where arbitrary latent endogenous variables $\boldsymbol{L}$ exist and $Pa(\Pi) \subset \boldsymbol{O} \setminus \{X, Y\}$. We will apply the latent projection [37, Def. 5] to transforms $\mathcal{G}$ into a simplified causal diagram $\mathcal{H}$ discussed above. More specifically, we construct a causal diagram $\mathcal{G}'$ from $\mathcal{G}$ by marking each $V \in \boldsymbol{V} \setminus (Pa(\Pi) \cup \{X, Y\})$ latent. We then apply Alg. 2 and obtain a simplified diagram $\mathcal{H} = \textsc{Project}(\mathcal{G}')$ where $\boldsymbol{V} = \boldsymbol{O} = Pa(\Pi) \cup \{X, Y\}$. Since $\textsc{Project}$ preserves topological relationships among observed nodes [37, Lem. 5], $(Y \not\perp\!\!\!\perp X|\boldsymbol{Z})_{\mathcal{G}_{\underline{X}}}$ implies $(Y \not\perp\!\!\!\perp X|\boldsymbol{Z})_{\mathcal{H}_{\underline{X}}}$. Following our previous argument, there exists a POSCM M' associated with $\mathcal{H}$ such that for any policy $\pi \in \Pi$, $P(y|\text{do}(\pi); M') \neq P(y; M')$. By Lem. 8, we could construct a POSCM M associated with $\mathcal{G}$ such that for any $\pi \in \Pi$, $P(y|\text{do}(\pi); M) = P(y|\text{do}(\pi); M')$ and $P(y; M) = P(y; M')$. It follows immediately that for any policy $\pi \in \Pi$, $P(y|\text{do}(\pi); M) \neq P(y; M)$, i.e., $P(y)$ is not imitable w.r.t. $\langle \mathcal{G}, \Pi \rangle$. □

Theorem 2 (Imitation by Backdoor)**.** *Given a causal diagram $\mathcal{G}$ and a policy space Π, $P(y)$ is imitable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if and only if there exists an i-backdoor admissible set $\boldsymbol{Z}$ w.r.t. $\langle \mathcal{G}, \Pi \rangle$. Moreover, the imitating policy $\pi \in \Pi$ is given by $\pi(x|pa(\Pi)) = P(x|\boldsymbol{z})$.*

Proof. We first prove the the "if" direction. Given an i-backdoor admissible set $\boldsymbol{Z}$ relative to $\langle X, Y \rangle$ in $\mathcal{G}$. By Rule 2 of do-calculus, we have,

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

Since $\boldsymbol{Z} \subseteq Pa(\Pi)$, the conditional distribution $P(x|\boldsymbol{z})$ can be represented as a policy in Π. Let $\pi(x|Pa(\Pi)) = \pi(x|\boldsymbol{z}) = P(x|\boldsymbol{z})$. We must have

$$P(y) = \sum_{x,\boldsymbol{z}} P(\boldsymbol{z})\pi(x|\boldsymbol{z})P(y|\text{do}(x), \boldsymbol{z}) = P(y|\text{do}(\pi)).$$

We now consider the "only if" direction. Suppose there exists no i-backdoor admissible set $\boldsymbol{Z}$ relative to $\langle X, Y \rangle$ in $\mathcal{G}$. We must have that $\boldsymbol{Z} = An(Y)_{\mathcal{G}} \cap Pa(\Pi)$ is not i-backdoor admissible, i.e., the independent relationship $(Y \perp\!\!\!\perp X|\boldsymbol{Z})_{\mathcal{G}_{\underline{X}}}$ does not hold. It follows immediately from Lem. 5 that $P(y)$ is not imitable relative to $\langle \mathcal{G}, \Pi \rangle$. □

Lemma 2. *Given a causal diagram $\mathcal{G}$, a policy space Π, and an observational distribution $P(\boldsymbol{o})$, let $\langle \mathcal{G}, \Pi' \rangle$ be an i-instrument w.r.t. $\langle \mathcal{G}, \Pi \rangle$. If for an arbitrary $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$, there exists a policy*

$\pi \in \Pi'$ *such that* $P(\boldsymbol{s}|do(\pi); M) = P(\boldsymbol{s})$, $P(y)$ *is p-imitable w.r.t.* $\langle \mathcal{G}, \Pi, P(\boldsymbol{o})\rangle$. *Moreover,* π *is an imitating policy for* $P(y)$ *w.r.t.* $\langle \mathcal{G}, \Pi, P(\boldsymbol{o})\rangle$.

Proof. Since $\boldsymbol{S}$ is an i-surrogate relative to $\langle \mathcal{G}, \Pi'\rangle$, by definition, we have $(Y \perp\!\!\!\perp \hat{X}|\boldsymbol{S})_{\mathcal{G}\cup\Pi'}$. For any causal diagram $\mathcal{G}$ and policy space Π, let $\mathcal{G}_\Pi$ denote a manipulated diagram obtained from $\mathcal{G}$ by adding arrows from nodes in $Pa(\Pi)$ to X in the subgraph $\mathcal{G}_{\overline{X}}$. By definition, it is obvious that $\mathcal{G} \cup \Pi'$ is a supergraph containing both $\mathcal{G}$ and $\mathcal{G}_{\Pi'}$. By definition of d-separation, we must have $(Y \perp\!\!\!\perp \hat{X}|\boldsymbol{S})_{\mathcal{G}}$ and $(Y \perp\!\!\!\perp \hat{X}|\boldsymbol{S})_{\mathcal{G}_{\Pi'}}$. The basic operations of distribution marginalization implies, for any POSCM $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$,

$$\begin{aligned} P(y|\text{do}(\pi); M) &= \sum_{\boldsymbol{s}} P(y|\boldsymbol{s}, \text{do}(\pi); M) P(\boldsymbol{s}|\text{do}(\pi); M) \\ &= \sum_{\boldsymbol{s}} P(y|\boldsymbol{s}, \text{do}(x); M) P(\boldsymbol{s}|\text{do}(\pi); M) \\ &= \sum_{\boldsymbol{s}} P(y|\boldsymbol{s}; M) P(\boldsymbol{s}|\text{do}(\pi); M) \end{aligned}$$

The last two steps hold since $(Y \perp\!\!\!\perp \hat{X}|\boldsymbol{S})_{\mathcal{G}_{\Pi'}}$ and $(Y \perp\!\!\!\perp \hat{X}|\boldsymbol{S})_{\mathcal{G}}$. Fix an arbitrary POSCM $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$. Suppose there exists an imitating policy $\pi \in \Pi'$ in M such that $P(\boldsymbol{s}|\text{do}(\pi); M) = P(\boldsymbol{s}; M)$. We must have

$$P(y|\text{do}(\pi); M) = \sum_{\boldsymbol{s}} P(y|\boldsymbol{s}; M) P(\boldsymbol{s}|\text{do}(\pi); M) = \sum_{\boldsymbol{s}} P(y|\boldsymbol{s}; M) P(\boldsymbol{s}; M) = P(y; M). \quad (2)$$

Since $\langle \mathcal{G}, \Pi'\rangle$ is an i-instrument w.r.t. $\langle \mathcal{G}, \Pi\rangle$, $P(\boldsymbol{s}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi'\rangle$. This means that $P(\boldsymbol{s}|\text{do}(\pi); M)$ is uniquely computable from $P(\boldsymbol{o}; M)$ in any POSCM $M \in \mathscr{M}(\mathcal{G})$. That is, for any POSCM $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$ where its observational distribution $P(\boldsymbol{o}; M) = P(\boldsymbol{o})$, the interventional distribution $P(\boldsymbol{s}|\text{do}(\pi); M)$ for any policy $\pi \in \Pi'$ remains as an invariant. This implies that the derivation in Eq. (2) is applicable for any POSCM $M \in \mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$, i.e., $P(y)$ is p-imitable w.r.t. $\langle \mathcal{G}, \Pi, P(\boldsymbol{o})\rangle$. □

Lemma 3. *Given a causal diagram* $\mathcal{G}$, *a policy space* Π, *let a subspace* $\Pi' \subseteq \Pi$. *If there exists* $\boldsymbol{S} \subseteq \boldsymbol{O}$ *such that* $\langle \boldsymbol{S}, \Pi'\rangle$ *is an i-instrument w.r.t.* $\langle \mathcal{G}, \Pi\rangle$, Π' *is an i-subspace w.r.t.* $\langle \mathcal{G} \cup \{Y\}, \Pi, Y\rangle$.

