TOWARDS CHARACTERIZING DOMAIN COUNTERFACTUALS FOR INVERTIBLE LATENT CAUSAL MODELS

A PREPRINT

Sean Kulinski*, **Zeyu Zhou***, **Ruqi Bai***, **Murat Kocaoglu, David I. Inouye**
Elmore Family School of Electrical and Computer Engineering
Purdue University
{skulinsk, zhou1059, bai116, mkocaogl, dinouye}@purdue.edu

June 21, 2023

ABSTRACT

Learning *latent* causal models from data has many important applications such as robustness, model extrapolation, and counterfactuals. Most prior theoretic work has focused on full causal discovery (i.e., recovering the true latent variables) but requires strong assumptions such as linearity [Seigal et al., 2022] or fails to have any analysis of the equivalence class of solutions (e.g., IRM [Arjovsky et al., 2019]). Instead of full causal discovery, we focus on a specific type of causal query called the *domain counterfactual*, which hypothesizes what a sample would have looked like if it had been generated in a different domain (or environment). Concretely, we assume domain-specific invertible latent structural causal models and a shared invertible observation function, both of which are less restrictive assumptions than prior theoretic works. Under these assumptions, we define *domain counterfactually equivalent* models and prove that any model can be transformed into an equivalent model via two invertible functions. This constructive property provides a tight characterization of the domain counterfactual equivalence classes. Building upon this result, we prove that every equivalence class contains a model where all intervened variables are at the end when topologically sorted by the causal DAG, i.e., all non-intervened variables have non-intervened ancestors. This surprising result suggests that an algorithm that only allows intervention in the last k latent variables may improve model estimation for counterfactuals. In experiments, we enforce the sparse intervention hypothesis via this theoretic result by constraining that the latent SCMs can only differ in the last few causal mechanisms and demonstrate the feasibility of this algorithm in simulated and image-based experiments.

1 Introduction

Causal reasoning has recently found many applications in machine learning from domain adaptation and generalization to fairness and explainability [Kusner et al., 2017, Moraffah et al., 2020]. The two fields that historically have evolved disconnected from each other started to merge with several recent results leveraging the available causal knowledge to develop better ML solutions. One such setting is when we have access to multiple datasets from different domains. From a causal perspective, each domain is generated via an intervention on some of the data-generating mechanisms. If one has access to or can recover this causal structure, it can be used to generate samples from interventional and counterfactual queries [Kocaoglu et al., 2018, Sauer and Geiger, 2021, Nemirovsky et al., 2022]. Most of these existing works, however, assume that the causal variables are observable.

In applications such as computer vision where data is composed of pixels, such an assumption is not realistic as the higher-level factors that generate the data are seldom observed [Koh et al., 2021]. Moreover, data from multiple domains are generated due to an intervention not on the observed variables in pixel space, but on these higher-level unobserved causal factors. Thus, causal methods that learn latent causal factors are important for these applications. While most of the existing causal discovery and inference algorithms are not applicable when the causal graph is

* Equal contribution.

completely unobserved, and we can only observe a projection of the variables, there has been some results on learning the latent structure from observed data under well-defined assumptions [Xie et al., 2023, Yang et al., 2022, Huang et al., 2022, Liu et al., 2022a,c,d, Xie et al., 2022, Chen et al., 2022]. However, many of these theoretic works make strong assumptions such as linearity. Indeed, these works suggest that full latent causal discovery may be infeasible without strong assumptions, which may not match real-world datasets. On the other side, other works such as invariant risk minimization (IRM) and its variants [Arjovsky et al., 2019, Ahuja et al., 2020, Rosenfeld et al., 2021] focus on a more narrow causally-inspired problem of model robustness while not making strong assumptions. IRM tries to bridge the gap between causal learning and ML robustness without claiming to discover latent causal factors.

In a similar spirit, this paper does not aim for full causal identifiability or discovery. Rather, we focus on a specific type of causal query we call a *domain counterfactual*, which hypothesizes what the sample would be if it had been generated in a different domain (or environment). For example, what would this medical image have looked like if it had been taken at hospital B even though it was taken at hospital A. Or, what would this photo of wheat look like if it had been taken in country B even though it was taken in country A? Or, what would be the properties of this material if it had been tested under high heat even though it was tested in low heat? This type of domain counterfactual query could have applications in explainability, knowledge discovery, and potentially model robustness. Yet, for latent causal models, this type of domain counterfactual query and the distributed equivalence classes of models for this query has not been analyzed or tightly characterized.

To make progress towards a better theoretic understanding of domain counterfactual queries, we start by defining a generic invertible latent domain causal model (ILD), which makes an invertibility assumption and a DAG assumption but otherwise is unconstrained. Given this ILD model, we define the notion of domain counterfactually equivalent models. Importantly, we provide an alternative necessary and sufficient property for models to be domain counterfactually equivalent that tightly characterizes the equivalence classes and provides a way to construct equivalent models. Building on this property, we prove that given any ILD model where k causal mechanisms are different (i.e., intervened) among domains, we can construct an equivalent latent "canonical" model where all k intervened mechanisms are the last k latent causal mechanisms. In combination with the sparse mechanism shift hypothesis (SMS) from Schölkopf et al. [2021] that assumes k is small, this theoretic result suggests a practical algorithm for finding counterfactually equivalent models by encouraging all domain differences to be concentrated in the last k variables. We propose practical methods for encouraging this constraint both from the generative direction via constraints on the generative model and the inference direction via distribution alignment. We demonstrate the feasibility of our approach on simulated datasets. We summarize our contributions as follows.

1. We prove a necessary and sufficient characterization of *domain* counterfactual equivalence.
2. We prove that given any model, we can construct a *canonical* model that is both distributionally and counterfactually equivalent to the given model. And we further show that all counter-factual equivalent and distributionally equivalent linear canonical model share the same sparsity k.
3. We propose theory-inspired methods to find canonical models from both the generative and inference directions and demonstrate the feasibility of these methods.

Notation We denote function equality between two functions $f : \mathcal{X} \to \mathcal{Y}$ and $f' : \mathcal{X} \to \mathcal{Y}$ as simply $f = f'$, which more formally can be stated as $\forall\, \boldsymbol{x} \in \mathcal{X}, f(\boldsymbol{x}) = f'(\boldsymbol{x})$. Similarly, $f \neq f'$ means $\exists\, \boldsymbol{x} \in \mathcal{X}, f(\boldsymbol{x}) \neq f'(\boldsymbol{x})$. We use $\circ$ to denote function composition, e.g., $g(f(\boldsymbol{x})) = g \circ f(\boldsymbol{x})$ or simply $h = g \circ f$. We denote $\mathcal{F}_I$ as the class of invertible (or bijective) functions and let Id denote the identity function. We denote $\mathcal{F}_A$ as the class of autoregressive functions, and $\mathcal{F}_{IA} = \mathcal{F}_I \cap \mathcal{F}_A$ as the invertible autoregressive functions. We use subscripts to denote particular indices (e.g., $x_j \in \mathbb{R}$ is the j-th value of the vector $\boldsymbol{x}$ and $\boldsymbol{x}_{<j} \in \mathbb{R}^{j-1}$ is the subvector corresponding to the indices 1 to $j-1$). For function outputs, we use bracket notation to select a single item (e.g., $[f(\boldsymbol{x})]_j \in \mathbb{R}$ refers to the j-th output of $f(\boldsymbol{x})$) or subvector (e.g., $[f(\boldsymbol{x})]_{\leq j} \in \mathbb{R}^j$ refers to the subvector for indices 1 to j inclusive). Similarly, for (unbound) functions, let $[f]_j : \mathbb{R}^m \to \mathbb{R}$ refer to the scalar function corresponding to the j-th output or $[f]_{\leq j} : \mathbb{R}^m \to \mathbb{R}^j$ refer to the function corresponding to first to j-th output. For any positive integer m, we define $[m] \triangleq \{1, \ldots, m\}$.. We denote N_d as number of domains in the ILD model.

2 Background

Structural Causal Models A structural causal model (SCM) considers m variables $\boldsymbol{x} \in \mathbb{R}^m$ in which each variable is a deterministic function of its parents and an independent exogenous noise variable ϵ, where the parents are defined by a corresponding directed acyclic graph (DAG). Formally, for all $j \in [m]$, $x_j \triangleq f'_j(\epsilon_j, \boldsymbol{x}_{\text{Pa}(j)})$, where $\epsilon_j \sim \mathcal{N}(0, 1)$. The deterministic function f'_j is called the *causal mechanism* of the j-th variable. If the variables are topographically

Table 1: This table of related causal representation learning works, focused mostly on works that study learning a *latent* SCM, shows that most prior works in this area aim for identifiability of the (latent) SCM, and thus require strong technical assumptions which may not hold in real-world scenarios (e.g., perfect single-node interventions for each variable). While a summary of the main assumptions for each work is listed, please see the references for more details.

	SCM	Observation Function	Other Assumptions	Observ. Function Identifiability	Characterization of Counterfactual Equivalence
Nasr-Esfahany et al. [2023b]	Invertible observed	N/A (Does not study latent SCM)	1) Access to ground-truth DAG	N/A	Single mechanism counterfactuals under specific contexts
Brehmer et al. [2022]	Invertible latent	Invertible	1) Atomic stochastic hard interv. per node 2) Training set is counterfactuals pairs 3) SCM is faithful DAG	Mixing and elementwise transform	N/A - Counterfactuals are input
Seigal et al. [2022]	Linear latent	Linear	1) Atomic hard interv.	Scaling	No
Liu et al. [2022b]	Linear latent	Non-linear	1) Significant causal weights variation	Mixing and scaling	No
Varici et al. [2023]	Latent non-linear	Linear	1) Atomic stochastic hard interv. 2) Each latent variable is intervented on	Mixing or scaling	No
Khemakhem et al. [2021]	Invertible observed (implicit)	Affine	1) Bivariate requirement for identifiability	Full (for bivariate case)	No
Ours	Invertible latent	Invertible	1) Access to domain labels 2) Sparse Mechanism Hypothesis	No	Domain counterfactual

sorted based on the DAG, then the SCM can be equivalently written as $x_j = \tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j}) \triangleq f'_j(\epsilon_j, \boldsymbol{x}_{\text{Pa}(j)})$, because $\boldsymbol{x}_{\text{Pa}(j)}$ is contained within $\boldsymbol{x}_{<j}$ by the topological property. Furthermore, because each x_j is dependent on ϵ_j, we can further write the SCM as $x_j = f_j(\epsilon_{\leq j}) \triangleq \tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j}(\epsilon_{<j}))$, because $\boldsymbol{x}_{<j}$ is a deterministic function of the $\epsilon_{<j}$. Thus we have the following different ways to write an SCM:

$$x_j = f'_j(\epsilon_j, \boldsymbol{x}_{\text{Pa}(j)}) \equiv \tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j}) \equiv f_j(\epsilon_{\leq j})\,. \tag{1}$$

This last viewpoint of SCMs motivates our autoregressive characterization of an SCM, where a multivariate autoregressive function is defined as follows.

Definition 1 (Autoregressive Function)**.** *A function $f : \mathcal{X} \to \mathcal{Y}$ is autoregressive, denoted $f \in \mathcal{F}_A$, if for all i, the i-th output can be written as a function of its corresponding input predecessors, i.e.,*

$$f \in \mathcal{F}_A \Leftrightarrow \forall\, i, \exists\, f_i \;\; s.t. \;\; [f(\boldsymbol{x})]_i \equiv f_i(\boldsymbol{x}_{\leq i})\,. \tag{2}$$

Thus, an autoregressive f with a distribution of exogeneous variables ϵ fully encodes an SCM, and we will use this in our definition of an invertible causal model in the next section. Usually, an SCM is defined on observed random variables $\boldsymbol{x}$ but we will define a latent SCM that encodes the causal structure among the latent variables $\boldsymbol{z}$ and an observation function that projects these latent causal factors into the observed space. We define each domain as a separate structural causal model (SCM). Interventions applied to an SCM generate another SCM, resulting in a new domain.

Assumption 1 (Hard and Soft Intervention)**.** *We consider two types of nondeterministic intervention: hard interventions, which completely remove the dependence of an intervened variable on its parents, and soft interventions (also called mechanism changes), which modify the dependence.*

Assumption 2 (Sparse Mechanism Shift (SMS))**.** *We assume the total number of changes, i.e. causal interventions, between domains is $k < m$.*

We adopt this hypothesis from Schölkopf et al. [2021], positing that distribution changes typically affect only a sparse or local subset of factors, rather than all factors simultaneously. For example, changing the latent variable corresponding to which hospital a patient visited for tumor imaging can lead to large changes in the observed images Kulinski and Inouye [2022].

3 Related Work

Causal reasoning has seen a surge of work regarding the combination of causal methods with machine learning methods. Some of this excitement is related to causal representation learning Schölkopf et al. [2021] and further latent

causal representation learning which aims to learn causal relations between latent variables which are then used to generate an observed distribution. This work falls under the latent causal representation learning category, and in Table 1, we make a direct comparison between our work and recent highly relevant related works. In this table, we aim to pay explicit attention to the assumptions required in each work.

Causal Representation Learning Causal representation learning is a rapidly developing field that aims to learn representations of data that are causally informative. This is in contrast to traditional representation learning, which does not consider the causal relationships between variables. For example, in Khemakhem et al. [2021] the authors use an autoregressive normalizing flow to implicitly estimate an SCM of observed variables. Given access to a ground-truth DAG, Nasr-Esfahany et al. [2023b] are able to recover the structural equation for a single causal mechanism under specific contexts such as access to an instrumental variable when there are unobserved confounders between exogenous noise variables. For more examples, please see the survey Schölkopf et al. [2021].

Latent Causal Representation Learning For latent causal representation learning, the general goal of these works is to identify the observational function that projects from the latent causal space to the observed space which can then be used to learn the representation of the latent causal space. As this is a highly difficult task, most works make assumptions on the problem structure such as [Seigal et al., 2022, Varici et al., 2023] who assume access to atomic hard interventions as well as the observation function being linear. Other works such as [Brehmer et al., 2022, Ahuja et al., 2022, Von Kügelgen et al., 2021] assume a weakly-supervised setting where one can train on counterfactual pairs $(x, \tilde{x})$ during training. In our work, we aim to maximize the practicality of our assumptions while still maintaining our theoretical goal of equivalent domain counterfactuals (as seen in Table 1).

Counterfactual Generation Counterfactual examples are answers to hypothetical queries such as "What would the outcome have been if we were in setting B instead of A?". Under the strict definition of counterfactual generation Pearl [2009], Ch. 7, this requires access to the underlying SCM. In this setting, to generate a counterfactual, one must take the three following steps: 1) *Abduction* which recovers the noise ϵ which generated the original observed event, 2) *Action* which makes the intervention in the SCM (e.g., applying a *do* operation), and 3) *Prediction* which generated the counterfactual using the intervened SCM. Works such as Nasr-Esfahany et al. [2023a], Shah et al. [2022,?] fall under this category of counterfactual generation; however, they do not consider the case of latent causal learning and thus are out of scope for this paper. There is a weaker form of counterfactual generation which does not use causal reasoning but instead uses generative models to generate counterfactuals Nemirovsky et al. [2022], Zhu et al. [2017]. These typically involve training a generative model which has a meaningful latent representation that can be intervened on to guide a counterfactual generation Ilse et al. [2020]. To constrain the generated sample to be "similar" to the original observed sample, some works apply a cost function between (input, output) pairs Zhu et al. [2017], Kulinski and Inouye [2022], Zhou et al. [2023]. As these works do not directly incorporate causal learning in their frameworks, we consider them out of scope for this paper.

4 Invertible Latent Domain Causal Model (ILD)

In this section, we aim to define a latent domain causal model that makes only weak assumptions yet can still be analyzed. Our main assumption is that both the latent SCMs for each domain and the observation function that projects the latent causal factors to the observed space are invertible. We first analyze the properties of an invertible SCM and then define our larger invertible latent domain causal model (ILD).

4.1 Invertible Structural Causal Model

Definition 2. *Given a causal DAG, an* invertible or bijective SCM *includes all SCMs such that the exogenous noise can be recovered from the SCM variables $\boldsymbol{x}$, i.e., the SCM defines a one-to-one mapping between ϵ and $\boldsymbol{x}$.*

While it may at first seem like we are limiting ourselves by only considering invertible SCMs, the following remark shows that this constraint does not reduce the expressivity of distributions.

Lemma 1 (Expressivity of Invertible SCM)**.** *Invertible SCMs can model any continuous distribution if the exogenous noise distribution is continuous.*

The full proof is in Appendix A.1 and leverages the invertible Rosenblatt transformation [Rosenblatt, 1952, Melchers and Beck, 2018, Chapter B] that can transform any distribution to the uniform distribution or vice versa for its inverse.

We now show that an invertible SCM (that is topologically sorted) is uniquely defined by a simple invertible autoregressive function that takes all exogenous noise variables simultaneously. This makes an important connection

between an autoregressive invertible function and the corresponding SCM it represents. In particular, this will be critical for analyzing the intervention set corresponding to the invertible SCM.

Lemma 2. *An invertible autoregressive SCM can be equivalently defined by the set of causal mechanisms* $\{\tilde{f}_j(\epsilon_j, \boldsymbol{z}_{<j})\}_{j=1}^m$ *or by a single invertible autoregressive function* $f \in \mathcal{F}_{IA}$*, and there is one-to-one mapping between these equivalent representations of an invertible SCM:*

$$f(\epsilon) = [\tilde{f}_1(\epsilon_1), \quad \tilde{f}_2(\epsilon_2, \underbrace{\tilde{f}_1(\epsilon_1)}_{\text{recover } x_1}), \quad \tilde{f}_3(\epsilon_3, \underbrace{\tilde{f}_1(\epsilon_1), \tilde{f}_2(\epsilon_2, \tilde{f}_1(\epsilon_1))}_{\text{recover } \boldsymbol{x}_{<3}}), \quad \cdots]^\top, \tag{3}$$

where for all j*,*

$$\tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j}) = [f(\underbrace{[f^{-1}(\boldsymbol{x}_{<j}, \cdot)]_{<j}}_{\text{recover } \epsilon_{<j} \text{ from } \boldsymbol{x}_{<j}}, \epsilon_j, \cdot)]_j \,. \tag{4}$$

See Appendix A.2 for proof. We can now define the intervention set between two invertible autoregressive functions in terms of their corresponding invertible SCM causal mechanisms.

Definition 3 (Intervention Set)**.** *The intervention set between* $f, f' \in \mathcal{F}_{IA}$ *is simply the soft intervened variables of the corresponding equivalent SCMs from* (4) *represented by the mechanisms* $\tilde{f}_j$ *and* $\tilde{f}'_j$ *respectively, i.e.,*

$$\mathcal{I}(f, f') \triangleq \mathcal{I}\big(\{\tilde{f}_j\}_{j=1}^m, \{\tilde{f}'_j\}_{j=1}^m\big) = \big\{j : \tilde{f}_j \neq \tilde{f}'_j\big\} \,. \tag{5}$$

Because we are only considering invertible SCMs, *do* interventions are not within scope because the intervened causal mechanism is not invertible (i.e., it is a constant). The interventions can change the dependencies on previous variables arbitrarily, i.e., the interventions could add or remove dependencies on possible parent variables (any predecessor variables). Thus, both soft and hard interventions are allowed and even interventions that increase the number of parent variables. The key requirement is that the causal mechanism changes. Furthermore, while the previous intervention set definition is sufficient, it is much simpler to analyze f or f^{-1} rather than the corresponding $\tilde{f}_j$ mechanisms so we prove in the following Lemma 3 that the intervention set can be determined by comparing the inverses of f (proof can be found in subsection A.3).