Proof. Let $\boldsymbol{S}$ be an i-surrogate w.r.t. $\langle \mathcal{G}, \Pi'\rangle$. Following the proof of Lem. 2,

$$\begin{aligned} P(y|\text{do}(\pi)) &= \sum_{\boldsymbol{s}} P(y|\boldsymbol{s}, \text{do}(\pi)) P(\boldsymbol{s}|\text{do}(\pi)) \\ &= \sum_{\boldsymbol{s}} P(y|\boldsymbol{s}, \text{do}(x)) P(\boldsymbol{s}|\text{do}(\pi)) \\ &= \sum_{\boldsymbol{s}} P(y|\boldsymbol{s}) P(\boldsymbol{s}|\text{do}(\pi)). \end{aligned}$$

Suppose $\langle \boldsymbol{S}, \Pi'\rangle$ form an i-instrument w.r.t. $\langle \mathcal{G}, \Pi\rangle$, i.e., $P(\boldsymbol{s}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi'\rangle$. The above equation implies that $P(y|\text{do}(\pi))$ can be uniquely determined from $P(\boldsymbol{o}, y)$ in any POSCM that induces $\mathcal{G}$. That is, $P(y|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G} \cup \{Y\}, \Pi\rangle$, which completes the proof. □

Lemma 4. *Given a causal diagram* $\mathcal{G}$, *a policy space* Π, *an observational distribution* $P(\boldsymbol{o})$ *and a subset* $\boldsymbol{S} \subseteq \boldsymbol{O}$. $P(\boldsymbol{s})$ *is p-imitable only if for any* $\boldsymbol{S}' \subseteq \boldsymbol{S}$, $P(\boldsymbol{s}')$ *is p-imitable w.r.t.* $\langle \mathcal{G}, \Pi, P(\boldsymbol{o})\rangle$.

Proof. For any POSCM M in $\mathscr{M}(\mathcal{G}, P(\boldsymbol{o}))$, if there exists a policy $\pi \in \Pi$ such that $P(\boldsymbol{s}|\text{do}(\pi); M) = P(\boldsymbol{s}; M)$, we must have for any $\boldsymbol{S}' \subseteq \boldsymbol{S}$,

$$P(\boldsymbol{s}'|\text{do}(\pi); M) = \sum_{\boldsymbol{s}\setminus\boldsymbol{s}'} P(\boldsymbol{s}|\text{do}(\pi); M) = \sum_{\boldsymbol{s}\setminus\boldsymbol{s}'} P(\boldsymbol{s}; M) = P(\boldsymbol{s}; M).$$

which completes the proof. □

Theorem 3. *Given a causal diagram* $\mathcal{G}$, *a policy space* Π, *and an observational distribution* $P(\boldsymbol{o})$, *if* IMITATE *returns a policy* $\pi \in \Pi$, $P(y)$ *is p-imitable w.r.t.* $\langle \mathcal{G}, \Pi, P(\boldsymbol{o})\rangle$.

Proof. For a policy subspace Π', Step 3 outputs an i-surrogate $\boldsymbol{S}$ w.r.t. $\langle \mathcal{G}, \Pi' \rangle$. Step 4 ensures that Π' is an i-subspace w.r.t. $\langle \mathcal{G}, \Pi, \boldsymbol{S} \rangle$. That is, $\langle \boldsymbol{S}, \Pi' \rangle$ forms an i-instrument w.r.t. $\langle \mathcal{G}, \Pi \rangle$. Since IMITATE only outputs a policy π when $P(\boldsymbol{s})$ is p-imitable w.r.t. $\langle \mathcal{G}, \Pi', P(\boldsymbol{o}) \rangle$, the statement follows from Lem. 2. □

B Causal Identification in POSCMs

In this section, we introduce algorithms for identifying causal effects in Partially Observable Structural Causal Models (POSCMs), which are sufficient and complete. We will consistently assume that $\boldsymbol{Y} \subseteq \boldsymbol{O}$, i.e., the primary outcomes $\boldsymbol{Y}$ are all observed. For settings where $\boldsymbol{Y} \not\subseteq \boldsymbol{O}$, Corol. 1 implies that $P(\boldsymbol{y}|\text{do}(\pi))$ is always non-identifiable w.r.t. the causal diagram $\mathcal{G}$. For convenience, we focus on the problem of determining whether the target effect is identifiable w.r.t. $\mathcal{G}$. However, our algorithms could be easily extended to derive identification formulas of the causal effect.

We start with the identificaiton of causal effects induced by atomic interventions $\text{do}(x)$. Formally,

Definition 9 (Identifiability (Atomic Interventions))**.** Given a causal diagram $\mathcal{G}$, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{O}$. $P(\boldsymbol{y}|\text{do}(x))$ is said to be identifiable w.r.t. $\mathcal{G}$ if $P(\boldsymbol{y}|\text{do}(x); M)$ is uniquely computable from $P(\boldsymbol{o}; M)$ and π for any POSCM $M \in \mathscr{M}(\mathcal{G})$.

A causal diagram $\mathcal{G}$ is said to be semi-Markovian if it does not contain any latent endogenous variables $\boldsymbol{L}$; or equivalently, $\boldsymbol{V} = \boldsymbol{O}$. [37, Alg. 5] is an algorithm that identify causal effects, say $P(\boldsymbol{y}|\text{do}(\boldsymbol{x}))$, from the observational distribution $P(\boldsymbol{o})$ in a semi-Markovian diagram $\mathcal{G}$, which we consistently refer to as IDENTIFYHELPER. More specifically, IDENTIFYHELPER takes as input a set of action variables $\boldsymbol{X} \subseteq \boldsymbol{O}$, a set of outcome variables $\boldsymbol{Y} \subseteq \boldsymbol{O}$ and a semi-Markovian causal diagram $\mathcal{G}$[5]. If $P(\boldsymbol{y}|\text{do}(\boldsymbol{x}))$ is identifiable w.r.t. $\mathcal{G}$, IDENTIFYHELPER returns an identificaiton formula that represents $P(\boldsymbol{y}|\text{do}(\boldsymbol{x}))$ as an algebraic expression of the observational distribution $P(\boldsymbol{o})$; otherwise, IDENTIFYHELPER returns "FAIL". [13, 32] showed that IDENTIFYHELPER is complete from identifying effects from observational data with respect to semi-Markovian causal diagrams.

We will utilize IDENTIFYHELPER to identifying causal effects in POSCMs where latent endogenous variables are allowed. The key to this reduction is an algorithm PROJECT [37, Def. 5] that transforms an arbitrary causal diagram $\mathcal{G}$ with observed endogenous variables $\boldsymbol{O}$ into a semi-Markovian causal diagram $\mathcal{H}$ such that its endogenous variables $\boldsymbol{V} = \boldsymbol{O}$. For completeness, we rephrase PROJECT and describe it in Alg. 2.

Algorithm 2: PROJECT [37, Def. 5]

Input: A causal diagram $\mathcal{G}$.
Output: A causal diagram $\mathcal{H}$ where all endogenous variables are observed, i.e., $\boldsymbol{V} = \boldsymbol{O}$.
Let $\boldsymbol{O}, \boldsymbol{L}$ be, respectively, observed endogenous variables and latent endogenous variables in $\mathcal{G}$.
Let $\mathcal{H}$ be a causal diagram constructed as follows.
for each observed $V \in \boldsymbol{O}$ in $\mathcal{G}$ **do**
    Add an observed node V in $\mathcal{H}$.
end for
for each pair $S, E \in \boldsymbol{O}$ in $\mathcal{G}$ s.t. $S \neq E$ **do**
    if there exists a directed path $S \to E$ in $\mathcal{G}$ **then**
        Add an edge $S \to E$ in $\mathcal{H}$.
    else if there exists a path $S \to V_1 \to \cdots \to V_n \to E$ in $\mathcal{G}$ s.t. $V_1, \cdots V_n \in \boldsymbol{L}$ **then**
        Add an edge $S \to E$ in $\mathcal{H}$.
    else if there exists a bidirected edge $S \leftrightarrow E$ in $\mathcal{G}$ **then**
        Add a bidirected edge $S \leftrightarrow E$ in $\mathcal{H}$.
    else if there exists a path $S \leftarrow V_{l,1} \leftarrow \cdots \leftarrow V_{l,n} \leftrightarrow V_{r,m} \to \cdots \to V_{r,1} \to E$ in $\mathcal{G}$ s.t. $V_{l,1}, \cdots, V_{l,n}, V_{r,1}, \cdots, V_{r,m} \in \boldsymbol{L}$ **then**
        Add a bidirected edge $S \leftrightarrow E$ in $\mathcal{H}$.
    else if there exists a path $S \leftarrow V_{l,1} \leftarrow \cdots \leftarrow V_{l,n} \leftarrow V_c \to V_{r,m} \to \cdots \to V_{r,1} \to E$ in $\mathcal{G}$ s.t. $V_{l,1}, \cdots, V_{l,n}, V_c, V_{r,1}, \cdots, V_{r,m} \in \boldsymbol{L}$ **then**
        Add a bidirected edge $S \leftrightarrow E$ in $\mathcal{H}$.
    end if
end for
Return $\mathcal{H}$.

[5] In the original text, the semi-Markovian causal diagram $\mathcal{G}$ is implicitly assumed; [37, Alg. 5] only takes $\boldsymbol{X}, \boldsymbol{Y}$ as input. We rephrase the algorithm and explicitly represent the dependency on the causal diagram $\mathcal{G}$.

The following lemma, introduced in [18, Props. 2-3], shows that the parameter space of observational distributions $P(\boldsymbol{o})$ and interventional distributions $P(\boldsymbol{y}|\text{do}(x))$ induced by POSCMs associated with a causal diagram $\mathcal{G}$ is always equivalent to that induced by the corresponding semi-Markovian causal diagram $\mathcal{H} = \text{PROJECT}(\mathcal{G})$.

Lemma 6. *Given a causal diagram $\mathcal{G}$, let $\mathcal{H} = \text{PROJECT}(\mathcal{G})$. For any POSCM M_1 associated with $\mathcal{G}$, there exists a POSCM M_2 associated with $\mathcal{H}$ such that $P(\boldsymbol{y}|do(\boldsymbol{x}); M_1) = P(\boldsymbol{y}|do(\boldsymbol{x}); M_2)$ for any $\boldsymbol{X}, \boldsymbol{Y} \subseteq \boldsymbol{O}$, and vice versa.*

Proof. The statement follows from [18, Props. 2-3]. □

Lem. 6 implies a general algorithm for identifying $P(\boldsymbol{y}|\text{do}(x))$ in a causal diagram $\mathcal{G}$ with latent endogenous variables: it is sufficient to consider the identifiability of $P(\boldsymbol{y}|\text{do}(x))$ in the projection $\mathcal{H} = \text{PROJECT}(\mathcal{G})$. We describe such an algorithm in Alg. 3.

Algorithm 3: IDENTIFY (Atomic Interventions)

1: **Input:** A causal diagram $\mathcal{G}$, primary outcomes $\boldsymbol{Y} \subseteq \boldsymbol{O}$.
2: Let $\mathcal{H} = \text{PROJECT}(\mathcal{G})$.
3: **if** $\text{IDENTIFYHELPER}(\{X\}, \boldsymbol{Y}, \mathcal{H}) = \text{FAIL}$ **then**
4: Return NO.
5: **else**
6: Return YES.
7: **end if**

Corollary 2. *Given a causal diagram $\mathcal{G}$, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{O}$.* $\text{IDENTIFY}(\mathcal{G}, \boldsymbol{Y}) =$ "YES" *if and only if $P(\boldsymbol{y}|do(x))$ is identifiable w.r.t. $\mathcal{G}$.*

Proof. Lem. 6 implies that $P(\boldsymbol{y}|\text{do}(x))$ is identifiable w.r.t. a causal diagram $\mathcal{G}$ if and only if $P(\boldsymbol{y}|\text{do}(x))$ is identifiable w.r.t. the projection $\mathcal{H} = \text{PROJECT}(\mathcal{G})$. To see this, suppose $P(\boldsymbol{y}|\text{do}(x))$ is not identifiable w.r.t. $\mathcal{G}$. That is, there exist two POSCMs M_1, M_2 associated with $\mathcal{G}$ such that $P(\boldsymbol{o}; M_1) = P(\boldsymbol{o}; M_2)$ while $P(\boldsymbol{y}|\text{do}(x); M_1) \neq P(\boldsymbol{y}|\text{do}(x); M_2)$. By Lem. 6, for any M_i and $i = 1, 2$, we could find a POSCM M_i' associated with $\mathcal{H}$ such that $P(\boldsymbol{o}; M_i) = P(\boldsymbol{o}; M_i')$ and $P(\boldsymbol{y}|\text{do}(x); M_i) = P(\boldsymbol{y}|\text{do}(x); M_i')$. This implies that we could obtain two POSCMs M_1', M_2' associated with $\mathcal{H}$ such that $P(\boldsymbol{o}; M_1') = P(\boldsymbol{o}; M_2')$ while $P(\boldsymbol{y}|\text{do}(x); M_1') \neq P(\boldsymbol{y}|\text{do}(x); M_2')$, i.e., $P(\boldsymbol{y}|\text{do}(x))$ is identifiable w.r.t. $\mathcal{H}$. Similarly, we could prove the "only if" direction.