Lemma 3. *The intervention set* $f, f' \in$ *is equivalent to the set of variables where the inverse subfunctions are different, i.e.,* $\mathcal{I}(f, f') = \big\{j : \big[f^{-1}\big]_j \neq \big[f'^{-1}\big]_j\big\}$.

4.2 Invertible Latent Domain Causal Model (ILD)

Given the concepts from the previous section on invertible SCMs, we can now define our invertible latent domain causal model, which defines *latent* causal models f_d for each domain and a shared invertible observation function g.

Definition 4 (Invertible Latent Domain Causal Model)**.** *An* invertible latent domain causal model *(ILD), denoted* (g, f)*, makes the following assumptions:*

1. [Latent Invertible SCMs] *The latent domain-specific SCMs are invertible and represented by autoregressive invertible functions* $f \triangleq \{f_d(\epsilon) \in \mathcal{F}_{IA}\}_{d=1}^D$ *(see Lemma 2).*
2. [Invertible Observation Function]$g : \mathcal{Z} \to \mathcal{X} \in \mathcal{F}_I$ *is shared across domains.*
3. [Continuous Exogenous Noise] *The exogenous noise* ϵ *comes from a continuous distribution. Without loss of generality, we will assume* $\epsilon \sim \mathcal{N}(0, I)$.

Given these assumptions, the generative model for the d*-th domain is simply:* $\epsilon \sim \mathcal{N}(0, I)$, $\boldsymbol{z}|d = f_d(\epsilon)$, *and* $\boldsymbol{x} = g(\boldsymbol{z})$*. Because* f_d *and* g *are invertible, we can write the observed distribution using the change of variables formula as:* $p(\boldsymbol{x}|d) = p_{\mathcal{N}}\left(f_d^{-1} \circ g^{-1}(\boldsymbol{x})\right) |J_{f_d^{-1} \circ g^{-1}}(\boldsymbol{x})|$.

Discussion of ILD Assumptions ILD assumption 1 can be decomposed into the constraint on invertibility and autoregressiveness. The invertibility assumption does not restrict the SCMs expressivity as shown in Lemma 1. This assumption can be relaxed in practice using pseudo-invertible or approximately invertible functions, as seen with a VAE in subsection 6.2. The autoregressive assumption ensures that the invertible function properly represents a DAG causal graph. While it assumes a fixed ordering of variables, we note that there is no such restriction on g and thus g can absorb any reordering of the variables to match the autoregressive structure of f_d. Thus, in view of the observation function g, this autoregressive assumption does not reduce expressivity of this model class. ILD assumption 2 has two components: that g is (1) invertible and (2) shared across domains. Again invertibility does not hinder expressivity

similar to Lemma 1. The shared property will be critical for producing useful constraints on ILD but it does not inherently reduce expressivity as g could (in theory) just be the identity. As another example, suppose we have an ILD model (g, f) where g is the identity. We could construct other models (g', f') that produce the same distributions, where g' is an arbitrary invertible function and $f'_d = {g'}^{-1} \circ f_d$ (see Def. 5 for the formalization of distribution equivalence). In ILD assumption 3 we assume the exogenous noise distribution is standard Gaussian, which is made mostly for convenience and can be made without loss of generality using the invertible Rosenblatt transformation.

ILD Distribution Equivalence Compared to prior SCM works that operate in the observed space, the non-identifiability of ILD models is extenuated because we are considering *latent* SCMs. A natural *necessary* (though certainly not sufficient) condition for estimation is that the ILD matches the observed distributions, which in practice is implemented as minimizing a distribution divergence with respect to the observed samples. Thus, we now formally define the distribution equivalence relation for ILD.

Definition 5 (Distribution Equivalence)**.** *Two ILD models (g, f) and (g', f') are* distributionally equivalent*, denoted by $(g, f) \simeq_D (g', f')$, if the induced domain distributions are equal, i.e.,*

$$\forall\, d, \quad p_{\mathcal{N}}\left(f_d^{-1} \circ g^{-1}(\boldsymbol{x})\right) \left|J_{f_d^{-1} \circ g^{-1}}(\boldsymbol{x})\right| = p_{\mathcal{N}}\left({f'}_d^{-1} \circ {g'}^{-1}(\boldsymbol{x})\right) \left|J_{{f'}_d^{-1} \circ {g'}^{-1}}(\boldsymbol{x})\right| \tag{6}$$

The distributional equivalence property defines a true equivalence relation because (6) has the properties of reflexivity, symmetry, and transitivity by the properties of the equality of measures.

5 Analysis of ILD Domain Counterfactuals

In this section, our objective is not to identify a specific SCM but rather to explore any counterfactual equivalence SCM that aligns with the ground truth. Our study proceeds through the following steps: **1)** We introduce domain counterfactual equivalence relation between ILD models. **2)** We restrict the search space to the canonical space by demonstrating the existence of equivalent canonical ILD models maintaining intervention set size (cf. Theorem 2). **3)** We enhance the findings for linear SCMs, where we show that all canonical ILD models with linear SCMs equivalent to the original model share the intervention set size (cf. Theorem 4).

5.1 ILD Domain Counterfactuals

While distributional equivalence is a natural and common constraint for learning causal models, we now focus on our core contribution in the space of characterizing domain counterfactually equivalent models. We first provide a natural definition of this equivalence and prove that it is an equivalence relation. We proceed with briefly discussing the idea of a domain counterfactual.

The main idea of domain counterfactuals is that we can invert the causal model to retrieve the exogeneous noise variables from the observed variables and domain label and then push these exogeneous noise variables through the target domain SCM and the observation function. We formalize this for ILD models in the following definition.

Definition 6 (Domain Counterfactual)**.** *Given an ILD model (g, f), a counterfactual of $(\boldsymbol{x}, d)$ projected into the target domain d' can be written as:*

$$\boldsymbol{x}_{d \to d'} \triangleq g \circ f_{d'} \circ f_d^{-1} \circ g^{-1}(\boldsymbol{x})\,. \tag{7}$$

(7) can be interpreted as first projecting the sample into the latent space, i.e., g^{-1}, recovering the exogenous noise variables via f_d^{-1}, intervening by switching to the d' causal SCM $f_{d'}$ and then projectng back to the observed space via g. Given this notion of a domain counterfactual, we now provide an equivalence relation that will define which ILD models have the same domain counterfactuals (see Appendix B.1 for equivalence relation proof).

Definition 7 (Domain Counterfactual Equivalence)**.** *Two models (g, f) and (g', f') are* domain counterfactually equivalent*, denoted by $(g, f) \simeq_C (g', f')$, if all counterfactuals are equal, i.e., for all (d, d'), there holds*

$$g \circ f_{d'} \circ f_d^{-1} \circ g^{-1} = g' \circ f'_{d'} \circ {f'}_d^{-1} \circ {g'}^{-1}\,. \tag{8}$$

Lemma 4. *Domain counterfactually equivalent, denoted by $(g, f) \simeq_C (g', f')$ is an equivalence relation, i.e., the relation satisfies reflexivity, symmetry, and transitivity.*

5.2 ILD Domain Counterfactual Constructive Characterization

While Def. 7 succinctly defines the equivalence classes of models, it does not give much insight into the structure of the equivalence classes. To fill this gap in characterizing these domain counterfactual equivalence classes, we now present one of our main theoretic results. Namely, we prove that an alternative property is both *necessary and sufficient* to be counterfactually equivalent.

Theorem 1 (Characterization of Counterfactual Equivalence)**.** *Two ILD models are domain counterfactually equivalent, i.e.,* $(g, f) \simeq_C (g', f')$ *if and only if:*

$$\exists\, h_1, h_2 \in \mathcal{F}_I \text{ s.t. } g' = g \circ h_1^{-1} \in \mathcal{F}_I \text{ and } f'_d = h_1 \circ f_d \circ h_2 \in \mathcal{F}_A\,, \forall d. \tag{9}$$

See Appendix B.2 for proofs. While the *if* direction is relatively easy, the *only-if* direction is challenging. The proofs of Theorem 1 relies on recursively applying Lemma 6, where it provides an invertible composition equivalence. In particular, Lemma 6 ensures an existence of an intermediate invertible function which connects two pairs of equal compositions . Importantly, Theorem 1 can be used to *construct* counterfactually equivalent models and *verify* if a model is domain counterfactually equivalent (or determine they are not equivalent). More generally, this characterization exposes that the set of counterfactually equivalent models is actually very large. In fact, for *any* two invertible functions h_1 and h_2 that satisfy the implicit autoregressive constraint, i.e., for all $d, h_1 \circ f_d \circ h_2 \in \mathcal{F}_A$, we can construct a counterfactually equivalent model. In the next section, we demonstrate how to employ this novel characterization to establish a smaller set of domain counterfactual models, which we refer to as canonical domain counterfactuals.

5.3 Canonical Domain Counterfactual Models

We will now define the idea of a *canonical* ILD model that will allow each domain counterfactual equivalence class to be represented by a much smaller set of canonical models.

Definition 8 (Canonical Domain Counterfactual Model)**.** *An ILD model* (g, f) *is in* canonical counterfactual form*, denoted* $(g, f) \in \mathcal{C}$*, if the following two properties hold:*

1. [Identity Domain] *The SCM corresponding to one domain is the identity (which w.l.o.g. can be the first domain), i.e.,* $\exists\, d, f_d = \text{Id}$.
2. [Last Variables Intervened] *Only last variables are intervened, i.e.,* $j > m - |\mathcal{I}(f)|\,,\ \forall\, j \in \mathcal{I}(f)$.

The canonical form ILD's all the intervented nodes' descendant are also intervened nodes. All the unintervened nodes has no parents and follows standard gaussian distribution. While this definition may seem quite restrictive, in our next key result, we show that (surprisingly) *any* ILD model can be transformed to an equivalent *canonical* ILD model.

Theorem 2 (Existence of Equivalent Canonical ILD)**.** *Given an ILD model* (g, f)*, there exists a model* (g', f') *in canonical form that is both counterfactually and distributionally equivalent to* (g, f) *while maintaining the same size of the intervention set, i.e.,*

$$\forall (g, f), \exists\, (g', f') \in \mathcal{C} \text{ s.t. } (g', f') \simeq_C (g, f),\ (g', f') \simeq_D (g, f), \text{ and } |\mathcal{I}(f)| = |\mathcal{I}(f')|\,. \tag{10}$$

See Appendix subsection B.3 for full proof details. At high level, we use the constructive version of Theorem 1 to construct a series of counterfactually and distributionally equivalent models. In the first stage, we construct an ILD with $f_1 = \text{Id}$, canonical property 1 . In the second stage, we construct a sequence of ILDs that move the intervened variables to the end to enforce property 2 while maintaining 1. This second stage relies on a lemma that allows us to swap the position of latent random variables. We also provide an example to illustrate Theorem 2. See Example 1 for details. The practical implications of Theorem 2 extend to learning algorithms, as it implies that optimizing over only canonical ILDs is sufficient, i.e., it will not incur bias theoretically. Furthermore, we can easily prove the existence of a relaxed canonical model where the domain identity property is removed while maintaining the last variable property of canonical models.

Corollary 3 (Relaxed Canonical Existence)**.** *Given an ILD model* (g, f)*, there exists a model* (g', f') *that only satisfies the last variable property (Prop. 2) of Def. 8 that is both counterfactually and distributionally equivalent to* (g, f) *while maintaining the same size of the intervention set, i.e.,*

$$\forall (g, f), \exists\, (g', f') \in \mathcal{C} \text{ s.t. } (g', f') \simeq_C (g, f),\ (g', f') \simeq_D (g, f), \text{ and } |\mathcal{I}(f)| = |\mathcal{I}(f')|\,. \tag{11}$$

We omit the proof, it directly follows the proof of Theorem 2 where we apply one more step similar to step 1. The only difference is that we choose $h_1 = f_1$ and $h_2 = \text{Id}$ instead.

5.4 Stronger Result for Linear SCMs

In the previous section, we proved that an equivalent canonical model exists with the same sparsity as the original model. We now prove a stronger result for the case of linear latent SCMs (i.e., f_d is linear) that says all canonical ILDs with linear SCMs equivalent to the original model not only have the same sparsity but also the intervened variables. This has practical implications for algorithm design as it suggests that we should use the smallest sparsity level possible (to reduce estimation variance) while being above the true sparsity level (to avoid estimation bias). Consequently, in terms of sharing the same intervention set, counterfactually and distributional equivalent canonical ILD for Linear SCM is unique. (See Appendix B.4 for proofs.)

Theorem 4 (Canonical ILD with Linear SCM and Shared Intervention Sparsity)**.** *Given an ILD model (g, f) where f is a* linear*, all canonical models that are distributionally and counterfactually equivalent to (g, f) also have the same sparsity, i.e.,*

$$\forall (g', f') \in \left\{ (\tilde{g}, \tilde{f}) \in \mathcal{C} : (\tilde{g}, \tilde{f}) \simeq_D (g, f), (\tilde{g}, \tilde{f}) \simeq_C (g, f) \right\}, \quad \mathcal{I}(f) = \mathcal{I}(f') \,. \tag{12}$$

Discussion of Results It is important to note that the canonical ILD models we have discussed do not guarantee the recovery of the true causal model. Rather, their purpose is to assist in improving counterfactual recovery. Furthermore, it should be emphasized that these models are not identifiable. However, for non-linear models, we can narrow down the set of potential models to those with the same size of intervention set, leading to improved performance of estimation algorithms on average. In contrast, for linear models, we can successfully find the counterfactual and distributionally equivalent models that share the same intervention set.

6 Experiments

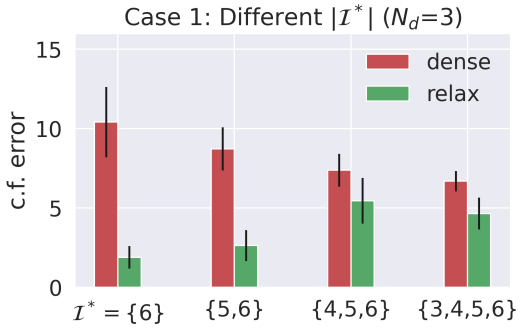
In this section, we aim to use our canonical ILD model to learn to produce domain counterfactuals given only observed data $\boldsymbol{x}$ and domain labels d. We first study this in a suite of simulated experiments in which we seek to learn a ground truth latent SCM under different parameterization settings. We then look at relaxing the invertibility constraint of our models to only require pseudoinvertibility (e.g., an autoencoder structure) and apply this relaxed ILD form to generate image counterfactuals with Rotated MNIST Deng [2012]. For both experiments, we only train our model to fit the observed data distribution. Details of the training algorithms and settings can be found in Appendix D.1 and Appendix E.1.

6.1 Simulated Dataset

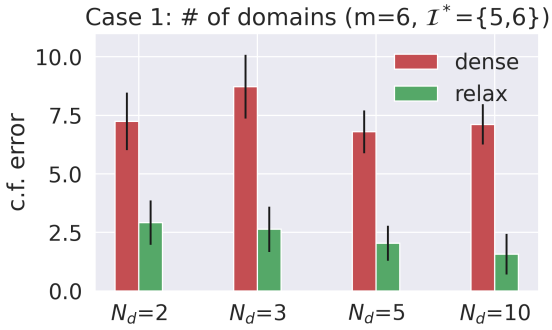
Here, we study the effects of two types of misspecification. First, the intervention set of the ground truth model, f^* might not be limited to the last few nodes. Theorem 2 proves the existence of the counterfactually and distributionally equivalent model and here we investigate how hard it would be to find a correct canonical model. Second, we might not have the knowledge of the correct size of the intervention set, and thus we investigate whether our algorithm is sensitive to its sparsity setting. Let $\mathcal{I}$ and $\mathcal{I}^*$ represent the intervention set of the model and ground truth model, respectively; we test our models with simulated datasets in 3 cases: (1) No model mispecfication: $|\mathcal{I}| = |\mathcal{I}^*|$ and $\mathcal{I}^*$ ony contains the last few nodes. (2) Intervention indices mismatch: $|\mathcal{I}| = |\mathcal{I}^*|$ but $\mathcal{I}^*$ does not only contain last few nodes. (3) Intervention set size mismatch: $|\mathcal{I}| \neq |\mathcal{I}^*|$.

We include details about dataset generation in Appendix D.1 and we use normalizing flows with structure similar to the ground truth to train two ILD models: *ILD-Relax-Can* which represents the relaxed canonical ILD form from Cor. 3 and a baseline model, *ILD-Dense*, which has no sparsity restrictions on its latent SCM. More details and illustrating figures of the models can be found in Appendix D.1. To evaluate the models, we compute the mean square error between the estimated counterfactual and ground truth counterfactual. As in practice, we can only check alignment performance instead of counterfactual estimation, we report the error computed with the test dataset when the likelihood computed with the validation set is highest.

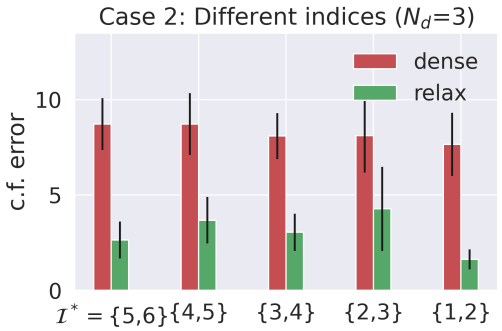
Case 1: No model misspecification We start with the simplest setup where the ground truth intervention set only contains the last few nodes and we have oracle knowledge of the size of the intervention set. In Figure 1a, we investigate the effect of the intervention set size and observe that the performance gap tends to get larger when the true intervention set gets more sparse. Besides, even when the ground truth model is relatively dense (when $|\mathcal{I}^*|$ is close to m), *ILD-Relax-Can* still outperforms *ILD-Dense*. Then we test the performance of *ILD-Dense* and *ILD-Relax-Can* with different number of domains. As shown in Figure 1b, our model performs consistently better than *ILD-Dense* with different number of domains. A more thorough investigation could be found in Appendix D.2.



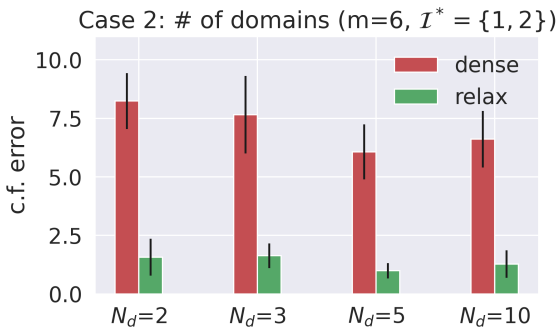
(a) Case 1: Different $|\mathcal{I}^*|$



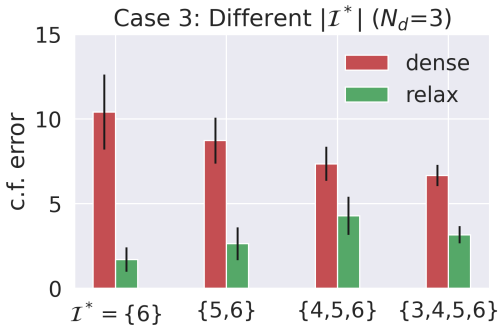
(b) Case 1: Number of domains



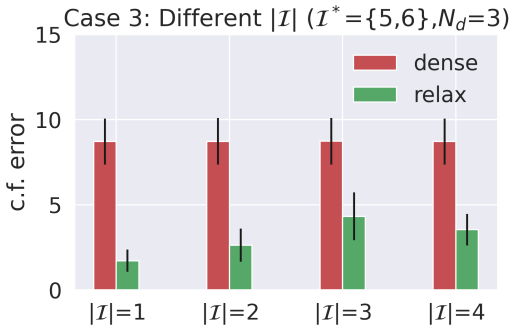
(c) Case 2: Different indices



(d) Case 2: Number of domains



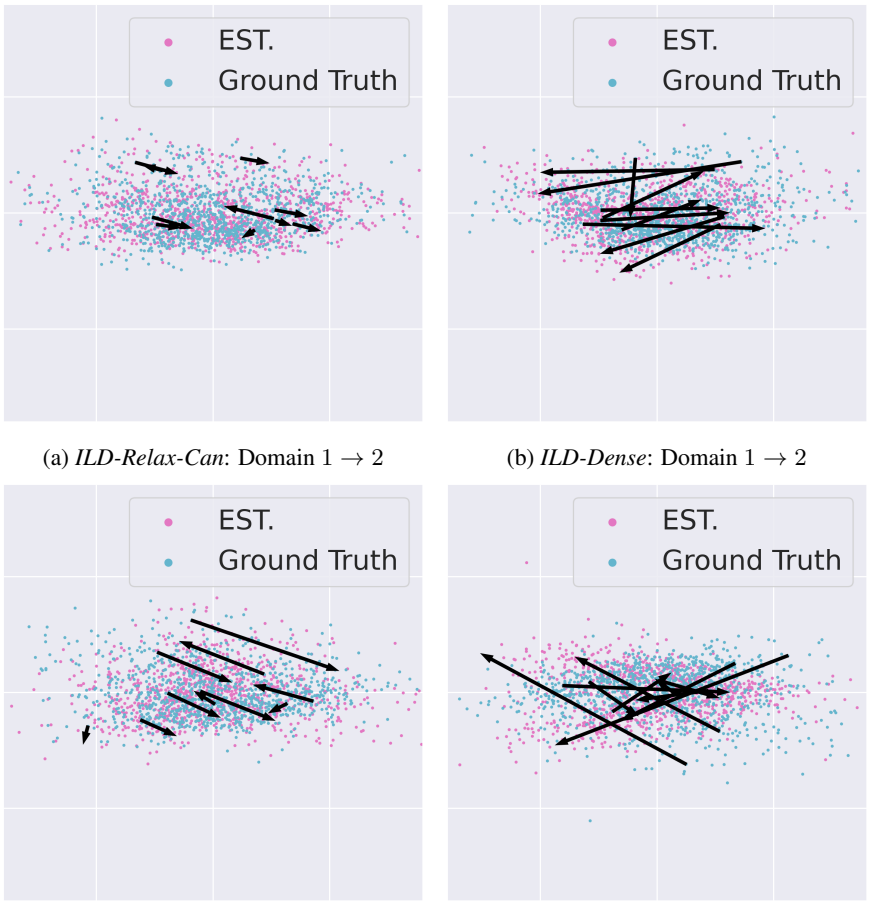
(e) Case 3: Different $|\mathcal{I}^*|$ with fixed $\mathcal{I}$



(f) Case 3: Different $|\mathcal{I}|$ with fixed $\mathcal{I}^*$

Figure 1: Results with simulated dataset. Counterfactual error is averaged over 10 runs with different seeds and the error bar represents the standard error. All results are with $m = 6$. (a) Case 1: Here we investigate varying the size of the true intervention set and observe that *ILD-Relax-Can* is better than *ILD-Dense* in all cases. Also, the gap tends to largest when $|\mathcal{I}^*|$ is smallest, which is in accordance with the fact that *ILD-Relax-Can* is a subset of *ILD-Dense* and they are equivalent when $|\mathcal{I}^*| = m$. (b) Case 1: This shows that *ILD-Relax-Can* outperforms *ILD-Dense* across varying numbers of domains. (c) Case 2: Here we investigate varying the indices of the intervened nodes, which shows *ILD-Relax-Can* is consistently better than *ILD-Dense* regardless of which nodes are intervened. (d) Case 2: This changes the number of domains and shows *ILD-Relax-Can* scales well with the number of domains. (e) Case 3: Here we test varying $|\mathcal{I}^*|$ while holding $\mathcal{I}$ fixed. We can see the performance gap becomes smaller but *ILD-Relax-Can* still performs better in all cases. (f) Case 3: This conversely sweeps $|\mathcal{I}|$ while holding $\mathcal{I}^*$ fixed. Similarly, the performance of *ILD-Relax-Can* approaches to that of *ILD-Dense* as we increase $|\mathcal{I}|$. An unexpected result is that *ILD-Relax-Can* performs best when $|\mathcal{I}| = 1$ and that results from a worse data fitting which is more carefully investigated in Appendix D.2.

Case 2: Intervention indices mismatch Here we investigate a more practical scenario where we have knowledge of $|\mathcal{I}^*|$ but the intervened nodes are not the last few nodes. To investigate the effect of different indices of the intervened



(a) *ILD-Relax-Can*: Domain $1 \to 2$ (b) *ILD-Dense*: Domain $1 \to 2$

(c) *ILD-Relax-Can*: Domain $3 \to 1$ (d) *ILD-Dense*: Domain $3 \to 1$

Figure 2: Visualization of counterfactual error when $m = 6, N_d = 3, |\mathcal{I}| = 2, \mathcal{I}^* = \{1, 2\}$. In each plot, we find the first two principle components and project the data along that direction. We select 10 points, then find the corresponding ground truth counterfactual and estimated counterfactual. The black arrow points from ground truth to estimated counterfactual.

nodes, in Figure 1c, we change the true intervention set $\mathcal{I}^*$ while keeping the number of intervened nodes $|\mathcal{I}^*|$ the same. We observe that *ILD-Relax-Can* outperforms *ILD-Dense* and performs consistently well in all cases. It provides evidence that as long as we know the correct $|\mathcal{I}^*|$, it is not hard to find a counterfactually and distributionally equivalent canonical model regardless which nodes are intervened on. In Figure 1d, we test with different number of domains and it shows that our algorithm scales well with the number of domains, which is similar to what we observe in case 1. In Figure 2, we visualize how *ILD-Relax-Can* leads to a lower counterfactual error in comparison to *ILD-Dense*. As shown in Figure 2a and Figure 2b, *ILD-Relax-Can* clearly does better in counterfactual estimation. In Figure 2c and Figure 2d, both of them have a relatively larger error. However, *ILD-Relax-Can* tends to find a closer solution while *ILD-Dense* matches distribution more randomly. This could result from the large search space of *ILD-Dense* and it can easily encodes a transformation such as rotation which will not hurt distribution fitting but will lead to a significant counterfactual error. In summary, we show that even without the knowledge of the specific nodes being intervened on, sparse constraint improves the performance in counterfactual estimation. More results could be found in Appendix D.2.

Case 3: Intervention set size mismatch Here we investigate what if we have no knowledge of the dataset. To check the effect of mismatch of the number of intervened nodes between the true model and the approximation, we first change $\mathcal{I}^*$ while keeping the model unchanged, i.e., $|\mathcal{I}| \neq |\mathcal{I}^*|$ and $\mathcal{I}$ is fixed. In Figure 1e, the performance gap between *ILD-Relax-Can* and *ILD-Dense* become smaller as the dataset becomes less sparse. However, *ILD-Relax-Can* still outperforms *ILD-Dense* in all cases. In the meanwhile, we notice that the data fitting performance of *ILD-Relax-Can* is obviously worse than *ILD-Dense*. This is as expected because there is model misspecification and our theorem does not guarantee the existence of an equivalent canonical model in this case. In Figure 1f, we change $|\mathcal{I}|$ while

keeping the dataset unchanged. Similarly, even though when $|\mathcal{I}| = 1$, the counterfactual error is lowest for *ILD-Relax-Can*, we observe a significant drop in the performance of data fitting. More results about data fitting performance and experiments with different setups could be found in Appendix D.2.

6.2 Rotated MNIST

In this section, we seek to learn domain counterfactuals in a high-dimensional regime such as images. To this end, we use the Rotated MNIST dataset, a common distribution shift dataset which generates pseudo-realistic domain shifts by applying domain-specific rotations to the MNIST digit dataset Deng [2012].

To allow for learning a complex mapping between the observed space and the latent space, we relax the invertibility requirement of the ILD models to allow for pseudoinvertibility. This relaxation allows us to modify the ILD models from Section 6.1 to fit a VAE Kingma and Welling [2013] structure where the variational encoder consists of $g \circ f_d$ and the decoder consists of $f_d^+ \circ g^+$. A detailed description and diagram of the models can be found in Fig. 15, but informally, these modified ILD models can be seen as training a VAE *per* domain with the restriction that each VAE shares parameters for its initial encoder and final decoder layers (i.e. g is shared). As an additional baseline, we compare against the naive setup, which we call *ILD-Independent*, where each VAE has no shared parameters (i.e. a custom g is learned for each domain). These models were trained using the β-VAE framework Higgins et al. [2017] with $\beta = 1,000$ (similar to the setting in Burgess et al. [2018]) to fit the observed data distribution via reconstruction error and align the latent Gaussian prior via the KL divergence loss, further details can be found in the Appendix E.1. After training, we can perform domain counterfactuals as described in Eqn. 7. Examples of domain counterfactuals can be seen in Fig. 3 where while all four models correctly capture the rotation for each domain, only the canonical models (especially *ILD-Relax-Can*) tend to produce counterfactuals that are similar to the original sample (e.g., the digit label stays the same). We note that no digit label information was seen during training, thus suggesting the intervention sparsity allowed the canonical models to preserve important non-domain-specific causal information. This is further corroborated in Table 2 which shows sparse ILD models having significantly lower ground truth counterfactual error than their non-sparse counterparts.

Table 2: MSE Loss between ground truth counterfactual and estimated counterfactual for RMNIST counterfactuals.

Latent-dim	*ILD-Independent*	*ILD-Dense*	*ILD-Can*	*ILD-Relax-Can*
10	0.0267 ± 0.0003	0.0287 ± 0.0018	0.0250 ± 0.0002	$\mathbf{0.0240 \pm 0.0016}$
20	0.0275 ± 0.0003	0.0250 ± 0.0004	0.0238 ± 0.0002	$\mathbf{0.0225 \pm 0.0003}$

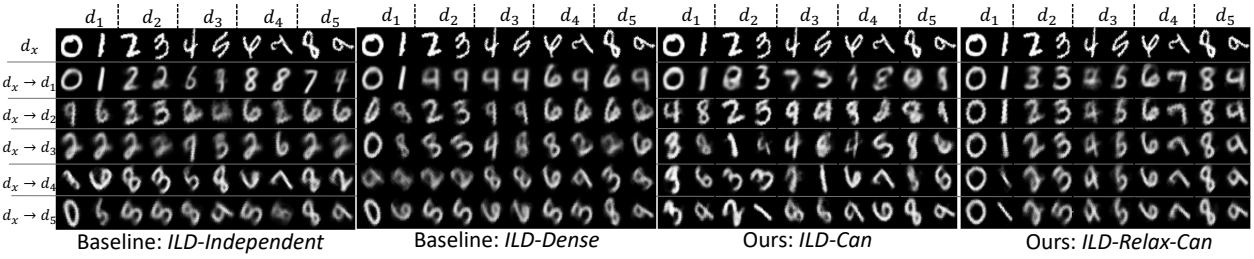


Figure 3: Counterfactual plots for the four relaxed ILD models, where across the columns we show examples of two digits from each domain and each row corresponds to the counterfactual to a different RMNIST domain. It can be seen that while all four models correctly recover the rotation for each domain counterfactual, the baseline models usually change the digit label during counterfactual, while *ILD-Relax-Can* tends to preserve the digit label, despite not being privy to any label information during training.

7 Conclusion

In this paper, we prove a necessary and sufficient characterization of domain counterfactual equivalence with more practical assumptions in comparison to existing works. We derive a model which is contained in this equivalence class where all intervened nodes variables are at the end. Then we empirically validate that algorithms inspired by our theory lead to better counterfactual estimation with extensive simulated experiments and in a high dimensional case using Rotated MNIST. We hope our theory could give inspiration to the design of practical algorithms which bridge the gap between causal reasoning and machine learning.

References

Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *International Conference on Machine Learning*, pages 145–155. PMLR, 2020.

Kartik Ahuja, Jason S Hartford, and Yoshua Bengio. Weakly supervised representation learning with sparse perturbations. *Advances in Neural Information Processing Systems*, 35:15516–15528, 2022.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Johann Brehmer, Pim De Haan, Phillip Lippe, and Taco S Cohen. Weakly supervised causal representation learning. *Advances in Neural Information Processing Systems*, 35:38319–38331, 2022.

Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in $beta$-vae. *arXiv preprint arXiv:1804.03599*, 2018.

Zhengming Chen, Feng Xie, Jie Qiao, Zhifeng Hao, Kun Zhang, and Ruichu Cai. Identification of linear latent variable model with arbitrary distribution. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 6350–6357. AAAI Press, 2022. URL https://ojs.aaai.org/index.php/AAAI/article/view/20585.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2017.

Biwei Huang, Charles Jia Han Low, Feng Xie, Clark Glymour, and Kun Zhang. Latent hierarchical causal structure discovery with rank constraints. *CoRR*, abs/2210.01798, 2022. doi: 10.48550/arXiv.2210.01798. URL https://doi.org/10.48550/arXiv.2210.01798.

Maximilian Ilse, Jakub M Tomczak, Christos Louizos, and Max Welling. Diva: Domain invariant variational autoencoders. In *Medical Imaging with Deep Learning*, pages 322–348. PMLR, 2020.

Ilyes Khemakhem, Ricardo Monti, Robert Leech, and Aapo Hyvarinen. Causal autoregressive flows. In *International conference on artificial intelligence and statistics*, pages 3520–3528. PMLR, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Murat Kocaoglu, Christopher Snyder, Alexandros G. Dimakis, and Sriram Vishwanath. Causalgan: Learning causal implicit generative models with adversarial training. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=BJE-4xW0W.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts, 2021.

Sean Kulinski and David I Inouye. Towards explaining distribution shifts. *arXiv preprint arXiv:2210.10275*, 2022.

Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. *Advances in neural information processing systems*, 30, 2017.

Yuhang Liu, Zhen Zhang, Dong Gong, Mingming Gong, Biwei Huang, Anton van den Hengel, Kun Zhang, and Javen Qinfeng Shi. Identifying weight-variant latent causal models, 2022a. URL https://arxiv.org/abs/2208.14153.

Yuhang Liu, Zhen Zhang, Dong Gong, Mingming Gong, Biwei Huang, Anton van den Hengel, Kun Zhang, and Javen Qinfeng Shi. Identifying weight-variant latent causal models. *arXiv preprint arXiv:2208.14153*, 2022b.

Yuhang Liu, Zhen Zhang, Dong Gong, Mingming Gong, Biwei Huang, Anton van den Hengel, Kun Zhang, and Javen Qinfeng Shi. Weight-variant latent causal models. *CoRR*, abs/2208.14153, 2022c. doi: 10.48550/arXiv.2208.14153. URL https://doi.org/10.48550/arXiv.2208.14153.

Yuhang Liu, Zhen Zhang, Dong Gong, Mingming Gong, Biwei Huang, Kun Zhang, and Javen Qinfeng Shi. Identifying latent causal content for multi-source domain adaptation. *CoRR*, abs/2208.14161, 2022d. doi: 10.48550/arXiv.2208.14161. URL https://doi.org/10.48550/arXiv.2208.14161.

Robert E Melchers and André T Beck. *Structural reliability analysis and prediction*. John wiley & sons, 2018.

Raha Moraffah, Mansooreh Karami, Ruocheng Guo, Adrienne Raglin, and Huan Liu. Causal interpretability for machine learning-problems, methods and evaluation. *ACM SIGKDD Explorations Newsletter*, 22(1):18–33, 2020.

Arash Nasr-Esfahany, Mohammad Alizadeh, and Devavrat Shah. Counterfactual identifiability of bijective causal models. *arXiv preprint arXiv:2302.02228*, 2023a.

Arash Nasr-Esfahany, Mohammad Alizadeh, and Devavrat Shah. Counterfactual identifiability of bijective causal models, 2023b.

Daniel Nemirovsky, Nicolas Thiebaut, Ye Xu, and Abhishek Gupta. Countergan: Generating counterfactuals for real-time recourse and interpretability using residual gans. In *Uncertainty in Artificial Intelligence*, pages 1488–1497. PMLR, 2022.

Judea Pearl. Causal inference in statistics: An overview. 2009.

Murray Rosenblatt. Remarks on a multivariate transformation. *The annals of mathematical statistics*, 23(3):470–472, 1952.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. In *International Conference on Learning Representations*, volume 9, 2021.

Axel Sauer and Andreas Geiger. Counterfactual generative networks. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=BXewfAYMmJw.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

Anna Seigal, Chandler Squires, and Caroline Uhler. Linear causal disentanglement via interventions. *arXiv preprint arXiv:2211.16467*, 2022.

Abhin Shah, Raaz Dwivedi, Devavrat Shah, and Gregory W Wornell. On counterfactual inference with unobserved confounding. *arXiv preprint arXiv:2211.08209*, 2022.

Burak Varici, Emre Acarturk, Karthikeyan Shanmugam, Abhishek Kumar, and Ali Tajer. Score-based causal representation learning with interventions. *arXiv preprint arXiv:2301.08230*, 2023.

Julius Von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. *Advances in neural information processing systems*, 34:16451–16467, 2021.

Feng Xie, Biwei Huang, Zhengming Chen, Yangbo He, Zhi Geng, and Kun Zhang. Identification of linear non-gaussian latent hierarchical structure. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 24370–24387. PMLR, 2022. URL https://proceedings.mlr.press/v162/xie22a.html.

Feng Xie, Yan Zeng, Zhengming Chen, Yangbo He, Zhi Geng, and Kun Zhang. Causal discovery of 1-factor measurement models in linear latent variable models with arbitrary noise distributions. *Neurocomputing*, 526:48–61, 2023. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2023.01.034. URL https://www.sciencedirect.com/science/article/pii/S0925231223000449.

Yuqin Yang, AmirEmad Ghassami, Mohamed S. Nafea, Negar Kiyavash, Kun Zhang, and Ilya Shpitser. Causal discovery in linear latent variable models subject to measurement error. *CoRR*, abs/2211.03984, 2022. doi: 10.48550/arXiv.2211.03984. URL https://doi.org/10.48550/arXiv.2211.03984.

Zeyu Zhou, Sheikh Shams Azam, Christopher Brinton, and David I Inouye. Efficient federated domain translation. In *The Eleventh International Conference on Learning Representations, ICLR*, 2023.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

Appendix

Table of Contents

A Auxiliary results and proofs of Section 4

In this section, we prove the lemmas in Section 4 which capture important properties of the ILD Model. Before proving Lemma 1, we first introduce another lemma that is useful later in proving Lemma 2.

Lemma 5 (Invertible Upper Subfunctions)**.** *The upper subfunctions of $f \in \mathcal{F}_I \cap \mathcal{F}_A$ are also invertible, i.e., $\bar{f}_j(\epsilon_{\leq j}) \triangleq [f(\epsilon_{\leq j}, \cdot)]_{\leq j}$ is an invertible function of $\epsilon_{\leq j}$.*

Proof. We will prove this by induction on k where $j = m - k$. For $k = 0$, it is trivial because $\bar{f}_{\leq m} \equiv f \in \mathcal{F}_I$. We will prove the inductive step by contradiction. Suppose $\bar{f}_{\leq m-k}$ is not invertible. This would mean it is not injective and/or not surjective.