Since IDENTIFY returns "YES" if and only if IDENTIFYHELPER finds an identification formula of $P(\boldsymbol{y}|\text{do}(x))$ and IDENTIFYHELPER is sound and complete, the statement is entailed. □

B.1 Identifying Conditional Plans

We will next study the general problem of identifying causal effects $P(\boldsymbol{y}|\text{do}(\pi))$ induced by interventions $\text{do}(\pi)$ following conditional plans in a policy space Π. The formal definition of such identifiability is given in Def. 2. Similar to atomic interventions, we consider first a simpler setting where the causal diagram $\mathcal{G}$ is semi-Markovian, without latent endogenous variables. Given a causal diagram $\mathcal{G}$, we denote by $\mathcal{G}_\Pi$ a manipulated diagram obtained from a subgraph $\mathcal{G}_{\overline{X}}$ by adding arrows from nodes in $Pa(\Pi)$ to the action node X. Let a set $\boldsymbol{Z} = an(\boldsymbol{Y})_{\mathcal{G}_\Pi} \setminus \{X\}$. Following [38, Eq. 15], the interventional distribution $P(\boldsymbol{y}|\text{do}(\pi))$ could be written as follows:

$$\begin{aligned} P(\boldsymbol{y}|\text{do}(\pi)) &= \sum_{x,\boldsymbol{z}} P(\boldsymbol{y}, \boldsymbol{z}|\text{do}(x), \text{do}(\boldsymbol{v} \setminus (\boldsymbol{y} \cup \boldsymbol{z} \cup \{x\})))\pi(x|pa(\Pi)) \\ &= \sum_{x,\boldsymbol{z}} P(\boldsymbol{y}, \boldsymbol{z}|\text{do}(x))\pi(x|pa(\Pi)). \end{aligned} \tag{3}$$

Among the above equations, the last step follows from Rule 3 of do-calculus [27, Thm. 3.4.1]. More specifically, if there is a directed path from a node $V_i \in \boldsymbol{V} \setminus (\boldsymbol{Y} \cup \boldsymbol{Z} \cup \{X\})$ to a node in $\boldsymbol{Y}, \boldsymbol{Z}$ in the subgraph $\mathcal{G}_{\overline{X}}$, V_i must also be included in set $\boldsymbol{Z}$. That is, $(\boldsymbol{Y}, \boldsymbol{Z} \perp\!\!\!\perp \boldsymbol{V} \setminus (\boldsymbol{Y} \cup \boldsymbol{Z} \cup \{X\}))_{\mathcal{G}_{\overline{X}, \overline{\boldsymbol{V} \setminus (\boldsymbol{Y} \cup \boldsymbol{Z} \cup \{X\})}}}$, which implies $P(\boldsymbol{y}, \boldsymbol{z}|\text{do}(x), \text{do}(\boldsymbol{v} \setminus (\boldsymbol{y} \cup \boldsymbol{z} \cup$

$\{x\}))) = P(\boldsymbol{y}, \boldsymbol{z}|\text{do}(x))$. It is thus sufficient to identify the causal effect $P(\boldsymbol{y}, \boldsymbol{z}|\text{do}(x))$ induced by atomic intervention $\text{do}(x)$ w.r.t. $\mathcal{G}$. [33, 5] showed that such an algorithm is complete for identifying $P(\boldsymbol{y}|\text{do}(\pi))$ in a semi-Markovian causal diagram $\mathcal{G}$ where all endogenous variables are observed.

Lemma 7. *Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{O}$. Assume that $\mathcal{G}$ is semi-Markovian, i.e., $\boldsymbol{V} = \boldsymbol{O}$. Let $\boldsymbol{Z} = an(\boldsymbol{Y})_{\mathcal{G}_\Pi} \setminus \{X\}$. $P(\boldsymbol{y}|do(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if and only if $P(\boldsymbol{y}, \boldsymbol{z}|do(x))$ is identifiable w.r.t. $\mathcal{G}$.*

Proof. The statement follows immediately from [5, Corol. 2]. □

We are now ready to consider the identificaiton of $P(\boldsymbol{y}|\text{do}(\pi))$ w.r.t. a policy space Π and a general causal diagram $\mathcal{G}$ where latent endogenous variables $\boldsymbol{L}$ are present. Our next result shows that it is sufficient to identify $P(\boldsymbol{y}|\text{do}(\pi))$ in the corresponding semi-Markovian projection $\mathcal{H} = \textsc{Project}(\mathcal{G})$.

Lemma 8. *Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\mathcal{H} = \textsc{Project}(\mathcal{G})$. For any POSCM M_1 associated with $\mathcal{G}$, there exists a POSCM M_2 associated with $\mathcal{H}$ such that $P(\boldsymbol{y}|do(\pi); M_1) = P(\boldsymbol{y}|do(\pi); M_2)$ for any $\pi \in \Pi$, any $\boldsymbol{Y} \subseteq \boldsymbol{O}$, and vice versa.*

Proof. For any policy $\pi \in \Pi$, $P(\boldsymbol{y}|\text{do}(\pi))$ could be written as a function of $P(\boldsymbol{y}, \boldsymbol{z}|\text{do}(x))$ and π following Eq. (3). Therefore, the statement is implied by Lem. 6. □

Algorithm 4: IDENTIFY

1: **Input:** A causal diagram $\mathcal{G}$, a policy space Π, primary outcomes $\boldsymbol{Y} \subseteq \boldsymbol{O}$.
2: Let $\mathcal{H} = \textsc{Project}(\mathcal{G})$.
3: Let $\boldsymbol{Z} = an(\boldsymbol{Y})_{\mathcal{H}_\Pi} \setminus \{X\}$.
4: **if** IDENTIFYHELPER$(\{X\}, \boldsymbol{Y} \cup \boldsymbol{Z}, \mathcal{H}) =$ FAIL **then**
5: Return NO.
6: **else**
7: Return YES.
8: **end if**

Details of our algorithm IDENTIFY is described in Alg. 4. It takes as input a causal diagram $\mathcal{G}$, a policy space Π and a set of observed outcomes $\boldsymbol{Y}$. At Step 1, it obtains a projection $\mathcal{H}$ of $\mathcal{G}$ such that all endogenous variables $\boldsymbol{V}$ in $\mathcal{H}$ are observed. It then constructs the covariates $\boldsymbol{Z}$ following Lem. 7 (Step 3). Finally, IDENTIFY calls IDENTIFYHELPER to identify $P(\boldsymbol{y}, \boldsymbol{z}|\text{do}(x))$ in the projection $\mathcal{H}$. It outputs "NO" if IDENTIFYHELPER fails to find an identification formula of $P(\boldsymbol{y}, \boldsymbol{z}|\text{do}(x))$; otherwise, it outputs "YES". Since Lem. 8 shows that the parameter space of $P(\boldsymbol{y}|\text{do}(\pi))$ and $P(\boldsymbol{o})$ induced by POSCMs $M \in \mathscr{M}(\mathcal{G})$ is equivalent to that induced by instances in the family $\mathscr{M}(\mathcal{H})$ of projection $\mathcal{H}$, the soundness and completeness of IDENTIFY is entailed.

Corollary 3. *Given a causal diagram $\mathcal{G}$ and a policy space Π, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{O}$.* $\textsc{Identify}(\mathcal{G}, \Pi, \boldsymbol{Y}) = \textsc{Yes}$ *if and only if $P(\boldsymbol{y}|do(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi \rangle$.*

Proof. Lem. 8 implies that $P(\boldsymbol{y}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{G}, \Pi \rangle$ if and only if $P(\boldsymbol{y}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{H}, \Pi \rangle$ where $\mathcal{H} = \textsc{Project}(\mathcal{G})$. The proof is similar to Corol. 2. It follows from Eq. (3) and Lem. 7 that IDENTIFYHELPER does not fail if and only if $P(\boldsymbol{y}|\text{do}(\pi))$ is identifiable w.r.t. $\langle \mathcal{H}, \Pi \rangle$. The soundness and completeness of *Identify* is thus entailed. □

C LISTIDSPACE

In this section, we describe Algorithm LISTIDSPACE that finds all identifiable subspaces with respect to a causal diagram $\mathcal{G}$, a policy space Π and a set of observed variables $\boldsymbol{Y} \subseteq \boldsymbol{O}$. The details of LISTIDSPACE are shown in Alg. 5.