If $\bar{f}_j$ is not injective, then $\exists \epsilon_{\leq j} \neq \epsilon'_{\leq j}$ such that $\bar{f}_{\leq j}(\epsilon_{\leq j}) = \bar{f}_{\leq j}(\epsilon'_{\leq j})$. We would then have for some $\epsilon_{>j}$ (e.g., all zeros):

$$\bar{f}_{\leq j+1}(\epsilon_{\leq j}, \epsilon_{j+1}) = [\bar{f}_{\leq j}(\epsilon_{\leq j}), [f(\epsilon_{\leq j}, \epsilon_{>j})]_{j+1}]^\top = [\bar{f}_{\leq j}(\epsilon'_{\leq j}), [f(\epsilon_{\leq j}, \epsilon_{>j})]_{j+1}]^\top = \bar{f}_{\leq j+1}(\epsilon'_{\leq j}, \epsilon_{j+1}) \,, \quad (13)$$

but this would contradict the fact that $\bar{f}_{\leq j+1}$ is invertible by the inductive hypothesis.

If $\bar{f}_{\leq j}$ is not surjective, then $\exists\ \boldsymbol{x}_{\leq j}$ such that $\forall \epsilon_{\leq j}, \bar{f}_{\leq j}(\epsilon_{\leq j}) \neq \boldsymbol{x}_{\leq j}$. We would then have that $\forall \epsilon_{\leq j}, \epsilon_{>j}$

$$\bar{f}_{j+1}(\epsilon_{\leq j}, \epsilon_{j+1}) = [\bar{f}_j(\epsilon_{\leq j}), [f(\epsilon_{>j})]_{j+1}]^\top \neq [\boldsymbol{x}_{\leq j}, x_{j+1}]^\top \,. \quad (14)$$

but this would contradict the fact inductive hypothesis that $\bar{f}_{j+1}$ is surjective. Therefore, $\bar{f}_j$ must be invertible for all $j \in [m]$. □

A.1 Proof of Lemma 1

The proof leverages the invertible Rosenblatt transformation [Rosenblatt, 1952, Melchers and Beck, 2018, Chapter B] that can transform any distribution to the uniform distribution or vice versa using its inverse. Given an ordering of a set of random variables, i.e., $\boldsymbol{x} = [x_1, x_2, \cdots, x_m]^\top$, the Rosenblatt transformation is defined as follows:

$$\begin{aligned} u_1 &:= F_1(x_1) \\ u_2 &:= F_2(x_2|x_1) \\ u_3 &:= F_3(x_3|x_1, x_2) \\ &\vdots \\ u_m &:= F_m(x_m|x_1, x_2, \cdots, x_{m-1}) \,, \end{aligned} \tag{15}$$

where $F_j(x_j|\boldsymbol{x}_{<j})$ is the conditional CDF of x_j given $\boldsymbol{x}_j$, i.e., the CDF corresponding to the distribution $p(x_j|\boldsymbol{x}_{<j})$. It's inverse can be written as follows:

$$\begin{aligned} x_1 &= F_1^{-1}(u_1) \\ x_2 &= F_2^{-1}(u_2|x_1) \\ x_3 &= F_3^{-1}(u_3|x_1, x_2) \\ &\vdots \\ x_m &= F_m^{-1}(u_m|x_1, x_2, \cdots, x_{m-1}) \,, \end{aligned} \tag{16}$$

where $F_j^{-1}(u_j|\boldsymbol{x}_{<j})$ is the conditional inverse CDF corresponding to the conditional CDF $F_j(x_j|\boldsymbol{x}_{<j})$. Let $F_p(\boldsymbol{x})$ denote the Rosenblatt transformation for distribution p, and let $F_p^{-1}(\boldsymbol{u})$ denote its inverse as defined above. Assuming the random variables are continuous, the Rosenblatt transformation transforms the samples from any distribution to samples from the Uniform distribution (i.e., the pushforward of the Rosenblatt transformation is the uniform distribution and the pushforward of a uniform distribution through the inverse Rosenblatt is the distribution p).

Proof. Given any continuous target distribution p, we can construct an invertible SCM whose observed distribution is p. Specifically, if we let q denote the exogenous noise distribution, then the following invertible and autoregressive function f—which defines an invertible SCM via Lemma 2—can be used to match the SCM distribution to p:

$$f(\epsilon) = F_p \circ F_q^{-1}(\epsilon) \,, \tag{17}$$

where F_q^{-1} maps to the uniform distribution and then F_p maps to the target distribution per the properties of the Rosenblatt transformation. The function is invertible since both functions are invertible. Additionally, both functions are autoregressive and thus the composition is autoregressive. Therefore, f represents a valid invertible SCM whose observed distribution is p. □

A.2 Proof of Lemma 2

Proof. We first prove one direction. Given an invertible SCM defined by it's causal mechanisms $\{\tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j})\}_{j=1}^m$, the observed variables are given recursively as:

$$x_j = \tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j}) \,. \tag{18}$$

We now define the corresponding f as in the lemma:

$$f(\epsilon) \triangleq [\tilde{f}_1(\epsilon_1), \quad \tilde{f}_2(\epsilon_2, \underbrace{\tilde{f}_1(\epsilon_1)}_{\text{recover } x_1}), \quad \tilde{f}_3(\epsilon_3, \underbrace{\tilde{f}_1(\epsilon_1), \tilde{f}_2(\epsilon_2, \tilde{f}_1(\epsilon_1))}_{\text{recover } \boldsymbol{x}_{<3}}), \quad \cdots]^\top \,. \tag{19}$$

We need to prove that the observed variables are equivalent to the given SCM. Formally, we will prove by induction on $j \in [m]$ the hypothesis that $[f(\epsilon)]_j = \tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j}) = x_j, \ \forall \epsilon \in \mathbb{R}^m$. The base case is trivial from the definition in (19), i.e., $\forall \epsilon \in \mathbb{R}^m$, $[f(\epsilon)]_j = \tilde{f}_1(\epsilon_1) = x_j$. For the inductive step, we have the following:

$$[f(\epsilon)]_{j+1} = \tilde{f}_{j+1}(\epsilon_{j+1}, \underbrace{\tilde{f}_1(\epsilon_1)}_{x_1}, \underbrace{\tilde{f}_2(\epsilon_2, \tilde{f}_1(\epsilon_1))}_{x_2}, \cdots) = \tilde{f}_{j+1}(\epsilon_{j+1}, \boldsymbol{x}_{<j+1}) = x_{j+1} \tag{20}$$

where the first equals is by (19), the second is by the inductive hypothesis, and the last is by definition of the SCM.

Now we prove the other direction. Given an invertible autoregressive function $f \in \mathcal{F}_I \cap \mathcal{F}_A$, we define the following recursive set of mechanism functions:

$$\forall j, x_j \equiv \tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j}) \triangleq [f([f^{-1}(\boldsymbol{x}_{<j}, \cdot)]_{<j}, \epsilon_j, \cdot)]_j \,. \tag{21}$$

Again, we will prove that these functional forms are equivalent via induction on j for the hypothesis that $\tilde{f}_j(\epsilon_j, \boldsymbol{x}_{<j}) = [f(\epsilon)]_j = x_j$. The base case is trivial based on (21):

$$\tilde{f}_1(\epsilon_1) = [f([f^{-1}(\boldsymbol{x}_{<1}, \cdot)]_{<1}, \epsilon_1, \cdot)]_1 = [f(\epsilon_1, \cdot)]_1 = x_1 \tag{22}$$

For the inductive step, we use the definition of $\bar{f}_{<j}$ and its inverse from Lemma 5 and derive the final result:

$$\tilde{f}_{j+1}(\epsilon_{j+1}, \boldsymbol{x}_{<j+1}) = [f([f^{-1}(\boldsymbol{x}_{<j}, \cdot)]_{<j}, \epsilon_j, \cdot)]_j = [f(\bar{f}^{-1}_{<j}(\boldsymbol{x}_{<j}), \epsilon_j, \cdot)]_j = [f(\epsilon_{<j}, \epsilon_j, \cdot)]_j = x_j \,. \tag{23}$$

□

A.3 Proof of Lemma 3

Proof. **Step 1:** Prove $\left\{j : \left[f^{-1}\right]_j \neq \left[f'^{-1}\right]_j\right\} \subset \mathcal{I}\left(\tilde{f}, \tilde{f}'\right)$.

For all $j \in \left\{j : \left[f^{-1}\right]_j \neq \left[f'^{-1}\right]_j\right\}$, there exists some $\boldsymbol{z}$, such that

$$\left[f^{-1}(\boldsymbol{z})\right]_j \neq \left[f'^{-1}(\boldsymbol{z})\right]_j, \tag{24}$$

given that f, f' are auto-regressive function, we conclude there exists some $(\boldsymbol{z}_{<j}, z_j)$ such that

$$\epsilon_j = [f^{-1}(\boldsymbol{z}_{<j}, z_j, \cdot)]_j \neq [f'^{-1}(\boldsymbol{z}_{<j}, z_j, \cdot)]_j = \epsilon'_j. \tag{25}$$

we have, for ϵ_j, ϵ'_j and such $\boldsymbol{z}_{<j}$ there holds

$$\begin{aligned}
\tilde{f}_j(\epsilon_j, \boldsymbol{z}_{<j}) \overset{(25)}{=} z_j &\overset{(25)}{=} \tilde{f}'_j(\epsilon'_j, \boldsymbol{z}_{<j}) \\
&= [f'([f'^{-1}(\boldsymbol{z}_{<j}, \cdot)]_{<j}, \epsilon'_j, \cdot)]_j \\
&\overset{(a)}{\neq} [f'([f'^{-1}(\boldsymbol{z}_{<j}, \cdot)]_{<j}, \epsilon_j, \cdot)]_j \\
&= \tilde{f}'_j(\epsilon_j, \boldsymbol{z}_{<j}).
\end{aligned} \tag{26}$$

where (a) comes from the $f' \in \mathcal{F}_I$. Thus it implies $j \in \mathcal{I}\left(\tilde{f}, \tilde{f}'\right)$.

Step 2: Prove $\mathcal{I}\left(\tilde{f}, \tilde{f}'\right) \subset \left\{j : \left[f^{-1}\right]_j \neq \left[f'^{-1}\right]_j\right\}$.

For all $j \in \mathcal{I}\left(\tilde{f}, \tilde{f}'\right)$, there exists some $(\epsilon_j, \boldsymbol{z}_{<j})$, such that

$$z_j \triangleq \tilde{f}_j(\epsilon_j, \boldsymbol{z}_{<j}) \neq \tilde{f}'_j(\epsilon_j, \boldsymbol{z}_{<j}) \triangleq z'_j, \tag{27}$$

Define

$$\boldsymbol{z}_{\leq j} \triangleq [\boldsymbol{z}_{<j}, z_j] \quad \text{and} \quad \boldsymbol{z}'_{\leq j} \triangleq [\boldsymbol{z}_{<j}, z'_j], \tag{28}$$

then we have

$$[f^{-1}(\boldsymbol{z}_{\leq j}, \cdot)]_j = \epsilon_j = [f'^{-1}(\boldsymbol{z}'_{\leq j}, \cdot)]_j, \tag{29}$$

given that $f, f' \in \mathcal{F}_I$, we conclude,

$$[f^{-1}(\boldsymbol{z}_{\leq j}, \cdot)]_j \neq [f'^{-1}(\boldsymbol{z}_{\leq j}, \cdot)]_j, \tag{30}$$

which implies $j \in \left\{j : \left[f^{-1}\right]_j \neq \left[f'^{-1}\right]_j\right\}$. □

B Auxillary results and proofs of Section 5

B.1 Proof of Lemma 4

Proof. We only need to prove that it satisfies reflexivity, symmetry, and transitivity.

1. Reflexivity - Letting $g' = g$ and $f' = f$ in the definition, it is trivial to see that $\forall d, d'$
$$g \circ f_{d'} \circ f_d^{-1} \circ g^{-1} = g \circ f_{d'} \circ f_d^{-1} \circ g^{-1},$$
and thus $(g, f) \simeq_C (g, f)$.

2. Symmetry - Similarly, it is trivial to see that $\forall d, d'$,
$$\begin{aligned} & g \circ f_{d'} \circ f_d^{-1} \circ g^{-1} = g' \circ f'_{d'} \circ f_d'^{-1} \circ g'^{-1} \\ \iff \quad & g' \circ f'_{d'} \circ f_d'^{-1} \circ g'^{-1} = g \circ f_{d'} \circ f_d^{-1} \circ g^{-1}, \end{aligned}$$
and thus $(g, f) \simeq_C (g', f') \Leftrightarrow (g', f') \simeq_C (g, f)$.

3. Transitivity - For (g, f), (g', f') and (g'', f''), we can derive the transitive property by applying the property twice to the first two and the last two pairs $\forall d, d'$:
$$g \circ f_{d'} \circ f_d^{-1} \circ g^{-1} = g' \circ f'_{d'} \circ f_d'^{-1} \circ g'^{-1} = g'' \circ f''_{d'} \circ f_d''^{-1} \circ g''^{-1},$$
which means that $(g, f) \simeq_C (g'', f'')$.

□

B.2 Proof of Theorem 1

The proof of Theorem 1 relies heavily on one the following key lemma that provides a necessary and sufficient condition for the composition of two invertible functions to be equal.

Lemma 6 (Invertible Composition Equivalence)**.** *For two pairs of invertible functions (f_1, f_2) and (f'_1, f'_2), the following two conditions are equivalent:*

1. *The compositions are equal:*
$$f_1 \circ f_2 = f'_1 \circ f'_2.$$

2. *There exists an intermediate invertible function h s.t.*
$$f'_1 = f_1 \circ h^{-1}, f'_2 = h \circ f_2. \tag{31}$$

See Appendix subsection C.2 for the proof of this Lemma.

Proof of Theorem 1. The basic idea is to use repeated application of Lemma 6 under the constraint that h_1 and h_2 must be shared across for all d and g and g^{-1} must be inverses of each other.

For one direction as in Lemma 6, if (9) holds, it is nearly trivial to show (8), for all d, d':
$$\begin{aligned} g' \circ f'_{d'} \circ f_d'^{-1} \circ g'^{-1} &= (g \circ h_1^{-1}) \circ (h_1 \circ f_{d'} \circ h_2) \circ (h_2^{-1} \circ f_d^{-1} \circ h_1^{-1}) \circ (h_1 \circ g^{-1}) \\ &= g \circ f_{d'} \circ f_d^{-1} \circ g^{-1}. \end{aligned}$$

To prove the other direction, let us define the following functions for a specific (d, d') (we will treat the case of all (d, d') afterwards): $f_1 \triangleq g^{-1}, f_2 \triangleq f_d^{-1}, f_3 \triangleq f_{d'}$, and $f_4 \triangleq g$ and similarly f'_1, f'_2, f'_3, and f'_4 for the other side. Given these definitions, we can write the property as:
$$f_4 \circ f_3 \circ f_2 \circ f_1 = f'_4 \circ f'_3 \circ f'_2 \circ f'_1.$$

By recursively applying Lemma 6 for each of the three function compositions, we arrive at the following fact:
$$\exists\, h_1, h_2, h_3, \quad \text{s.t.} \quad \begin{cases} f'_1 = h_1 \circ f_1 \text{ and } f'_4 \circ f'_3 \circ f'_2 = f_4 \circ f_3 \circ f_2 \circ h_1^{-1} \\ f'_2 = h_2 \circ f_2 \circ h_1^{-1} \text{ and } f'_4 \circ f'_3 = f_4 \circ f_3 \circ h_2^{-1} \\ f'_3 = h_3 \circ f_3 \circ h_2^{-1} \text{ and } f'_4 = f_4 \circ h_3^{-1} \end{cases}$$

By using the definitions of f_1, f_2, etc., we can now derive the following:

$$\begin{aligned} g' &= g \circ h_3^{-1} \\ f'_{d'} &= h_3 \circ f_{d'} \circ h_2^{-1} \\ f_d'^{-1} &= h_2 \circ f_d^{-1} \circ h_1^{-1} \\ g'^{-1} &= h_1 \circ g^{-1} . \end{aligned}$$

We can connect the first and the last equality to derive that $h_3 = h_1$:

$$\begin{aligned} & g'^{-1} = h_1 \circ g^{-1} \\ \Leftrightarrow\ & g' = g \circ h_1^{-1} = g \circ h_3^{-1} \\ \Leftrightarrow\ & h_1^{-1} = h_3^{-1} \\ \Leftrightarrow\ & h_1 = h_3 . \end{aligned}$$

Thus, there are only two free functions. Specifically, for any fixed pair of (d, d') there exist $h_{1,d,d'} (\equiv h_{3,d,d'})$ and $h_{2,d,d'}$ such that

$$g' = g \circ h_{1,d,d'}^{-1}, f'_d = h_{1,d,d'} \circ f_d \circ h_{2,d,d'}^{-1}, \text{ and } f'_{d'} = h_{1,d,d'} \circ f_{d'} \circ h_{2,d,d'}^{-1}.$$

Finally, we tackle the case of all (d, d') by assuming that there could be unique functions $h_{1,d,d'}$ and $h_{2,d,d'}$ for all pairs of (d, d') and show that they are in fact equal. Because the condition holds for *all* (d, d'), we know that for any particular (d, d') and (d'', d), we have the following two things based on the proof above:

$$\begin{aligned} & g' \circ f'_{d'} \circ f_d'^{-1} \circ g'^{-1} = g \circ f_{d'} \circ f_d^{-1} \circ g^{-1} \\ \Leftrightarrow\ & \exists\, h_{1,d,d'}, h_{2,d,d'} \text{ s.t. } \begin{cases} g' = g \circ h_{1,d,d'}^{-1} \\ f'_d = h_{1,d,d'} \circ f_d \circ h_{2,d,d'}^{-1} \\ f'_{d'} = h_{1,d,d'} \circ f_{d'} \circ h_{2,d,d'}^{-1} \end{cases} \end{aligned}$$

$$\begin{aligned} & g' \circ f'_d \circ f'_{d''}{}^{-1} \circ g'^{-1} = g \circ f_d \circ f_{d''}^{-1} \circ g^{-1} \\ \Leftrightarrow\ & \exists\, h_{1,d'',d}, h_{2,d'',d} \text{ s.t. } \begin{cases} g' = g \circ h_{1,d'',d}^{-1} \\ f'_{d''} = h_{1,d'',d} \circ f_{d''} \circ h_{2,d'',d}^{-1} \\ f'_d = h_{1,d'',d} \circ f_d \circ h_{2,d'',d}^{-1} \end{cases} . \end{aligned}$$

By equating the RHS for the g' equations, we can thus derive that:

$$\begin{aligned} & g \circ h_{1,d,d'}^{-1} = g \circ h_{1,d'',d}^{-1} \\ \Leftrightarrow\ & h_{1,d,d'} = h_{1,d'',d} . \end{aligned}$$

Using this fact and similarly by equating the RHS for the f'_d equations, we can derive:

$$\begin{aligned} & f'_d = h_{1,d,d'} \circ f_d \circ h_{2,d,d'}^{-1} = h_{1,d'',d} \circ f_d \circ h_{2,d'',d}^{-1} = h_{1,d,d'} \circ f_d \circ h_{2,d'',d}^{-1} \\ \Leftrightarrow\ & h_{2,d,d'}^{-1} = h_{2,d'',d}^{-1} \\ \Leftrightarrow\ & h_{2,d,d'} = h_{2,d'',d} . \end{aligned}$$

By applying these facts to all possible triples of (d, d', d''), we can conclude that $\forall d, d',\ h_{1,d,d'} = h_1,\ h_{2,d,d'} = h_2$, i.e., these intermediate functions must be independent of d and d'. Finally, we can adjust notation so that $\forall d,\ f'_d = \tilde{h}_1 \circ f_d \circ \tilde{h}_2$ and $g' = g \circ \tilde{h}_1^{-1}$, where $\tilde{h}_1 \triangleq h_1$ and $\tilde{h}_2 \triangleq h_2^{-1}$, which matches the result in the theorem. □

B.3 Proof of Theorem 2

Lemma 7 (Swapping Lemma)**.** *Given that the first canonical counterfactual property is satisfied, i.e.,* $f_1 = \text{Id}$*, denote* f' *as SCM constructed by* $f' = h_1 \circ f \circ h_2(x)$*, where* $h_1 = h_2$ *denote swapping the* j*-th feature with* j'*-th feature. Then there exists* g' *such that*

$$(g, f) \simeq_C (g', f'),\ f'_1 = \text{Id},\ \mathcal{I}(f') = (\mathcal{I}(f) \setminus \{j\}) \cup \{j'\}.$$

if the following conditions hold

$$j \in \mathcal{I}(f) \text{ and } \forall \tilde{j} : j < \tilde{j} \leq j',\ \tilde{j} \notin \mathcal{I}(f).$$

Built upon swapping Lemma, we move to our main result on the existence of equivalent Canonical ILD. See subsection C.3 for proofs.

Proof of Theorem 2. At high level the proof is organized in the following two steps.

(Step 1) we use Theorem 1 to construct an equivalent counterfactual $\left(g^{(0)}, f^{(0)}\right) \simeq_C (g, f)$ by choosing two invertible functions $h_1 = f_1^{-1}$ and $h_2 = \text{Id}$. In this way, Theorem 1 implies

$$f_1^{(0)} = h_1 \circ f_1 \circ h_2 = f_1^{-1} \circ f_1 \circ \text{Id} = \text{Id}$$
$$\forall d > 1, \quad f_d^{(0)} = h_1 \circ f_d \circ h_2 = f_1^{-1} \circ f_d \circ \text{Id} = f_1^{-1} \circ f_d, \quad \text{and} \quad g^{(0)} = g \circ h_1^{-1} = g \circ f_1 .$$

Equipped with $\left(g^{(0)}, f^{(0)}\right)$, we can show that part I of Def. 8 is satisfied, i.e., $f_1^{(0)} = \text{Id}$. Choosing $h_2 = \text{Id}$, we could prove **(Step 1)** could guarantee the distribution equivalence. We can further construct a series of equivalent counterfactuals iteratively to gradually satisfy part II of Def. 8. Specifically, in **(Step 2)**, we recursively construct, for all $k \in \{1, 2, \ldots, k^{\text{last}}\}$,

$$f^{(k)} \triangleq h_{j(k) \leftrightarrow j'(k)} \circ f^{(k-1)} \circ h_{j(k) \leftrightarrow j'(k)},$$

and

$$g^{(k)} \triangleq g^{(k-1)} \circ h^{-1}_{j(k) \leftrightarrow j'(k)} = g^{(k-1)} \circ h_{j(k) \leftrightarrow j'(k)} ,$$

where $h_{j(k) \leftrightarrow j'(k)}$ denotes swapping the $j(k)$-th and $j'(k)$-th feature values, i.e.,

$$h_{j \leftrightarrow j'}(\boldsymbol{x}) \triangleq [x_1, x_2, \cdots, x_{j-1}, x_{j'}, x_{j+1}, \cdots, x_{j'-1}, x_j, x_{j'+1}, \cdots, x_m]^T, \tag{32}$$

and further define

$$j'(k) \triangleq \max \left\{ j, j \notin \mathcal{I}\left(f^{(k-1)}\right) \right\}, \text{ and } j(k) \triangleq \max \left\{ j < j'(k), j \in \mathcal{I}\left(f^{(k-1)}\right) \right\} . \tag{33}$$

In high level, at each iteration, we seek the largest index $j'(k)$ which does not lies in the previous intervention set $\mathcal{I}\left(f^{(k-1)}\right)$, and swap it with the largest index $j(k)$ which is smaller than $j'(k)$. We terminate at k when $\left\{ j < j'(k), j \in \mathcal{I}\left(f^{(k-1)}\right) \right\} = \emptyset$.

By the definition of $j'(k), j(k)$ in (33), we can show that **1)** for each swap step k, there holds

$$j(k) \in \mathcal{I}\left(f^{(k-1)}\right), \text{ and } \forall \tilde{j} : j(k) < \tilde{j} \leq j'(k), \ \tilde{j} \notin \mathcal{I}\left(f^{(k-1)}\right), \tag{34}$$

which implies Lemma 7 can be applied to ensure the counterfactual equivalence at each step.

2) When meeting the stopping criterion at step k^{last}, i.e.,

$$\left\{ j < j'(k^{\text{last}}), j \in \mathcal{I}\left(f^{(k^{\text{last}}-1)}\right) \right\} = \emptyset, \tag{35}$$

there holds

$$\forall j \in \mathcal{I}\left(f^{(k^{\text{last}}-1)}\right), \quad j > m - \left| \mathcal{I}\left(f^{(k^{\text{last}}-1)}\right) \right| ,$$

i.e., $\left(g^{(k^{\text{last}}-1)}, f^{(k^{\text{last}}-1)}\right)$ is in canonical form. Chaining **1)** and **2)**, we conclude

$$\exists\, (g', f') \triangleq \left(g^{k^{\text{last}}-1}, f^{k^{\text{last}}-1}\right) \in \mathcal{C} \text{ s.t. } (g', f') \simeq_C (g, f).$$

Note that $g^{(k)} \circ f_d^{(k)} = g^{(k-1)} \circ f_d^{(k-1)} \circ h_{j(k) \leftrightarrow j'(k)}$, and linear operator $h_{j(k) \leftrightarrow j'(k)}$ is orthogonal, then iteratively, we conclude $(g', f') \simeq_D (g, f)$.

To prove **1)**, observe in (33), $j(k)$ is the largest index in the intervention set which is smaller than $j'(k)$. This simply implies (34).
To prove **2)**, suppose when meeting the stopping criterion at step k^{last}, there holds

$$\exists\, j \in \mathcal{I}\left(f^{(k^{\text{last}}-1)}\right) \quad \text{such that} \quad j \leq m - \left| \mathcal{I}\left(f^{(k^{\text{last}}-1)}\right) \right| . \tag{36}$$

It implies that

$$\exists\, \hat{j} \notin \mathcal{I}\left(f^{(k^{\text{last}}-1)}\right) \quad \text{and} \quad \hat{j} \in \left\{ m - \left| \mathcal{I}\left(f^{(k^{\text{last}}-1)}\right) \right| + 1, \ldots, m \right\} .$$

Then we can choose $j'(k) = \hat{j}$, implying $j \in \left\{ j < j'(k), j \in \mathcal{I}\left(f^{(k^{\text{last}}-1)}\right)\right\} \neq \emptyset$, contradict to (35). This concludes the proof of part I in Theorem 2.

It remains to prove that the construction of $f^{(0)}$ in the **step 1** does not change the intervention set.

1) For any $j \notin \mathcal{I}(f)$, for any pairs d, d', we have $\left[f_d^{-1}\right]_j = \left[f_{d'}^{-1}\right]_j$, based on the construction of $f^{(0)}$, we have

$$\left[f_d^{(0)^{-1}}\right]_j = [f_d^{-1} \circ f_1]_j = [f_{d'}^{-1} \circ f_1]_j = \left[f_{d'}^{(0)^{-1}}\right]_j \tag{37}$$

thus, $\mathcal{I}(f_d^{(0)}, f_{d'}^{(0)}) \subset \mathcal{I}(f_d, f_d')$.

2) For any $j \in \mathcal{I}(f)$, there exists d, d' and z, such that $[{f_d}^{-1}(z)]_j \neq [{f_{d'}}^{-1}(z)]_j$. Note that f_1 is a bijective function, there exists z' such that $z = f_1(z')$, we have

$$\begin{aligned}
&[f_d^{-1}(z)]_j \neq [f_{d'}^{-1}(z)]_j \\
\Leftrightarrow &\left[f_d^{-1}(f_1(z'))\right]_j \neq \left[f_{d'}^{-1}(f_1(z')\right]_j \\
\Leftrightarrow &\left[f_d^{(0)^{-1}}(z')\right]_j \neq \left[f_{d'}^{(0)^{-1}}(z')\right]_j \\
\Leftrightarrow &\, j \in \mathcal{I}\left(f_d^{(0)}, f_{d'}^{(0)}\right)
\end{aligned}$$

thus $\mathcal{I}\left(f_d^{(0)}, f_{d'}^{(0)}\right) \supset \mathcal{I}(f_d, f_{d'})$. Combining **1)** and **2)**, we have $\mathcal{I}\left(f_d^{(0)}, f_{d'}^{(0)}\right) = \mathcal{I}(f_d, f_{d'})$. This show that the construction of **step 1** does not change the intervention set, combining the fact in **step 1**, we iteratively used swapping Lemma 7, and swapping Lemma 7 does not change the intervention set size, i.e., $\mathcal{I}(f') = (\mathcal{I}(f) \setminus \{j\}) \cup \{j'\}$, we conclude that $\left|\mathcal{I}\left(f_d^{(0)}, f_{d'}^{(0)}\right)\right| = \left|\mathcal{I}(f_d, f_{d'})\right|$ This completes the proof. □

To help understanding, we design a simple linear ILD model to explain the theorem procedure.

Example 1. *Suppose we have a* 4*-dimensional ILD model* (g, f) *containing* 2 *domains, where*

$$f_1 \triangleq \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}, f_2 \triangleq \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}, g \textit{ invertible.}$$

Following the proof of Theorem 2, we have Following **Step 1** in the proof of Theorem 2, we have $h_1 = f_1^{-1}$,

$$f_1^{(0)} = f_1^{-1} \circ f_1 \,, f_2^{(0)} = f_1^{-1} \circ f_2$$

$$g^{(0)} = g \circ f_1 \,,$$

$$f_1^{(0)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, f_2^{(0)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ -1 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$g^{(0)} = g \circ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Notice that $\mathcal{I}(f^{(0)}) = \{2, 3\}$. Following **Step 2** in the proof of Theorem 2, we first swap $j = 3$ and $j' = 4$,

$$h_{3\leftrightarrow 4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, g^{(1)} = g \circ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

We have $f^{(2)} \triangleq h_{3\leftrightarrow 4} \circ f^{(1)} \circ h_{3\leftrightarrow 4}$

$$f_1^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, f_2^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -1 & -1 & 0 & 1 \end{bmatrix}.$$

Notice that $\mathcal{I}(f^{(1)}) = \{2, 4\}$. Following **Step 2** in the proof of Theorem 2, we first swap $j = 2$ and $j' = 3$,

$$h_{2\leftrightarrow 3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, g^{(2)} = g \circ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

We have $f^{(3)} \triangleq h_{2\leftrightarrow 3} \circ f^{(2)} \circ h_{2\leftrightarrow 3}$

$$f_1^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, f_2^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 2 & 0 \\ -1 & 0 & -1 & 1 \end{bmatrix}.$$

$$g^{(2)} = g \circ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

Notice that $(g^{(2)}, f^{(2)})$ is the canonical form. They are counterfactually equivalent to each other by checking definition.

B.4 Proof of Theorem 4

Proof. All $(g', f') \in \left\{(\tilde{g}, \tilde{f}) \in \mathcal{C} : (\tilde{g}, \tilde{f}) \simeq_D (g, f), (\tilde{g}, \tilde{f}) \simeq_C (g, f)\right\}$ satisfy $f'_d = h^{-1} \circ f_d \circ h, \quad \forall d$. According to Theorem 1, i.e., there exists h_1, h_2, such that $f'_d = h_1 \circ f_d \circ h_2$ when $d = 1$, $\text{Id} = f'_1 = h_2^{-1} \circ \text{Id} \circ h_2$, thus $h_1 = h_2^{-1}$. To satisfy distributed equivalence, according to (6), we have $f'_\sharp \mathcal{N}(0, I) = f_\sharp \mathcal{N}(0, I) \to h_\sharp \mathcal{N}(0, I) = \mathcal{N}(0, I)$. Since h is linear, we could write the transformation $h(x)$ for any $x \sim \mathcal{N}(0, 1)$ as Hx. Notice that the variance of $h_\sharp \mathcal{N}(0, I)$ is $HIH^\top$, we have

$$I = HIH^\top$$
$$H^\top = H^{-1}.$$

Thus, H is orthonormal.

To show that all such f' satisfy $\mathcal{I}(f) = \mathcal{I}(f')$, we consider all three cases of H.

H is a permutation matrix. Permutation matrix will not change the sparsity. Proof is in Lemma 7. Permutation over $i \neq j$ and $i, j \notin \mathcal{I}(f)$, $i, j \notin \mathcal{I}(f')$. If either $i \in \mathcal{I}(f)$ or $j \in \mathcal{I}(f)$, f' will not be autoregressive.

H is a reflection matrix. Since H is a diagonal matrix with only ± 1 on the diagonal, we have

$$[f'^{-1}_d]_{i,j} = H_{ii}[f_d^{-1}]_{ij}H_{jj}. \tag{38}$$

When $i = j$, there holds $H_{ii}H_{jj} = 1$. Thus, $\text{diag}(f'^{-1}_d) = \text{diag}(f_d^{-1})$.

For all d, recall that Lemma 3 implies

$$\mathcal{I}(f) = \bigcup_{d \neq d'} \mathcal{I}(f_d, f_{d'}) = \bigcup_d \mathcal{I}(f_1, f_d) \tag{39}$$

given that $f, f' \in \mathcal{C}$, we have

$$f_1 = f'_1 = \text{Id}.$$

Thus, we have for all $i \notin \mathcal{I}(f)$,

$$\left[f_1^{-1}\right]_i = \left[f_d^{-1}\right]_i = e_i.$$

Thus, we have

$$\left[f'^{-1}_d\right]_{i,j} = \left[f_d^{-1}\right]_{i,j} = 0, j \neq i$$
$$\left[f'^{-1}_d\right]_{i,j} = \left[f_d^{-1}\right]_{i,j} = 1, j = i$$

this indicates $i \notin \mathcal{I}(f')$.

For all $i \in \mathcal{I}(f)$, there exists $d, [f_d^{-1}]_{i,j} \neq 0, j < i$, we know

$$[f'^{-1}_d] = \pm[f_d^{-1}]_{i,j} \neq 0, j < i$$
$$[f'^{-1}_d] = [f_d^{-1}]_{i,j}, j = i$$
$$[f'^{-1}_d] = \pm[f_d^{-1}]_{i,j} = 0, j > i$$

. Thus $i \in \mathcal{I}(f')$.

H is a rotation matrix. All rotation matrix could be the multiplication of several rotation matrix along the eigenvectors, thus we only consider rotation along the eigenvectors with angle θ. There are three cases: 1) $\theta = \frac{2}{2k+1}\pi, k \in \mathbb{Z}$, 2) $\theta = k\pi, k \in \mathbb{Z}$, 3) any other rotation matrix. Case 1) Similar case with permutation matrix. Case 2) is reflection matrix, Case 3) does not preserve autoregressiveness.

For a general rotation matrix in case 3), the subspace span(e_i, e_j) where $i < j$, we have $H_{ij} \neq 0$ and $H_{ji} \neq 0$, thus for $f \in \mathcal{F}_A$ where $f_{ij} \neq 0$, then we have

$$f'_{i,j} = H_i^{-1}(fH_{:,j}) = (H_{i,i}^{-1} \cdot f_{i,i} \cdot H_{i,j} + H_{i,j}^{-1} \cdot f_{j,j} \cdot H_{j,j}) \neq 0$$

This breaks the autoregressiveness of the ILD.

□

C Proofs of Lemmata

C.1 Miscellaneous Proofs

Lemma 8 (Invertible Function Rewrite)**.** *Given any two invertible functions $f : \mathcal{X} \to \mathcal{Y}$ and $f' : \mathcal{X} \to \mathcal{Y}$, f' can be decomposed into the composition of f and another invertible function. Specifically, f' can be decomposed in the following two ways:*

$$f' \equiv f \circ h_{\mathcal{X}} \tag{40}$$
$$f' \equiv h_{\mathcal{Y}} \circ f\,, \tag{41}$$

where $h_{\mathcal{X}} \triangleq f^{-1} \circ f' : \mathcal{X} \to \mathcal{X}$ and $h_{\mathcal{Y}} \triangleq f' \circ f^{-1} : \mathcal{Y} \to \mathcal{Y}$ are both invertible functions.

Proof of Lemma 8. The proof is straightforward. We first note that $h_{\mathcal{X}}$ and $h_{\mathcal{Y}}$ are invertible because they are compositions of invertible functions. Then, we have that:

$$f \circ h_{\mathcal{X}} = f \circ f^{-1} \circ f' = f' \tag{42}$$
$$h_{\mathcal{Y}} \circ f = f' \circ f^{-1} \circ f = f'\,. \tag{43}$$

□

C.2 Proof of Invertible Composition Equivalence Lemma 6

Proof of Lemma 6. For notational simplicity in this proof, we will let $g \triangleq f_1$, $f \triangleq f_2$, $g' \triangleq f'_1$ and $f' \triangleq f'_2$—note that g and f are just arbitrary invertible functions in this proof. Furthermore, without loss of generality, we will prove for the property $\exists\, h : g' = g \circ h, f' = h^{-1} \circ f$ which is equivalent to $\exists\, h : g' = g \circ h^{-1}, f' = h \circ f$. Thus, in the new notation, we are seeking to prove:

$$g \circ f = g' \circ f' \Leftrightarrow \exists\, h : g' = g \circ h, f' = h^{-1} \circ f \tag{44}$$

If $\exists\, h : g' = g \circ h, f' = h^{-1} \circ f$, then it is easy to show that $g \circ f = g' \circ f'$:

$$g' \circ f' = g \circ h \circ h^{-1} \circ f = g \circ f\,. \tag{45}$$

For the other direction, we will prove by contradiction. First, using Lemma 8, we can first rewrite g' and f' using the two uniquely determined invertible functions h_1 and h_2:

$$g' = g \circ h_1 \tag{46}$$
$$f' = h_2 \circ f. \tag{47}$$

Now, suppose that $g \circ f = g' \circ f'$ but $\nexists h$ such that $g' = g \circ h, f' = h^{-1} \circ f$. By the first assumption and the facts above, we can derive the following:

$$g \circ f = g' \circ f' = g \circ h_1 \circ h_2 \circ f \tag{48}$$
$$\Leftrightarrow f = h_1 \circ h_2 \circ f \tag{49}$$
$$\Leftrightarrow h_1^{-1} \circ f = h_2 \circ f \tag{50}$$

From the second assumption, i.e., $\nexists h : g' = g \circ h, f' = h^{-1} \circ f$, we have the following:

$$\forall\, h \text{ s.t. } g' = g \circ h, \text{ it holds that } f' \neq h^{-1} \circ f \tag{51}$$
$$\Rightarrow f' \neq h_1^{-1} \circ f \tag{52}$$
$$\Leftrightarrow h_2 \circ f \neq h_1^{-1} \circ f \tag{53}$$
$$\Leftrightarrow h_2 \neq h_1^{-1} \tag{54}$$
$$\Leftrightarrow h_2^{-1} \neq h_1\,, \tag{55}$$

where (51) is by assumption, (52) follows from (46) because h_1 is one particular h, (53) is by our rewrite of f' in (47), (54) is by the invertibility of f, and (55) is by invertibility of h_1 and h_2. Thus, there exists $\tilde{\boldsymbol{y}}$, such that $h_1^{-1}(\tilde{\boldsymbol{y}}) \neq h_2(\tilde{\boldsymbol{y}})$. Let us choose $\tilde{\boldsymbol{x}} \triangleq f^{-1}(\tilde{\boldsymbol{y}})$ for the $\tilde{\boldsymbol{y}}$ that satisfies the condition. For this $\tilde{\boldsymbol{x}}$, we then know that:

$$h_1^{-1} \circ f(\tilde{\boldsymbol{x}}) = h_1^{-1}(\tilde{\boldsymbol{y}}) \neq h_2(\tilde{\boldsymbol{y}}) = h_2 \circ f(\tilde{\boldsymbol{x}}) \tag{56}$$
$$\Leftrightarrow h_1^{-1} \circ f \neq h_2 \circ f\,. \tag{57}$$

But this leads to a direct contradiction of (50). Therefore, if $g \circ f = g' \circ f'$, then $\exists\, h : g' = g \circ h, f' = h^{-1} \circ f$. □

C.3 Proof of swapping Lemma 7

Before proving the swapping Lemma, we introduce a Lemma which is useful for proving Lemma 7.