Algorithm 5: LISTIDSPACE

1: **Input:** $\mathcal{G}, \Pi, \boldsymbol{Y}$.
2: LISTIDSPACEHELPER$(\mathcal{G}, \boldsymbol{Y}, \{\pi : \mathscr{P}_X\}, \Pi)$.

Algorithm 6: LISTIDSPACEHELPER

1: **Input:** $\mathcal{G}, \boldsymbol{Y}$ and policy spaces Π_L, Π_R such that $\Pi_L \subseteq \Pi_R$.
2: **if** IDENTIFY$(\mathcal{G}, \Pi_L, \boldsymbol{Y}) =$ YES **then**
3: **if** $\boldsymbol{L} = \boldsymbol{R}$ **then** Output Π_L.
4: **else**
5: Pick an arbitrary $V \in Pa(\Pi_R) \setminus Pa(\Pi_L)$.
6: LISTIDSPACEHELPER$(\mathcal{G}, \boldsymbol{Y}, \Pi_L \cup \{V\}, \Pi_R)$.
7: LISTIDSPACEHELPER$(\mathcal{G}, \boldsymbol{Y}, \Pi_L, \Pi_R \setminus \{V\})$.
8: **end if**
9: **end if**

It calls a subroutine LISTIDSPACEHELPER which takes as input the diagram $\mathcal{G}$, variables $\boldsymbol{Y}$ and two policy subspace Π_L, Π_R of Π such that $\Pi_L \subseteq \Pi_R$. More specifically, LISTIDSPACEHELPER performs backtrack search to enumerate identifiable subspaces Π' w.r.t. $\langle \mathcal{G}, \Pi, \boldsymbol{Y} \rangle$ such that $\Pi_L \subseteq \Pi' \subseteq \Pi_R$. It aborts branches that will not lead to such identifiable subspaces. The aborting criterion (Step 2 of Alg. 6) follows the observation that $P(\boldsymbol{y}|\text{do}(\pi))$ is identifiable w.r.t. a policy space Π only if it is identifiable w.r.t. any subspace $\Pi' \subseteq \Pi$. At Step 5, it picks an arbitrary variable V that is included in the input covariates of Π_R but not in Π_L. Let $\Pi \cup \{V\}$ denote a policy space obtained from Π by including V as part of the input covariates, i.e., $\Pi \cup \{V\} = \{\pi : \mathscr{D}_{Pa(\Pi),V} \mapsto \mathscr{P}_X\}$. Similarly, we define $\Pi \cup \{V\} = \{\pi : \mathscr{D}_{Pa(\Pi)\setminus\{V\}} \mapsto \mathscr{P}_X\}$. LISTIDSPACEHELPER then recursively returns all identifiable subspaces Π' w.r.t. $\langle \mathcal{G}, \Pi, \boldsymbol{Y} \rangle$: the first recursive call returns i-subspaces taking V as an input and the second call return all i-subspaces that does not consider V.

Theorem 4. *Given a causal diagram $\mathcal{G}$, a policy space Π, and an oracle access to* IDENTIFY*, let $\boldsymbol{Y}$ be an arbitrary subset of $\boldsymbol{O}$.* LISTIDSPACE$(\mathcal{G}, X, \boldsymbol{Y}, \emptyset, Pa(\Pi))$ *enumerates identifiable subspaces w.r.t. $\langle G, \Pi, \boldsymbol{Y} \rangle$ with polynomial delay $\mathcal{O}(|Pa(\Pi)|)$.*

Proof. The recursive calls at Steps 6 and 7 guarantees that LISTIDSPACEHELPER generates every i-subspaces Π' exactly once. Since every leaf will output an i-subspace, the tree height is at most $|Pa(\Pi)|$ and the existence check is performed by IDENTIFY oracle, the delay time is $\mathcal{O}(|Pa(\Pi)|)$. □

D Experiments

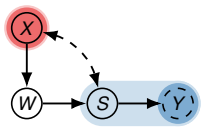
We perform 4 experiments, each designed to test different aspects of causal imitation learning. The basic results of the last two experiments are summarized in the paper's main text.

1. **Data-Dependent Binary Imitability** - given a binary model, we show that by sampling uniformly from binary distributions satisfying the graph in figure Fig. 1c (called the "front-door"), naïve imitation results in a biased answer. We also observe that in binary frontdoor models, 50% of distributions exhibit data-dependent imitability, despite not being imitable in generality.
2. **GAN-Based Binary Imitability** - Generative Adversarial Networks [9] allow explicit parameterization of a model, and can therefore be used to imitate causal mechanisms in the presence of confounding. We show in various binary models that a GAN-based approach to imitation leads to positive results, both with standard and data-dependent imitation.
3. **Highway Driving** - The binary models in experiment 2 show that GANs function in the causal imitation setting, however, one can simply use the corresponding formulae to efficiently get answers. In this experiment, we use the HighD dataset to make two variables continuous, and to show that naïvely choosing information to use for imitation can lead to bias.
4. **MNIST Digits** - The final experiment shows that the GAN-based approach is capable of handling complex, high-dimensional probability distributions. To show this, we perform data-dependent imitation on a frontdoor graph, replacing a node with pictures of MNIST digits, to represent a complex, high-dimensional probability distribution.

Experiments 3 and 4 are reported in the main text. However, each experiment builds upon the ideas introduced in the preceding experiment, so it is recommended that readers go in order. Details of each experiment are described in its own subsection. For an discrete variable X, we will consistently use x_i to represent an assignment $X = i$; therefore, we write $P(y_1|\text{do}(x_0)) = P(Y = 1|\text{do}(X = 0))$.

D.1 Data-Dependent Binary Imitability

If all variables in a causal model are discrete, one can find the imitating policy through the solution of a series of linear systems. As an example, in the front-door graph (Fig. 1c, and shown below), with all variables binary, and given an observational distribution $P(x, w, s)$ that is amenable to data-dependent imitation, the imitating policy π for X has probability of $\pi(x_1) = \alpha$:

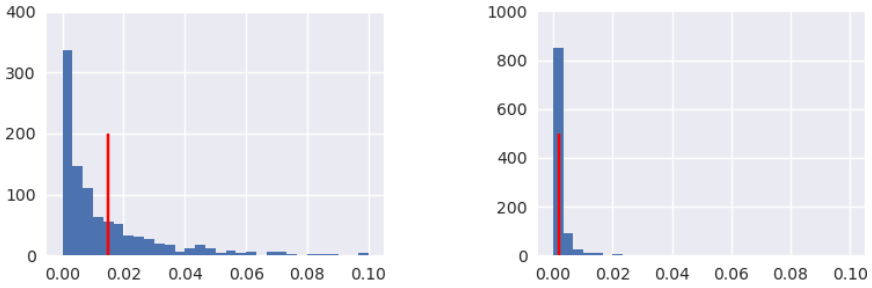


$$\alpha P(s|\text{do}(x_1)) + (1-\alpha)P(s|\text{do}(x_0)) = P(s)$$

$$\Rightarrow \alpha = \frac{P(s) - P(s|\text{do}(x_0))}{P(s|\text{do}(x_1)) - P(s|do(x=0))} \tag{4}$$

In this experiment, we show the bias of cloning $\pi(x) = P(x)$ as compared to imitation using Eq. (4) in binary models. It is important to note that we limited this experiment to binary models to permit the computation of optimal policies explicitly - higher dimensional/continuous models are likely to show different average bias due to their extra degrees of freedom.

We generate 1×10^5 random instances where X, W, S are binary variables. Probability distributions consistent with the graph decompose as $P(x, w, s, y) = P(x)P(w|x)P(s|x, w)P(y|s)$, so we draw uniformly over $[0, 1]$ for each conditional probability (i.e. $P(x), P(w|x), P(s|x, w), P(y|s) \sim U(0, 1)$) We report in Fig. 6 the L1 distance between the interventional distribution $P(y|\text{do}(\pi))$ induced by the imitator (*ci* and *bc*) and the actual expert's reward distribution $P(s)$ over 1×10^5 generated instances. The causal imitiation learning approach uses Eq. (4), while naïve cloning directly imitates $P(X)$, both using sample averages. We find that the causal imitation (*ci*, average L1 = 0.0016) dominates the naïve cloning approach (*bc*, average L1 = 0.0147). More interestingly, we find 50% of generated instances are p-imitable; this suggests that leveraging the observational distribution is beneficial in many imitation learning settings.



(a) *bc* (b) *ci*

Figure 6: When the models are sampled uniformly from probability distributions compatible with Fig. 1c, and for which there exists a perfect policy for imitating $P(y)$, a naïve cloning of $\pi(x) = P(x)$ (a) yields a biased answer, while our approach (b) gets the answer with only sampling error.

D.2 GAN-Based Binary Imitability

Given an identification formula for the causal effect of a target variable X on S, with X conditioned on a mediator W, there is a separate system of equations for each instantiation of the set W, akin to the one shown in Eq. (4). This means that the number of systems of equations to solve is exponential in the size of W, and can't easily be approached when the model contains continuous or high-dimensional variables.

To avoid reliance on these adjustment formulae, we solve for imitating policies through direct modeling of the SCM [19, 10]. In particular, we follow a similar procedure to [15, 11], by training a generative adversarial network (GAN) to imitate the observational distribution, with a separate generator for each observable variable in the causal graph.

The advantage of this approach is that once a model is trained that faithfully reproduces the observational distribution on a given causal graph, any identifiable quantity will be identical in the trained model as in the original distribution, no matter the form of the underlying mechanisms and latent variables [27, Definition 3.2.4]. This means that so long as you have an i-instrument for X, one can use the trained generator to optimize for a conditional policy by directly implementing it in the model, as shown in Lem. 2.

In this experiment, we show that such an approach is, in fact, practical. To maintain the ability to compare our results with the ground truth, we use binary variables for each node in a tested graph. This restriction is loosened in the final two experiments - our goal here is to show that it is possible to get very accurate model reconstruction and imitation (with accurate intervention effects) with a GAN, even when there are latent variables.

Like in the previous experiment, for "ground-truth" data, our models are sampled uniformly from the space of factorized distributions. For example, for graph 1 in Table 1, one can factorize $P(x, y, z) = P(x)P(z|x)P(y|x, z)$, and can choose ground-truth probabilities $P(x), P(z|x'), P(y|x', z') \sim U(0, 1), \forall x, y, z, x', z'$.

For the GANs, we adapt discrete BGAN [12] to arbitrary SCM. The advantage of f-divergence-based approaches (such as BGAN) is that the f-GAN discriminators explicitly optimize a lower bound on the value of the chosen f-divergence. In particular, defining an f-divergence over two distributions P and Q as $D_f(P||Q) = \int_X q(x) f\left(\frac{p(x)}{q(x)}\right) dx$[6], with f^* as the convex conjugate of f, and using $\mathcal{T}$ as the set of functions that a neural network can implement, the f-GAN discriminator optimizes the following [25, 24]:

$$D_f(P||Q) \geq \sup_{T \in \mathcal{T}} \left(\mathbb{E}_{x \sim P}[T(x)] - \mathbb{E}_{x \sim Q}[f^*(T(x))]\right)$$

with a tight bound for $T^*(x) = f'\left(\frac{p(x)}{q(x)}\right)$. This means that by choosing an appropriate function f and the class of functions T, one can estimate the divergence through optimization over T [24]. With

[6]The KL divergence is an example of an f-divergence with $f(x) = x \log(x)$

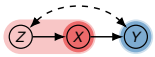
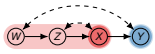
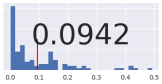
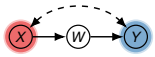
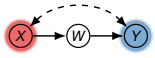
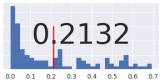
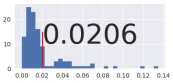
#	Graph	$\lvert y_{x_0} - \hat{y}_{x_0} \rvert$	$\lvert y_{x_1} - \hat{y}_{x_1} \rvert$	$\lvert \alpha - \hat{\alpha} \rvert$	$\lvert y - \hat{y} \rvert$
1		 0.0063	 0.0024	 0.0655	 0.0016
2		 0.0017	 0.0044	 0.0942	 0.0059
3		 0.0018	 0.0022	 0.0788	 0.0030
4		 0.0025	 0.0027	 0.2132	 0.0206

Table 1: Results of imitation using GANs in randomly sampled distributions. Each number is averaged over 100 randomly sampled models. Experiments in Row 3 and 4 are run with p-imitable, and not p-imitable distributions, respectively.

this trained discriminator $T(x)$, one can update generator Q_θ to be more similar to P with a update similar to a policy-gradient [12].

To adapt these GANs to work with causal models, we create a separate generator for each variable $V \in \boldsymbol{V}$, which is given as input samples from Pa_V, and outputs multinomial probabilities. Each latent variable is explicitly sampled from $\mathcal{N}(0,1)^k$ (with $k = 3$ in these experiments), and given as input to its children. Finally, instead of a single discriminator for the entire model, we exploit independence relations in the graph to localize optimization. In particular, in a Markovian causal model (without latent variables), one can create a separate discriminator for each variable, to directly learn the conditional distribution for each node. However, once there are latent variables, this variable-focused approach no longer captures all dependencies. Instead, we construct a separate conditional discriminator for each set of nodes that have a path made up of entirely bi-directed edges (c-components [40]), which allows each discriminator to specialize to a part of the model.

As an example, for the graph 1 in Table 1, the generator for X gets as input a sample $u \sim \mathcal{N}(0,1)^3$, and outputs a probability of 1. The generator for Z gets as input a sample according to the probability of X, and outputs a conditional probability of Z. Finally, Y gets as input u and the sampled value of Z. This graph has 2 c-components ($\{X, Y\}$ and $\{Z\}$), which corresponds to discriminators $P(z|x)$ and $P(x, y|z)$.

Similarly, once the model is optimized, the policy π is trained by manually replacing the generator for X with a new, untrained generator for π, and training it as a GAN with a discriminator comparing samples of Y in the original model (i.e., $Y \sim P(y)$) with samples of Y in the intervened model (i.e., $Y \sim P(y|\text{do}(\pi); \hat{M})$ where $\hat{M}$ is the parametrized model learned by generators.)

The graphs in Table 1 were each chosen to demonstrate a different aspect of imitability. The first graph is imitable by direct parents (Thm. 1), and corresponds to existing approaches to imitation, where an agent gets observations identical to the expert. The second graph demonstrates an example where an expert has additional information from a latent variable, but the agent can still successfully imitate $P(y)$ by using information that is not used by the expert, i.e., the i-backdoor admissible set $\{W\}$ (Thm. 2). Finally, Graphs 3 and 4 demonstrate data-dependent imitability (Def. 5). Graph 3 focuses on distributions that are amenable to imitation, meaning that imitation of $P(y)$ is possible without knowledge of the latent variable. Graph 4 uses only non-imitable distributions. Sampled uniformly, half of these instances are p-imitable, and half are not, so the expected performance of an unknown distribution is the average of Row 3 and 4.