Lemma 9. *For an ILD with* $f_1 = \text{Id}$, $\mathcal{I}(f_d, f_1) = \left\{j : [f_d]_j \neq [f_1]_j\right\}$.

Proof. Suppose $f_d^{-1}(\boldsymbol{x}) = \boldsymbol{x}'$ where $x'_j \neq x_j$, then $f_d(\boldsymbol{x}') = \boldsymbol{x}$ becasue that f_d is bijective. Then

$$[f_d(\boldsymbol{x}')]_j = x_j \neq x'_j = f_1(\boldsymbol{x}').$$

For any $j \notin \mathcal{I}(f)$, for any $\boldsymbol{x} = f_d(\boldsymbol{x}')$, we have $x'_j = [f_d^{-1}(\boldsymbol{x})]_j = [f_1^{-1}(\boldsymbol{x})]_j = x_j \Rightarrow x_j = x'_j$, thus

$$x_j = [f_d(\boldsymbol{x}')]_j = [f_d(\boldsymbol{x}')]_j = x'_j.$$

□

Proof of Lemma 7. First, note that because j' is not intervened, then we can derive that it's corresponding conditional function is independent of all but the j'-th value:

$$[f_d]_{j'} = [f_1]_{j'} \tag{58}$$
$$\Leftrightarrow \quad f_{d,j'}(\boldsymbol{x}_{\leq j'}) = f_{1,j'}(\boldsymbol{x}_{\leq j'}) = x_{j'}\,. \tag{59}$$

For the new model, we choose the invertible functions as swapping the j-th and j'-th feature values, i.e.,

$$h_1(\boldsymbol{x}) \triangleq [x_1, x_2, \cdots, x_{j-1}, x_{j'}, x_{j+1}, \cdots, x_{j'-1}, x_j, x_{j'+1}, \cdots, x_m]^T \tag{60}$$

and similarly for h_2, i.e., $h_2 \triangleq h_1$. Because h_1 and h_2 are invertible, we know that the new model will be in the same counterfactual equivalence class by Theorem 1. Construct $g' \triangleq g \circ h_1^{-1}$, and then for all d,

$$\begin{aligned}
f'_d(x) =& h_1 \circ f_d \circ h_2(x) \\
=& h_1 \circ f_d([x_1, x_2, \cdots, x_{j-1}, x_{j'}, x_{j+1}, \cdots, x_{j'-1}, x_j, x_{j'+1}, \cdots, x_m]^T]) \\
=& h_1 \circ f_d([y_1, y_2, \cdots, y_{j-1}, y_j, y_{j+1}, \cdots, y_{j'-1}, y_{j'}, y_{j'+1}, \cdots, y_m]^T]) \\
=& h_1 \circ [f_{d,i}(\boldsymbol{y}_{\leq i})]_{i=1}^m \\
=& \Big[f_{d,1}(\boldsymbol{y}_1), \cdots, f_{d,j-1}(\boldsymbol{y}_{\leq j-1}), f_{d,j'}(\boldsymbol{y}_{\leq j'}), f_{d,j+1}(\boldsymbol{y}_{\leq j+1}), \cdots, \\
& f_{d,j'-1}(\boldsymbol{y}_{\leq j'-1}), f_{d,j}(\boldsymbol{y}_{\leq j}), f_{d,j'+1}(\boldsymbol{y}_{\leq j'+1}), \cdots, f_{d,m}(\boldsymbol{y}_{\leq m}) \Big],
\end{aligned}$$

where we define $\boldsymbol{y} \triangleq h_2^{-1}(\boldsymbol{x})$.

We now need to check that the first canonical counterfactual property still holds.

$$f'_1 = h_1 \circ f_1 \circ h_2 = h_1 \circ \text{Id} \circ h_2 = h_1 \circ h_2 = \text{Id}\,, \tag{61}$$

where the last equals is because swap operations are self-invertible.

We move to check that the autoregressive property still holds for other domain SCMs.

1) For the j-th feature, we have that:

$$[f'_d(\boldsymbol{x})]_j = f_{d,j'}(\boldsymbol{y}_{\leq j'}) = f_{d,j'}(x_1, \cdots, x_{j-1}, x_{j'}, x_{j+1}, \cdots, x_{j'-1}, x_j) = x_j$$

where the last equals is because the $f_{d,j'}(\boldsymbol{y}_{\leq j'}) = y_{j'} = x_j$. This clearly satisfies the autoregressive property as $[f'_d]_j$ only depends on x_j.

2) For the j'-th feature, we have that:

$$[f'_d(\boldsymbol{x})]_{j'} = f_{d,j}(\boldsymbol{y}_{\leq j}) = f_{d,j'}(x_1, \cdots, x_{j-1}, x_{j'})$$

where again this satisfies the autoregressive property because all input indices are less than j' because $j < j'$. Now we handle the cases for other variables. If $\tilde{j} < j$, then we have the following:

$$[f'_d]_{\tilde{j}} = [h_1 \circ f_d \circ h_2]_{\tilde{j}} = [f_d \circ h_2]_{\tilde{j}} = f_{d,\tilde{j}}([h_2(\boldsymbol{x})]_{\leq \tilde{j}}) = f_{d,\tilde{j}}(x_1, \ldots, x_{\tilde{j}}) \tag{62}$$

3) Similarly if $j < \tilde{j} < j'$:

$$[f'_d]_{\tilde{j}} = f_{d,\tilde{j}}(x_1, \ldots, x_{j-1}, x_{j'}, x_{j+1}, \cdots, x_{\tilde{j}}) = x_{\tilde{j}}\,, \tag{63}$$

where we use the fact that there are no intervening nodes in between j and j'.

4) Finally, for $\tilde{j} > j'$, we have:

$$[f'_d]_{\tilde{j}} = f_{d,\tilde{j}}(x_1, \cdots, x_{j-1}, x_{j'}, x_{j+1}, \cdots, x_{j'-1}, x_j, x_{j'+1}, \cdots, x_{\tilde{j}})\,, \tag{64}$$

which is still autoregressive because $\tilde{j} > j'$ and $\tilde{j} > j$. Thus, the new f'_d is autoregressive and is thus a valid model.

It remains to prove that $\mathcal{I}(f') = (\mathcal{I}(f) \setminus \{j\}) \cup \{j'\}$.

1) When $k < j$, we have for all d,

$$[f'_d]_k = f_{d,k}(\boldsymbol{y}_{\leq k}) = f_{d,k}(\boldsymbol{x}_{\leq k}) = [f_d]_k\,,$$

then for all $k \in \mathcal{I}(f)$, there exists d_0, such that

$$[{f'_{d_0}}^{-1}]_k = [f'_{d_0}]_k \neq [f_1]_k = [{f'_1}^{-1}]_k.$$

Thus, $k \in \mathcal{I}(f')$.

If $k \notin \mathcal{I}(f)$,, we have for all d,

$$[{f'_d}^{-1}]_k = [f'_d]_k = [f_1]_k = [{f'_1}^{-1}]_k.$$

Thus $k \notin \mathcal{I}(f')$.

2) When $j \leq k < j'$, we have $\forall d, [f'_d(\boldsymbol{x})]_k = x_k \Rightarrow [{f'_d}^{-1}(\boldsymbol{x})]_k = x_k$. Thus we have $\forall d$, $[{f'_d}^{-1}]_k = [{f'_1}^{-1}]_k$, which means for all $j \leq k < j'$, $k \notin \mathcal{I}(f')$.

3) When $k = j'$, we have $\forall d, [f'_d]_{j'} = f_{d,j}(x_1, \cdots, x_{j-1}, x_{j'})$. Furthermore, since, $j \in \mathcal{I}(f)$, we have $\exists\, d_0$, $[f_{d_0}]_j \neq [f_1]_j$ by Lemma 9. Thus $[f'_{d_0}]_{j'} = [f_{d_0}]_j \neq [f_1]_j = [f'_1]_{j'} \Rightarrow j' \in \mathcal{I}(f')$ also by Lemma 9.

4) When $k > j'$, if $k \in \mathcal{I}(f), \exists\, d_1, d_2, [f_{d_1}]_k \neq [f_{d_2}]_k$, Chaining with (64), we have $[f'_{d_1}]_k \neq [f'_{d_2}]_k$. Thus, $k \in \mathcal{I}(f')$ by Lemma 9. Similarly, if $k \notin \mathcal{I}(f)$, then $k \notin \mathcal{I}(f')$

To summarize, $\mathcal{I}(f') = (\mathcal{I}(f) \setminus \{j\}) \cup \{j'\}$. □

D Simulated Experiment

D.1 Experiment Details

Dataset The ground truth latent SCM $f_d^* \in \mathcal{F}_{IA}$ takes the form $f_d^*(\boldsymbol{\epsilon}) = F_d^*\, \boldsymbol{\epsilon} + b_d^* \mathbb{1}_{\mathcal{I}}$ where $F_d^* = (I - L_d^*)^{-1}$, $L_d^* \in \mathbb{R}^{m \times m}$ is domain-specific lower triangular matrix that satisfies sparsity constraint, $b_d^* \in \mathbb{R}$ is a domain-specific bias and $\mathbb{1}_{\mathcal{I}}$ is an indicator vector where any entries corresponding to the intervention set are 1. To be specific, $[L_d^*]_{i,j} \sim \mathcal{N}(0, 1)$ and $b_d^* \sim \text{Uniform}(-2\sqrt{m/|\mathcal{I}|}, 2\sqrt{m/|\mathcal{I}|})$. The observation function takes the form $g^*(\boldsymbol{x}) = G^*\, \text{LeakyReLU}\,(\boldsymbol{x})$ where $G^* \in \mathbb{R}^{m \times m}$ and the slope of LeakyReLU is 0.5. To allow for similar scaling across problem settings, we set the determinant of G^* to be 1 and standardize the intermediate output of the LeakyReLU. The generated F_d^*, b_d^*, G^* all vary with random seeds and all experiments are repeated for 10 different seeds. We generate 100,000 samples from each domain for the training set and 1,000 samples from each domain in the validation and test set.

Model We test with two ILD models: *ILD-Relax-Can* which represents the relaxed canonical ILD form from Cor. 3 and a baseline model, *ILD-Dense* which has no sparsity restrictions on its latent SCM. To be specific, the latent SCM of *ILD-Dense* could be any model in $\mathcal{F}_{IA}$. We use $\mathcal{I}$ and $\mathcal{I}^*$ to represent the intervention set of the model and dataset, respectively. We note that for *ILD-Dense*, $\mathcal{I}$ contains all nodes and for *ILD-Relax-Can*, $\mathcal{I}$ contains only the last few nodes. Both models follow a similar structure as the ground truth. To be specific, the latent SCM takes the form $f_d(\boldsymbol{\epsilon}) = F_d\, \boldsymbol{\epsilon} + \boldsymbol{b}_d$ where $F_d = (I - L_d)^{-1} S_d, L_d \in \mathbb{R}^{m \times m}$, $S_d \in \mathbb{R}^{m \times m}$, and $\boldsymbol{b}_d \in \mathbb{R}^m$. The observation takes the form $g(\boldsymbol{x}) = G\, \text{LeakyReLU}\,(\boldsymbol{x}) + \boldsymbol{b}$ where $G \in \mathbb{R}^{m \times m}$, $\boldsymbol{b} \in \mathbb{R}^m$, and the slope of LeakyReLU is 0.5. In Figure 4a and Figure 4b, we add an illustration of the latent SCM for *ILD-Dense* and *ILD-Relax-Can* respectively. We emphasize a few main differences between the dataset and models here: (1)*ILD-Relax-Can*, $\mathcal{I}$ only contains the last few nodes while for the dataset, $\mathcal{I}^*$ could contain any node we specify. We note that *ILD-Dense* is equivalent to the *ILD-Relax-Can* which has all nodes in its intervention set. (2) There is no constraint on the determinant of G and standardization in $g(\boldsymbol{x})$. (3) The bias added to all dimensions in the ground truth model is the same scalar value, but the bias in the model is allowed to vary for each axis. (4) In the model, g is allowed a learnable bias.

Algorithm In the experiment, our algorithm only tries to fit the observed distribution for all models. As all models are strictly invertible, we fit the distribution via maximum likelihood estimation (MLE). To be specific, the objective is as below

$$\max_{g, f_1, \ldots, f_{N_d}} \mathbb{E}_d[p(\boldsymbol{x}|d)] \tag{65}$$

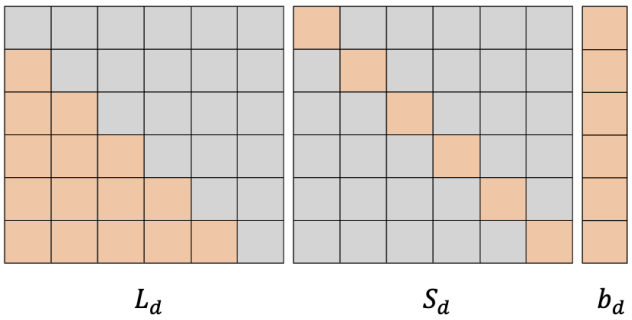
where $p(\boldsymbol{x}|d) = p_{\mathcal{N}}\left(f_d^{-1} \circ g^{-1}(\boldsymbol{x})\right) |J_{f_d^{-1} \circ g^{-1}}(\boldsymbol{x})|$.

Metric To evaluate the models, we compute the mean square error between the estimated counterfactual and ground truth counterfactual, i.e. Error $= \frac{2}{N_d(N_d-1)} \sum_{d' \neq d} \sum_d \|g^* \circ f_{d'}^* \circ (f_d^*)^{-1} \circ (g^*)^{-1}(\boldsymbol{x}_d) - g \circ f_{d'} \circ f_d^{-1} \circ g^{-1}(\boldsymbol{x}_d)\|^2$. As in practice, we can only check data fitting instead of counterfactual estimation, and we report the counterfactual error computed with the test dataset when the likelihood computed with the validation set is highest.

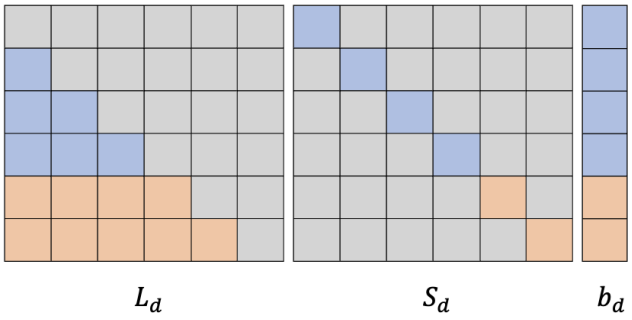
Training details We use Adam optimizer for both f and g with a lr $= 0.001, \beta_1 = 0.5, \beta_2 = 0.999$, and a batch size of is 500. We run all experiments for 50,000 iterations and compute validation likelihood and test counterfactual error every 100 steps. f is randomly initialized. Regarding g, G is initialized as an identity matrix and $\boldsymbol{b}$ is initialized as $\mathbf{0}$.

D.2 Extra Results

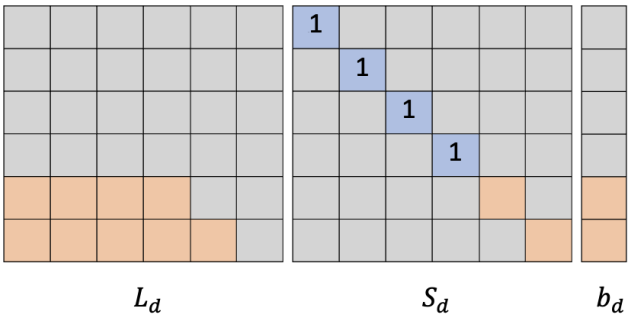
Case 1: No model misspecification In this section, we investigate the performance of *ILD-Dense* and *ILD-Relax-Can* while assuming that the ground truth intervention set only contains the last few nodes and we know the size of the intervention set.



(a) *ILD-Dense*



(b) *ILD-Relax-Can*

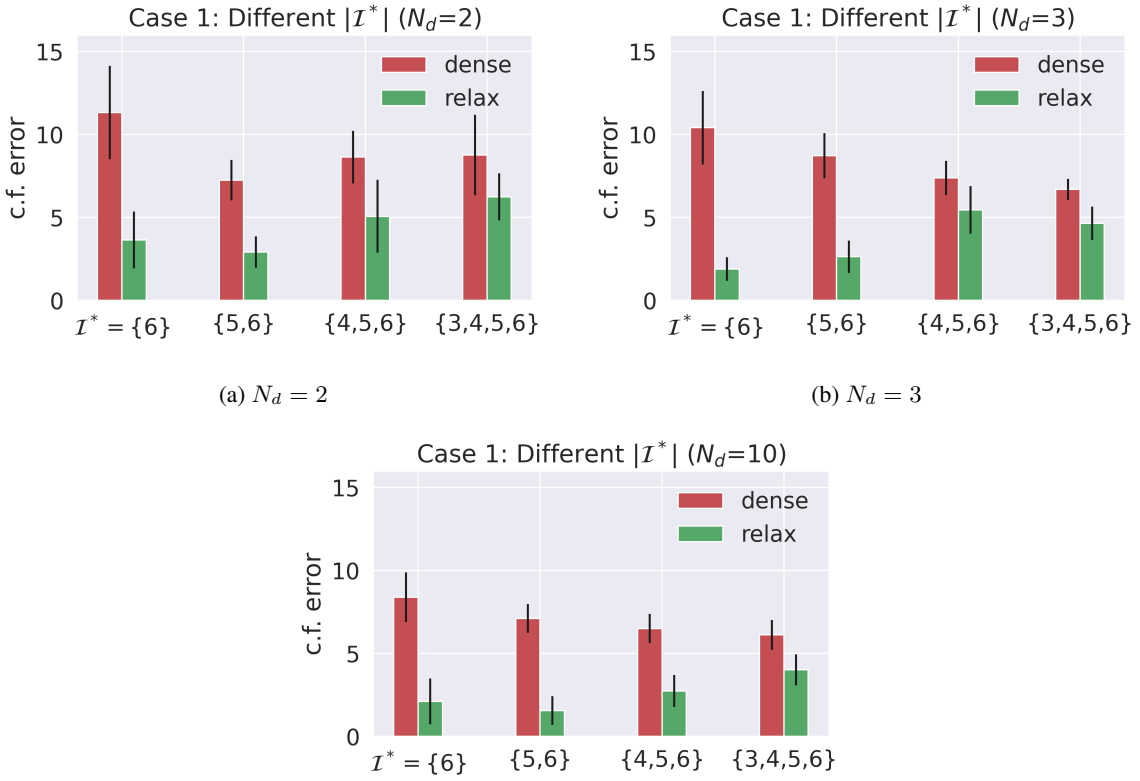


(c) *ILD-Can*

Figure 4: An illustration of the matrices/vector used to create f_d across the three ILD models when $m = 6$ and $|\mathcal{I}| = 2$. These are used such that $f_d(\boldsymbol{\epsilon}) = F_d\ \boldsymbol{\epsilon} + \boldsymbol{b}_d$ where $F_d = (I - L_d)^{-1} S_d$. The grey elements are 0, the orange elements are parameters that are different for different domains, and the blue elements are parameters shared across domains. We specify the value if it is a fixed number other than 0.

To understand how the true intervention set affects the gap between *ILD-Dense* and *ILD-Relax-Can*, we varied the size of the ground truth intervention. In Figure 5, we observe that the performance gap tends to be largest when the true intervention set is the most sparse and the performance of *ILD-Relax-Can* approaches to the performance of *ILD-Dense* as we increase the size. This makes sense as *ILD-Relax-Can* is a subset of *ILD-Dense* and they are equivalent when $\mathcal{I} = \{1, 2, 3, 4, 5, 6\}$. Additionally, even when the ground truth model is relatively dense (when $|\mathcal{I}^*|$ is close to m), *ILD-Relax-Can* is still better than *ILD-Dense*. We then investigate how models perform under tasks with different numbers of domains. In Figure 6, we change the number of domains in the datasets, and we observe that the performance gap does not seem to be sensitive to the number of domains though the absolute error seems to slightly decrease with more domains. Finally, we test how our algorithm scales with dimension when the number of domains is different. In Figure 7, we notice that *ILD-Relax-Can* is significantly better than *ILD-Dense* in 9 out of 12 cases. In the next paragraphs, we further investigate the 3 cases that do not outperform *ILD-Dense* to understand if it seems to be a theoretic or algorithmic/optimization problem.

We take a further investigation on the three cases where *ILD-Relax-Can* is close to or worse than *ILD-Dense*. As shown in Figure 8, when the latent dimension is 10 and the number of domains is 2, i.e. $m = 10$ and $N_d = 2$, the validation likelihood of *ILD-Relax-Can* is much lower than *ILD-Dense* especially in comparison to that with $m = 4, 6$.



(a) $N_d = 2$

(b) $N_d = 3$

(c) $N_d = 10$

Figure 5: Case 1: Test counterfactual error with different $\mathcal{I}^*$. To understand how the true intervention set affects the gap between *ILD-Dense* and *ILD-Relax-Can*, we varied the size of the ground truth intervention. It can be observed that the performance gap tends to be largest when the true intervention set is the sparsest and the performance of *ILD-Relax-Can* approaches to the performance of *ILD-Dense* as we increase the size.

We conjecture that the performance drop in terms of counterfactual error could be a result of the worse data fitting, i.e., the model does not fit the data well in terms of log-likelihood. As further evidence, we show the counterfactual error and corresponding validation log-likelihood in Table 3. We observe that the log-likelihood of *ILD-Dense* tends to be much lower when it has a larger counterfactual error than that of *ILD-Dense*. As for the relatively worse performance of *ILD-Relax-Can* when $m = 4, N_d = 2$ and $m = 4, N_d = 3$, we report the counterfactual error corresponding to each seed in Table 4 and Table 5 respectively. When the latent dimension is 4 and the number of domains is 2, i.e., $m = 4, N_d = 2$, *ILD-Relax-Can* is better than *ILD-Dense* with 9 out of 10 seeds. However, it fails significantly with seed 0 and thus leads to a larger average of counterfactual error. When $m = 4, N_d = 3$, *ILD-Relax-Can* is better than *ILD-Dense* with 7 out of 10 seeds but *ILD-Relax-Can* is not significantly better than *ILD-Dense* in terms of average error. We think this is more likely an optimization issue with lower dimensions, which is not explored by our theory. We conjecture that larger models with smoother optimization landscapes will perform better as we see in the Rotated MNIST case. We also note that these models are not significantly overparametrized and thus may not benefit from the traditional overparameterization that aids the performance of deep learning in many cases. Further investigation into overparameterized models may alleviate this algorithmic issue.

Despite some corner cases in which the optimization landscape may be difficult for these simple models, all the results point to the same trend that the sparse constraint motivated by our theoretic derivation indeed aids in counterfactual performance—despite not explicitly training for counterfactual performance.

Case 2: Intervention indices mismatch In this section, we include more results in the more practical scenario where we have knowledge of the number of the intervened nodes but they are not necessarily the last few nodes in the latent SCM. This experiment is related to our canonical ILD theory, i.e., that there exists a canonical counterfactual

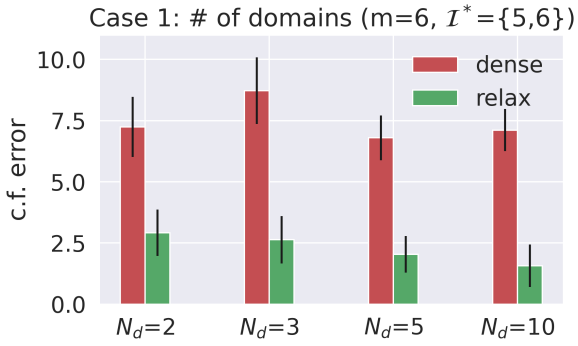
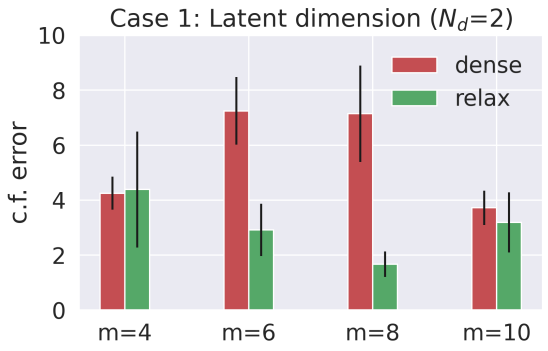
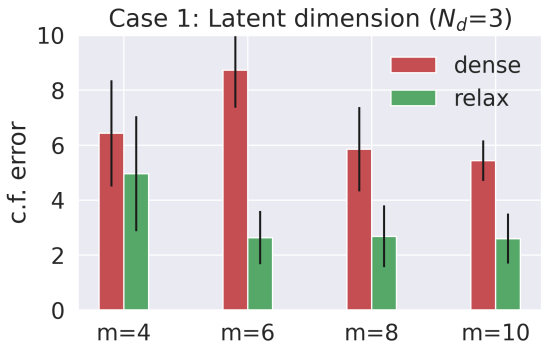


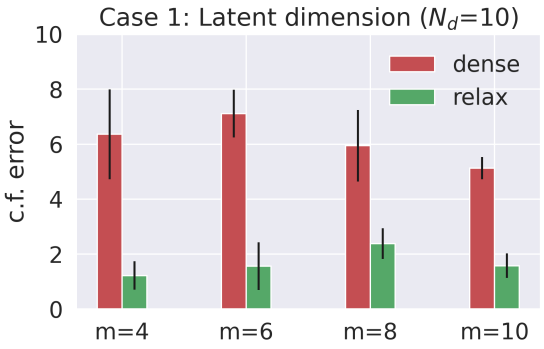
Figure 6: Case 1: Test counterfactual error with different number of domains. Here we investigate how the number of domains affects the performance gap between *ILD-Dense* and *ILD-Relax-Can*. We observe that the gap is not sensitive to the number of domains.



(a) $N_d = 2$



(b) $N_d = 3$



(c) $N_d = 10$

Figure 7: Case 1: Test counterfactual error with different dimension. We investigate how our algorithm scales with dimension. We observe that *ILD-Relax-Can* is significantly better than *ILD-Dense* in 9 out of 12 cases, and we also notice that there 3 cases where their performance is close to that of each other. Here the intervention set contains the last two nodes. For example, when $m = 4, \mathcal{I} = \{3, 4\}$, and when $m = 10, \mathcal{I} = \{9, 10\}$.

model (where the intervened nodes are the last ones) corresponding to any true non-canonical ILD that has the same sparsity. As a starting point, we first illustrate the existence of a canonical model we try to find in Figure 11.

To investigate the effect of different indices of the intervened nodes, in Figure 9, we change the true intervention set $\mathcal{I}^*$ while keeping the number of intervened nodes $|\mathcal{I}^*|$ the same. We observe that *ILD-Relax-Can* is consistently better

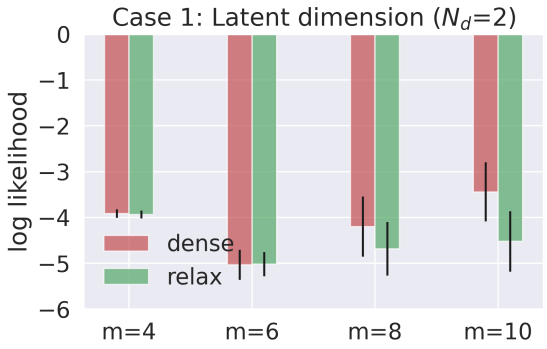


Figure 8: Case 1: Lowest validation log likelihood (same as when we report the test counterfactual error) when testing different dimension with $N_d = 2$. We observe that the likelihood gap between *ILD-Relax-Can* and *ILD-Dense* is largest when $m = 10$.

Table 3: Case 1: Test counterfactual error and validation log likelihood for each seed when $m = 10, N_d = 2$. We observe that the log likelihood of *ILD-Dense* tends to be much lower when it has a larger counterfactual error than that of *ILD-Dense*.

	Seed	0	1	2	3	4	5	6	7	8	9
Counterfactual error	*ILD-Relax-Can*	4.625	0.111	0.120	0.072	4.572	10.617	4.360	6.809	0.099	0.479
	ILD-Dense	23.821	0.611	2.178	5.823	4.779	0.694	0.487	1.653	3.170	6.365
Log likelihood	*ILD-Relax-Can*	-6.873	-7.066	-5.672	-4.637	-0.572	-3.261	-6.062	-4.552	-1.367	-5.170
	ILD-Dense	-4.034	-6.434	-5.679	-4.197	0.711	-1.908	-4.180	-2.413	-1.483	-4.796

Table 4: Case 1: Test counterfactual error for each seed when $m = 4, N_d = 2$. *ILD-Relax-Can* is better than *ILD-Dense* except when seed is 0. However, there is a significant failure for *ILD-Relax-Can* with seed 0.

Seed	0	1	2	3	4	5	6	7	8	9
ILD-Relax-Can	23.790	2.309	1.747	3.180	1.265	0.864	0.779	0.227	3.325	6.362
ILD-Dense	3.321	3.435	2.838	4.209	5.356	6.456	1.615	2.165	5.195	7.937

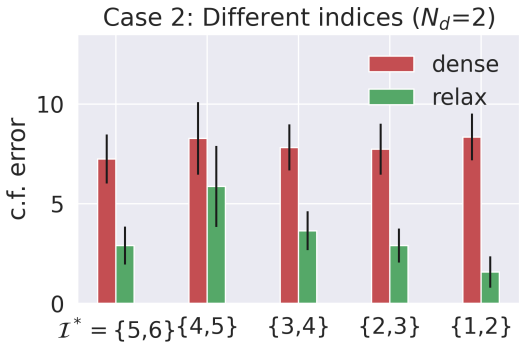
Table 5: Case 1: Test counterfactual error for each seed when $m = 4, N_d = 3$. *ILD-Relax-Can* is better than *ILD-Dense* with seed $1, 2, 3, 5, 6, 7, 8$.

Seed	0	1	2	3	4	5	6	7	8	9
ILD-Relax-Can	23.821	0.611	2.178	5.823	4.779	0.694	0.487	1.653	3.170	6.365
ILD-Dense	24.472	3.658	2.925	5.785	3.260	5.795	3.878	4.560	4.009	5.965

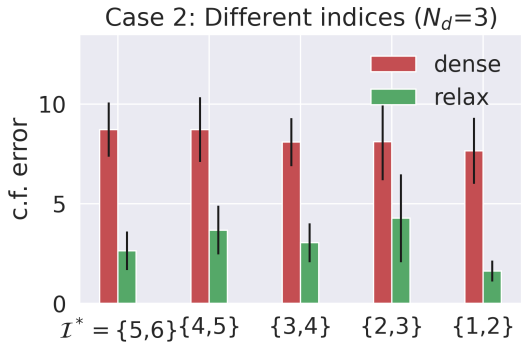
Table 6: Case 2: Test counterfactual error and validation log likelihood for each seed when $N_d = 2$ and $\mathcal{I} = \{4, 5\}$. When seed is 5, the error of *ILD-Relax-Can* is much larger than that of *ILD-Dense*. In the meanwhile, we notice that the log likelihood of *ILD-Relax-Can* is much lower than that of *ILD-Dense* which indicates *ILD-Relax-Can* fails to fit the observed distribution well. When seed is 6, there is also a gap in log likelihood. But both models perform very badly in terms of counterfactual error in this case, and we conjecture this results from a very hard dataset.

	Seed	0	1	2	3	4	5	6	7	8	9
Counterfactual error	*ILD-Relax-Can*	1.395	0.862	1.338	0.193	7.557	12.422	21.762	3.879	2.352	0.479
	ILD-Dense	8.610	5.979	4.134	2.983	9.795	4.719	24.232	5.327	8.497	8.500
Log likelihood	*ILD-Relax-Can*	-4.441	-5.737	-4.448	-5.504	-4.393	-3.376	-5.187	-5.073	-4.033	-4.102
	ILD-Dense	-4.170	-5.632	-4.316	-5.458	-4.174	-2.181	-4.052	-5.010	-5.270	-4.302

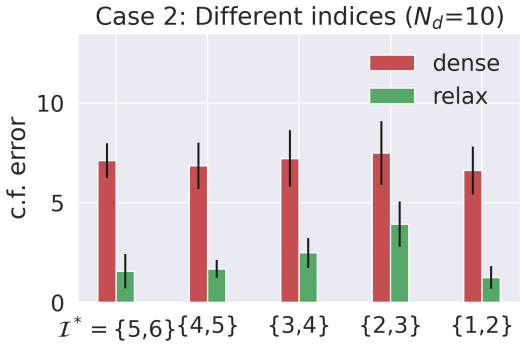
than *ILD-Dense* regardless of which nodes are intervened except for one case. When the number of domains is 2 and $\mathcal{I}^* = \{4, 5\}$, we find the gap is much smaller mainly because *ILD-Relax-Can* fails to fit the observed distribution in one case as shown in Table 6. We then test the effect of the number of domains with different latent dimensions in Figure 10. We observe that our model performs consistently well with different numbers of domains and latent dimensions.



(a) $N_d = 2$



(b) $N_d = 3$



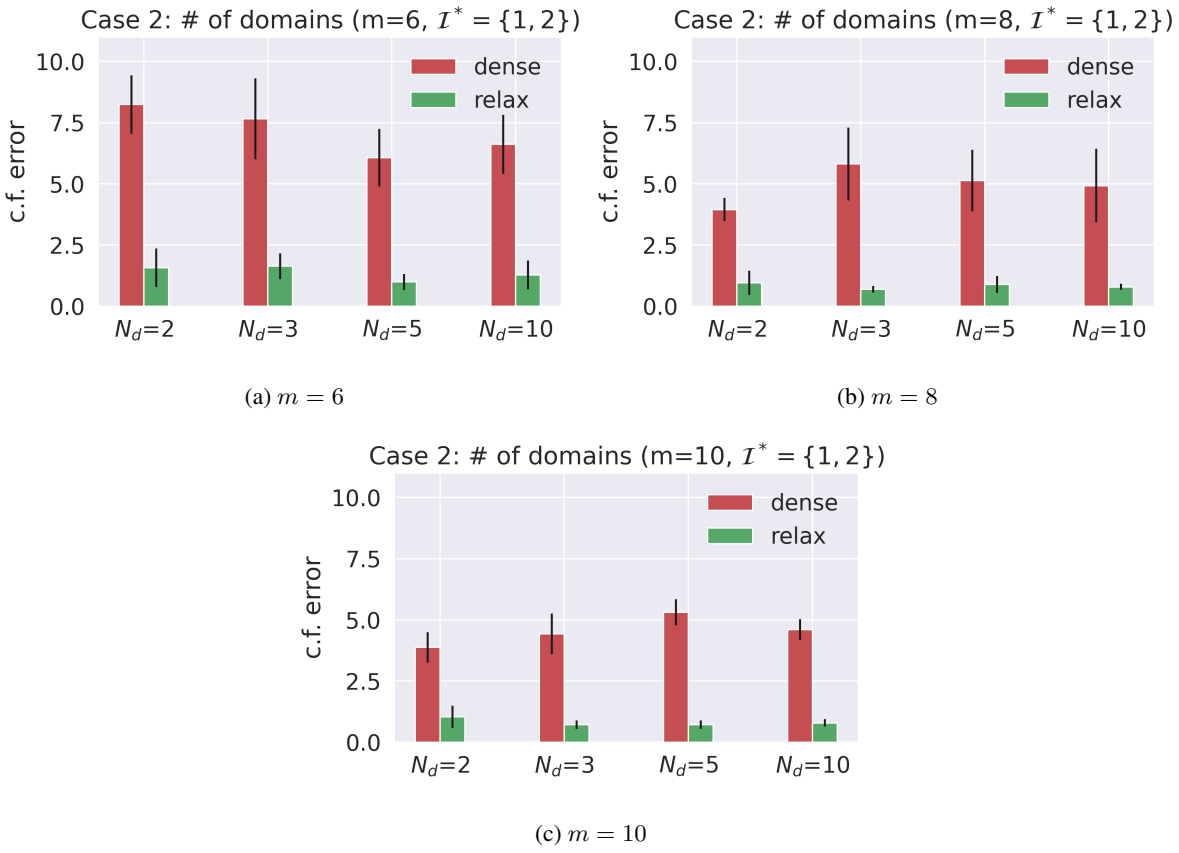
(c) $N_d = 10$

Figure 9: Case 2: Test counterfactual error with different indices. Here we observe that *ILD-Relax-Can* performs consistently better than *ILD-Dense*. When $N_d = 2$ and $\mathcal{I} = \{4, 5\}$, the performance of *ILD-Relax-Can* gets relatively higher because it fails significantly in one case as shown in Table 6.

Even though we do not know the specific nodes being intervened on, similar to case 1, we show that sparse constraint leads to better counterfactual estimation.

Case 3: Intervention set size mismatch In this section, we include more results in the most difficult cases where we have no knowledge of the dataset. To investigate what will happen if there is a mismatch of the number of intervened nodes between the true model and the approximation, i.e., $|\mathcal{I}| \neq |\mathcal{I}^*|$, we first change $\mathcal{I}^*$ while keeping the model unchanged, i.e., $\mathcal{I}$ is fixed. As shown in Figure 12, the performance gap between *ILD-Relax-Can* and *ILD-Dense* become smaller as the dataset becomes less sparse while *ILD-Relax-Can* outperforms *ILD-Dense* in all cases. We then change $\mathcal{I}$ while keeping $\mathcal{I}^*$ unchanged. As shown in Figure 13, the performance of *ILD-Relax-Can* approaches to that of *ILD-Dense* as we increase $|\mathcal{I}|$. A somewhat surprising result is that *ILD-Relax-Can* has the lowest counterfactual error when $|\mathcal{I}| = 1$. However, as we check data fitting in Figure 14, we can tell *ILD-Relax-Can* fails to fit the observed distribution in this case which helps explain the higher counterfactual error in this case. We conjecture there are two reasons for this. First, we cannot rely on the counterfactual estimation when the observed distribution is not fitted. Second, our theory does not guarantee the existence of a distributionally and counterfactually equivalent canonical model in those cases as we are using a model that is more sparse than the ground truth dataset.