The table columns show the average L1 distance between the computed interventional distributions, the predicted & optimal policy, as well as the effect $P(y|\text{do}(\pi))$ of the learned policy vs the observed $P(y)$. Each element of the table shows an average value of its corresponding distance from ground-truth over 100 runs (with each run sampling a different ground-truth probability distribution over the graph), overlaid over the histogram of distances over the 100 distributions/trained GANs.

The first two columns allow determining the error in reconstructing the interventional distribution of atomic intervention $\text{do}(x)$. The $|y_x - \hat{y}_x|$ represents the differences between the ground-truth $E[y|\text{do}(x)]$ and the imitated $E[y|\text{do}(x); \hat{M}]$ in the parametrized model $\hat{M}$. Notice that the policies are often conditioned, which is not reflected in these values.

The $|y - \hat{y}|$ value represents the difference in the ground-truth expert's reward $E[y]$ with the distribution $E[y|\text{do}(\pi)]$ induced by the learned policy π *in the ground-truth model*, meaning that the policy is trained in the imitated model, but is tested by replacing the true mechanism, as if the learned mechanism was tried in real life.

The main point of possible confusion could be $|\alpha - \hat{\alpha}|$ in the graphs where the policy is conditional. In the frontdoor cases, when the policy has no conditioning (Rows 3 and 4), the optimal value can be cleanly found, using Eq. (4). However, in the backdoor case (as seen in Row 1 and in Row 2), there can be multiple possible valid solutions. Here we show the precise procedure used to compute $|\alpha - \hat{\alpha}|$ in these two cases.

- In Row 1, which is commonly called the "backdoor" graph, we have defining $\alpha_1 = \pi(x_1|z_1)$ and $\alpha_0 = \pi(x_1|z_0)$:

$$\begin{aligned} P(y) &= \sum_{x,z} P(z)\pi(x|z)P(y|x,z) \\ &= P(z_1)\alpha_1 P(y|x_1,z_1) + P(z_1)(1-\alpha_1)P(y|x_0,z_1) \\ &\quad + P(z_0)\alpha_0 P(y|x_1,z_0) + P(z_0)(1-\alpha_0)P(y|x_0,z_0) \\ &= P(z_1)P(y|x_0,z_1) + P(z_0)P(y|x_0,z_0) \\ &\quad + \alpha_1 P(z_1)(P(y|x_1,z_1) - P(y|x_0,z_1)) \\ &\quad + \alpha_0 P(z_0)(P(y|x_1,z_0) - P(y|x_0,z_0)) \end{aligned} \tag{5}$$

  This means that multiple possible α_0, α_1 satisfy the given constraint. In such situations, given $\hat{\alpha}_1$ and $\hat{\alpha}_0$ found by GAN, we find the "closest correct comparison" by minimizing

$$P(z_0)|\hat{\alpha}_0 - \alpha_0| + P(z_1)|\hat{\alpha}_1 - \alpha_1|$$

  subject to the constraint in Eq. (5), and with $\alpha_1, \alpha_0 \in [0, 1]$. We then report this minimized value in the column labeled $|\alpha - \hat{\alpha}|$.
- Row 2 has more complex relations, since the policy has as inputs both values from Z and W. However, the values of W are irrelevant to imitating y, since W is in a different c-component of the graph than Y. This means that we can use the same approach as for Row 1, considering only Z (and averaging the policy over W values).

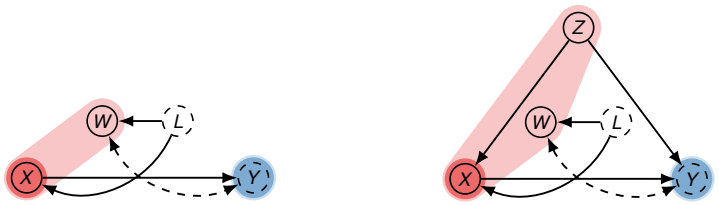
D.2.1 Discussion

The results in Table 1 suggest that GANs are capable of training accurate imitating policies in the presence of latent variables. Of particular note is the relatively large error in α can sometimes be present in the policies, despite the policies yielding very accurate samples of y. This happens when the imitable policy has $P(y|\text{do}(x_0))$ and $P(y|\text{do}(x_1))$ with very similar values, meaning that the policy has little effect on the probability of Y.

D.3 Highway Driving

The purpose of this experiment is to demonstrate that when imitating $P(y)$, it is important to choose a set of covariates that is i-backdoor admissible (Def. 4). To witness, in Fig. 7a, using knowledge of W to imitate X effectively adds confounding between X and Y, possibly affecting the imitated distribution of Y. That is, $\{W\}$ is not i-backdoor admissible. Similarly, in Fig. 7b, one must use knowledge of Z when imitating $P(y)$, but using either only W or both Z and W can lead to bias and inferior performance on reward measure Y. In other words, set $\{Z\}$ is i-backdoor admissible while $\{W, Z\}$ is not due to the active path $X \leftarrow L \rightarrow W \leftrightarrow Y$.

We demonstrate this in a two-step procedure. First, we performed an automated adversarial search over the binary distributions consistent with Fig. 7a, to *maximize* $\mathbb{E}[Y]$ when imitating just $\pi(x) = P(x)$, but *minimizing* $\mathbb{E}[Y]$ when exploiting knowledge of W to imitate $\pi(x|w) = P(x|w)$. This search led to the following mechanisms ($\oplus$ is xor):



(a) (b)

Figure 7: The graphs used for the backdoor experiments. First, Fig. 7a was optimized adversarially with binary variables to maximize naïve imitation error, then continuous data from the highD dataset were added in Fig. 7b, while maintaining the adversarial imitation error.

$$P(l_1) = P(u_1) = \frac{\sqrt{5}-1}{2} \approx 0.62 \quad W \leftarrow L \wedge U \quad X \leftarrow L \quad Y \leftarrow X \oplus U$$

This means that under "correct" imitation (not using knowledge of W), we have:

$$\mathbb{E}[Y|\text{do}(\pi_1)] = \sum_{x,u} \mathbb{E}[Y|x,u]\pi_1(x)P(u) = 2\sqrt{5} - 4 \approx 0.472$$

Under "incorrect" imitation (using W) we get:

$$\pi_2(x_1|w) = P(x_1|w) = \frac{\sum_{l,u} P(u)P(l)P(x_1|l)P(w|l,u)}{\sum_{l,u} P(l)P(u)P(w|l,u)} = \left\{ \begin{array}{r} 2\frac{\sqrt{5}-2}{\sqrt{5}-1} \approx 0.382 \text{ if } W = 0 \\ 1 \text{ if } W = 1 \end{array} \right\}$$

This means:

$$\mathbb{E}[Y|\text{do}(\pi_2)] = \sum_{x,l,u,w} \mathbb{E}[Y|x,u]\pi_2(x|w)P(w|l,u)P(l)P(u) = 4\frac{4\sqrt{5}-9}{\sqrt{5}-3} \approx 0.2918$$

Therefore, using the mechanisms above in Fig. 7a with binary variables, using W leads to a bias of 18%:

$$\mathbb{E}[Y|\text{do}(\pi_1)] - \mathbb{E}[Y|\text{do}(\pi_2)] = 2\sqrt{5} - 4 - 4\frac{4\sqrt{5}-9}{\sqrt{5}-3} \approx 0.18$$

Indeed, by sampling 10,000 data points, and using the empirical $P(x)$ for one "imitator", and $P(x|w)$ for the other, we get 0.1793 difference between the two, corroborating this result.

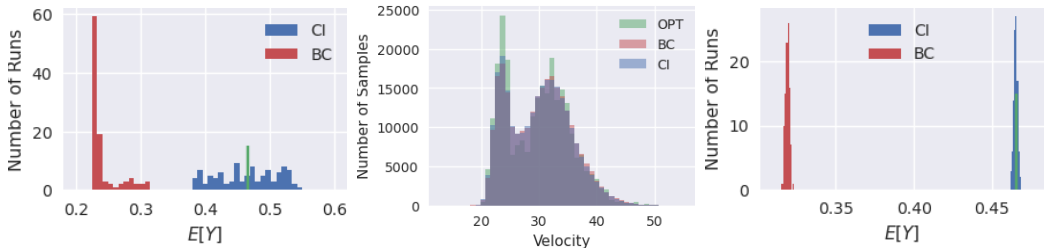
D.3.1 Making Things Continuous

We next show that this same issue can show up when variables are continuous. To achieve this, we adapted car velocity data from the `highD` dataset to a model similar to the binary version. In this model, one must use Z (velocity of the front car) to predict X (velocity of the driving car), but W would bias the prediction (with the same error as in previous section). The full model specification is as follows:

1. $P(L = 1) = P(U = 1) = 0.62$ (with values of L constructed from values of X)
2. $W \leftarrow L \wedge U$
3. Z is velocity of preceding car from `highD` dataset.
4. X is velocity of current car from `highD` dataset. L was constructed such that X satisfies the relation $L = I_{\{X-Z>-0.4\}}$ (this was achieved by choosing the threshold -0.4 to give the correct distribution over L).

5. $Y \leftarrow U \wedge I_{\{X-Z\leq -0.4\}} \vee \neg U \wedge I_{\{X-Z>-0.4\}}$

The values and mechanisms here were specifically chosen to have similar outputs to the previous model (Fig. 7a), despite using continuous car velocity data in place of boolean for node values. Fig. 7b shows a faithful graphical representation of the model. However, during the experiment, all algorithms are provided with only the causal diagram in Fig. 4a. That is, only independence relationships encoded in Fig. 4a are exploited.



(a) (b) (c)

Figure 8: Results of experiments using the continuous “backdoor" model. (a) and (c) are histograms of 100 independent runs of a direct supervised learning and GAN approach, respectively. (b) shows the distribution over X of a trained GAN model

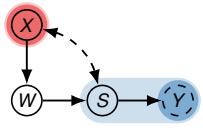
We performed two separate imitation experiments, both of which show that (1) using the covariates set $\{W, Z\}$ introduces biases into the imitator's reward $P(y|\text{do}(\pi))$; (2) using the i-backdoor admissible set $\{Z\}$ allows one to successfully imitate the expert's performance $P(y)$. The first experiment was imitation using standard supervised learning. Two separate (2 layer) neural networks were trained on a small subset of data using L2 loss over X. The first had as its input the values of Z (velocity of car ahead), while the second had as input values of W and Z. The trained outputs of these networks were used for imitation using the full dataset, and the resulting policy was evaluated by its induced $\mathbb{E}[Y|\text{do}(\pi)]$ (Y represents an unknown reward, so bigger is better). This experiment was repeated 100 times, giving a distribution over performance of trained models, shown in Fig. 8a. Note that while the performance varied widely, knowledge of W had a clear negative effect.

The outputs of a supervised learning algorithm are single values - they do not represent a distribution over possible values. This can affect imitation, since it is possible that the full range of the distribution is necessary for optimal imitation. We next trained two GANs to imitate the distribution over X, in the same way as done with the supervised models - one using only Z, and one using both W and Z. Due to the sampling needs of GANs, however, they had access to the full dataset over X, Z, W. This experiment was also repeated 100 times, giving the performance shown in Fig. 8c. Once again, the GAN using only Z has clearly superior performance. This difference exists despite both trained GANs seemingly recovering a good approximation over the distribution of X, as seen in Fig. 8b.

D.4 MNIST Digits

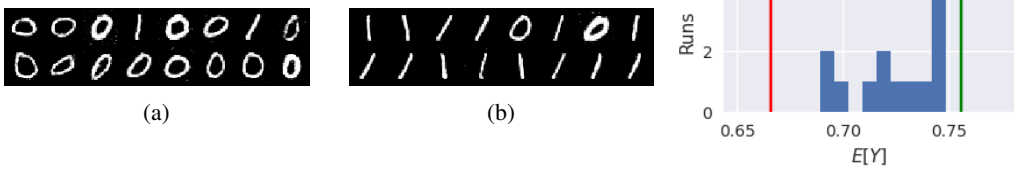
The purpose of this experiment is to show how Algorithm Alg. 1 using GANs for policy estimation could obtain reasonable imitation results in the p-imitable setting, even when the probability distribution of some observed endogenous variables is high-dimensional.

Specifically, we use the frontdoor graph, same as in the first experiment. There, when X and S are binary, we can compute the value α using Eq. (5). However, in this case, the formula for the interventional distribution is [28]:



$$P(s|\text{do}(\pi)) = \sum_x \pi(x) \sum_w P(w|x) \sum_{x'} P(s|x', w) P(x') \quad (6)$$

Critically, computing the effect of an intervention here requires a sum (or integral if continuous) over W. If W is a complex distribution or is high-dimensional, this quantity can be difficult to estimate.



(a) (b) (c)

Figure 10: In (a) and (b), results of a trained GAN conditional on $X = 0$ and $X = 1$ respectively (0 for FALSE, 1 for TRUE). In (c) is shown a histogram of the values obtained over multiple runs of the GAN. The red line represents naïve imitation, and the green line represents optimal imitation.

Once again, to allow us the ability to compare results to a ground-truth value, we repeat the procedure performed in the previous experiment, by first constructing a binary model with known characteristics, and then by replacing the binary value of W with a high-dimensional distribution with a property that follows the same underlying mechanism as the original binary value. Specifically, we use the `MNIST` digits for 0 and 1, which are a $28 \times 28 = 784$ dimensional vector representing a complex probability distribution. The pictures of 0s replace "FALSE" values of W, while pictures of 1s replace "TRUE" values. This allows a direct translation between a binary variable and the desired complex distribution. The underlying binary distribution had the following mechanisms:

$$
\begin{aligned}
P(u_1) &= 0.9 \\
P(x_1|u_0) = 0.1 \quad & P(x_1|u_1) = 0.9 \\
P(w_1|x_1) = 0.1 \quad & P(w_1|x_1) = 0.9 \\
P(s_1|u_0, w_0) = 0.1 \quad & P(s_1|u_0, w_1) = 0.9 \\
P(s_1|u_1, w_0) = 0.9 \quad & P(s_1|u_1, w_1) = 0.1
\end{aligned}
$$

This leads to $P(s_1) = 0.245$, but with naïve cloning of $P(x)$, the resulting $P(s_1|\text{do}(\pi)) = 0.334$, a difference of 0.0888. We chose the reward signal $Y \leftarrow \neg S$, since the assumption is that the original values of X were decided by an expert.

The GAN for the c-component $\{W\}$ is much larger than that of the other variables, since W is an image, so we pre-trained this component, and inserted the trained version into the full graph for optimization. We repeated experiment 16 times, of which 3 runs were discarded due to collapse in the GANs associated with $P(w|x)$. The results are visible in Fig. 10. These results show that despite the high-dimensional nature of the distribution over W, the GAN was consistently able to perform better than naïve imitation, approaching the optimal value.