In summary, we observe that while *ILD-Relax-Can* always tends to get a lower counterfactual error, in the cases where our model is more sparse than ground truth, the data fitting performance of *ILD-Relax-Can* would drop more significantly. We believe this could also be a good indicator of whether we find a reasonable $|\mathcal{I}|$.

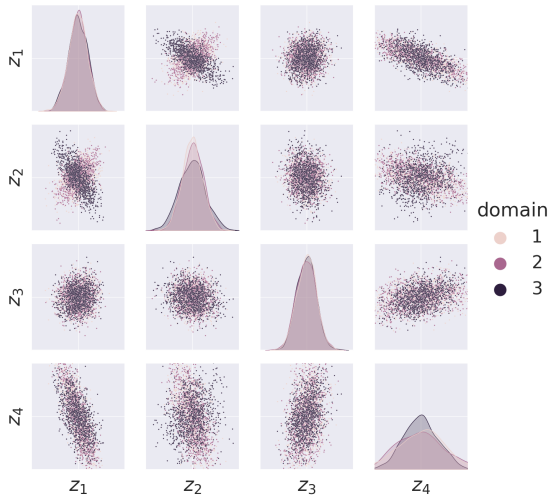


(a) $m = 6$

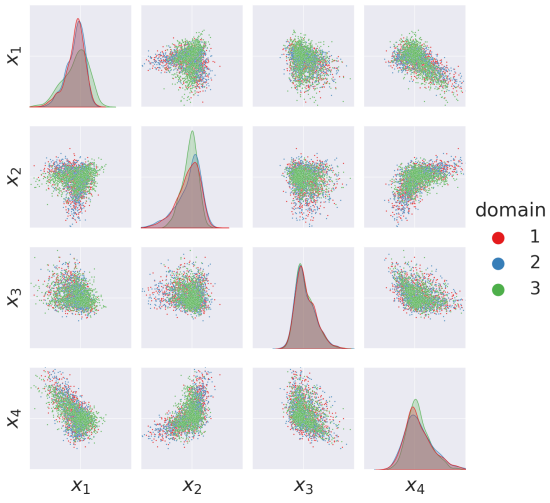
(b) $m = 8$

(c) $m = 10$

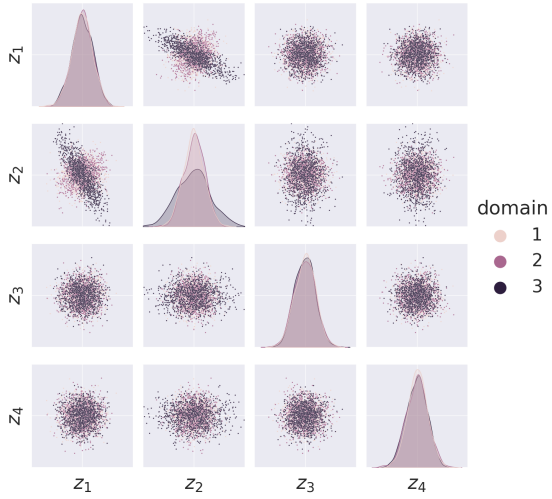
Figure 10: Case 2: Test counterfactual error with different number of domains when $\mathcal{I} = \{1, 2\}$. *ILD-Relax-Can* performs consistently well with different number of domains and latent dimension.



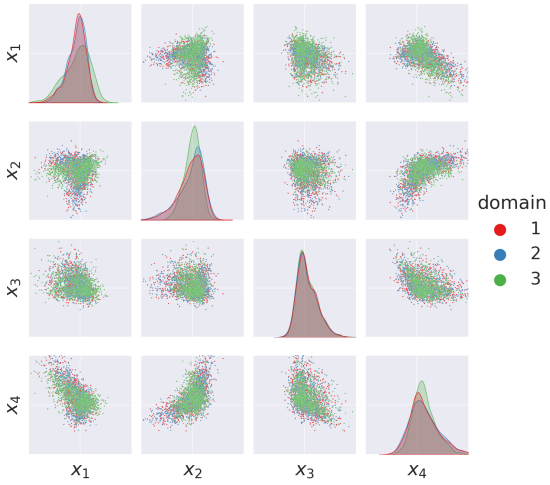
(a) Step 0: $f^{(0)} = f_d$



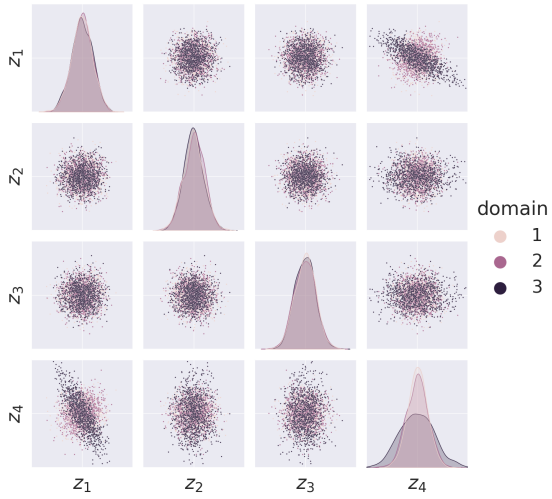
(b) Step 0: $g^{(0)} = g$



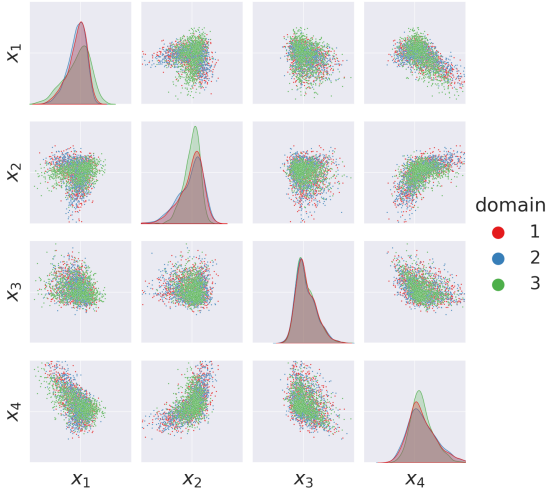
(c) Step 1: $f^{(1)} = f_1^{-1} \circ f_d$



(d) Step 1: $g^{(1)} = g \circ f^{-1}$

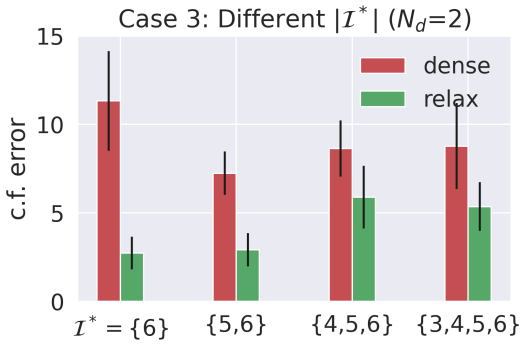


(e) Step 2: $f^{(2)} = h_{1\leftrightarrow 3} \circ f_1^{-1} \circ f_d \circ h_{1\leftrightarrow 3}$

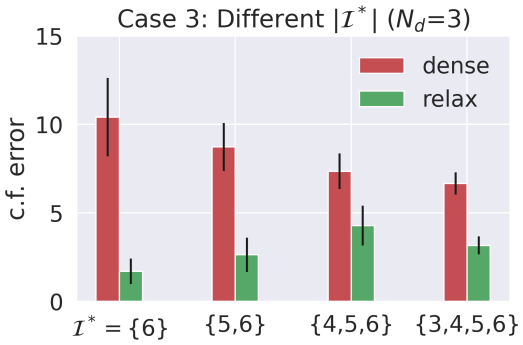


(f) Step 2: $g^{(2)} = g \circ f^{-1} \circ h_{1\leftrightarrow 3}$

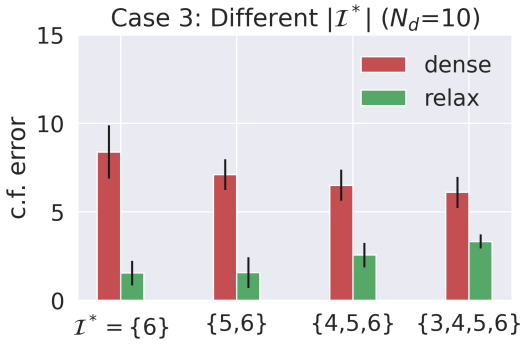
Figure 11: An illustration of the existence of a distributionally and counterfactually equivalent model in canonical form when $m = 4$ and $\mathcal{I} = \{2\}$. $h_{1\leftrightarrow 3}$ represents a swapping matrix. $g^{(2)} \circ f^{(2)}$ is one of the caonical model we try to find. Note that the observed distributions in the right column are always the same while the latent distributions on the left change. In particular, the canonical ILD model on the bottom left has independent distributions for the first three variables and is only the non-identity on the last node.



(a) $N_d = 2$

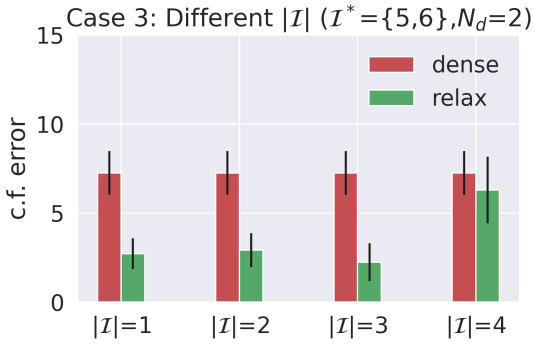


(b) $N_d = 3$

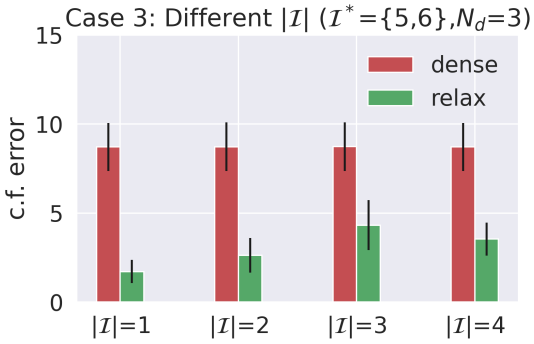


(c) $N_d = 10$

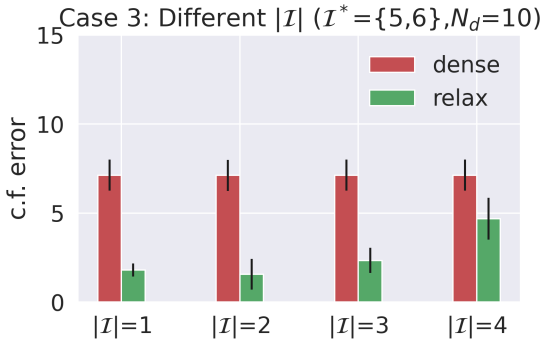
Figure 12: Case 3: Test counterfactual error with different $|\mathcal{I}^*|$ and fixed $|\mathcal{I}| = 2$. The performance of *ILD-Relax-Can* gets worse as the dataset becomes less sparse. But it is still better than *ILD-Dense*. Note that when $|\mathcal{I}| = 2$ and $\mathcal{I}^* = \{6\}$, the ground truth canonical model is still a subset of the models we search over.



(a) $N_d = 2$

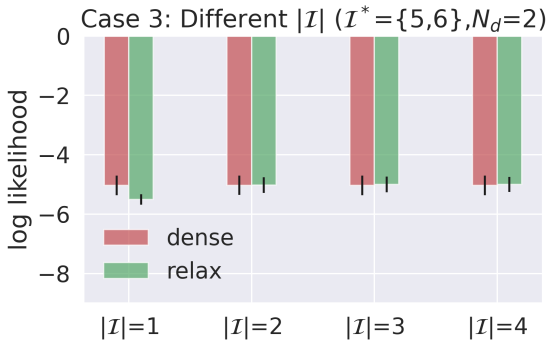


(b) $N_d = 3$

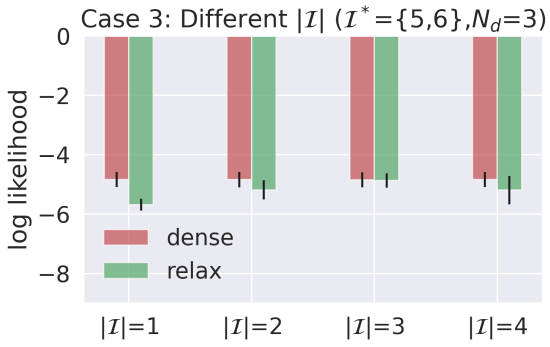


(c) $N_d = 10$

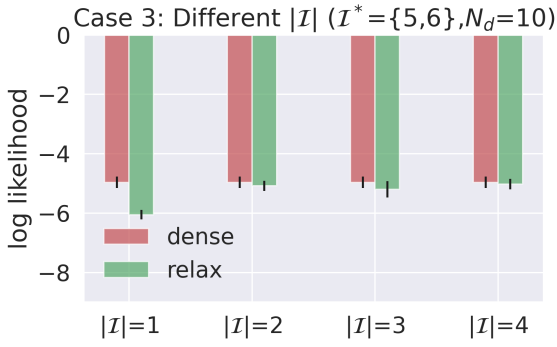
Figure 13: Case 3: Test counterfactual error with different $|\mathcal{I}|$ and fixed $\mathcal{I}^*$. The performance of *ILD-Relax-Can* approaches to that of *ILD-Dense* as we increase $|\mathcal{I}|$.



(a) $N_d = 2$



(b) $N_d = 3$



(c) $N_d = 10$

Figure 14: Case 3: Lowest validation log likelihood with different $|\mathcal{I}|$ and fixed $\mathcal{I}^*$. When $|\mathcal{I}| = 1$, there is a more significant gap between *ILD-Relax-Can* and *ILD-Dense* with all N_d which indicates *ILD-Relax-Can* might fail to fit the observed distribution.

E RMNIST Experiment

The goal for this experiment was to test our setup in a high dimensional setting where we must learn a complex mapping from the observed space to the latent space. To do this, we relax the invertibility requirement of the ILD models to allow for pseudoinvertibility. Further, this relaxation allows for the latent space to have a much smaller dimensionality than the observed space. We test our models on a rotated version of the MNIST dataset Deng [2012] where each of the 5 domains has a domain-specific rotation applied to all digits in that domain's dataset. The rotations used are $\theta_d \in \{0, 15, 30, 45, 60\}$ degree counterclockwise rotations. The objective for this problem is to see if we can train a model to produce domain counterfactuals (e.g., rotate a digit from 15 degrees to 60 degrees) while keeping non-domain-specific information unchanged (e.g., maintaining the same look of a 4throughout the domain counterfactual). While here the ground truth SCM is unknown, we conjecture that this disentangling should be possible as the domain intervention happens independently of the digit label, line thickness, etc. As we will discuss next, the models are only being trained to align their generated data distribution with the observed data distribution – which is not specific to producing counterfactuals.

E.1 RMNIST Experiment Details

Model setup This relaxation to pseudo invertibility allows us to modify the ILD models to fit a VAE Kingma and Welling [2013] structure. The overall VAE structure can be seen in Fig. 15, where the variational encoder first projects to the latent space via g^+ to produce the latent encoding $\boldsymbol{z}$, which is then passed to two domain-specific autoregressive models $f^+_{d,\mu}, f^+_{d,\sigma}$ which produce the mean and variance parameters (respectively) of the Gaussian posterior distribution. The decoder of the VAE follows the structure typical ILD structure: $g \circ f_d$. Here, g^+ can be viewed as the pseudoinverse of the observation function g and f_d can be viewed as a pseudoinverse of $f^+_{d,\mu}$ During training, the exogenous noise variable ϵ is then found via sampling from the posterior distribution ($\epsilon \sim \mathcal{N}(\mu_d, \sigma_d)$) which can be viewed as a stochastic SCM, however, to reduce noise when producing counterfactuals, when performing inference the exogenous variable is set to the mean of the latent posterior distribution (i.e. $\epsilon = \mu_d$). In this experiment, g and g^+ follow the β-VAE architecture seen in Higgins et al. [2017], and the structure of the f models is determined by the type of ILD model used (e.g., dense, canonical, or relaxed canonical) and matches that seen in the simulated experiments and visualized in Fig. 4. For the f models which enforce sparsity (i.e. *ILD-Can* and *ILD-Relax-Can*), we use a sparsity level, $|\mathcal{I}|$, of 5. We also introduce an additional baseline, *ILD-Independent*, which has an architecture similar to the *ILD-Dense* baseline, with the exception that the$g$ and g^+ functions are no longer shared across domains. The *ILD-Independent* baseline can be seen as training an independent β-VAE for each domain, where each β-VAE an autoregressive f_{dense} model as it's last (first) layer for the encoder (decoder), respectively.

Training We split the MNIST trainset into 90% training data, 10% validation, and for testing we use the MNIST test set. Within each dataset, we create the domain-specific data by applying a fixed θ_d counterclockwise rotation to all samples within that domain. We train each ILD model for 200K steps using the Adam optimizer Kingma and Ba [2014] with $\beta_1 = 0.5, \beta_2 = 0.999$, and a batch size of 1024. The learning rate for g and g^+ is 10^{-4}, and all f models use 10^{-3}. During training, we calculate two loss terms: a reconstruction loss $\ell_{recon} = |\boldsymbol{x} - \hat{\boldsymbol{x}}|_2^2$ where $\hat{\boldsymbol{x}}$ is the reconstructed image of $\boldsymbol{x}$ and the ℓ_{align} alignment loss which measures the KL-divergence between the posterior distribution $Q_d(\epsilon|\boldsymbol{x})$ and the prior $P(\epsilon)$. Following the β-VAE loss calculation in Higgins et al. [2017], we apply a β_{KLD} upscaling to the alignment loss such that $\ell_{total} = \ell_{recon} + \beta_{KLD} * \ell_{align}$. Through empirical testing, we found $\beta_{KLD} = 1000$ leads to the lowest counterfactual error on the validation datasets across all models; this also matches the β_{KLD} used in Burgess et al. [2018].

E.2 RMNIST Counterfactual Results

We train each ILD model across two latent dimensionality settings ($M \in \{10, 20\}$) and with 20 random seeds. To evaluate the models, we used measured the mean-squared error between the estimated counterfactual and the ground truth counterfactual (where the ground truth counterfactual is generated by applying the d' rotation to the original MNIST digit). As the number of domains grows large, the identity counterfactual ($x_{d \to d'}$ where $d = d'$) becomes diluted compared to the other domain counterfactuals, and as this is significant to ensure invertibility, we control this dilution by scaling the identity counterfactuals by the number of domains, N_d.

The results are shown in Table 2, where it can be seen that *ILD-Relax-Can* has the lowest MSE counterfactual loss amongst all four ILD models for both settings of M. This quantitative result matches the visual result seen in Fig. 3, where the *ILD-Relax-Can* model seems to properly disentangle the domain rotation information from the digit label information – unlike the baseline models which seem to commonly change the digit label during counterfactual. We again note that the training process for all of the models only included the typical VAE invertibility loss (i.e.

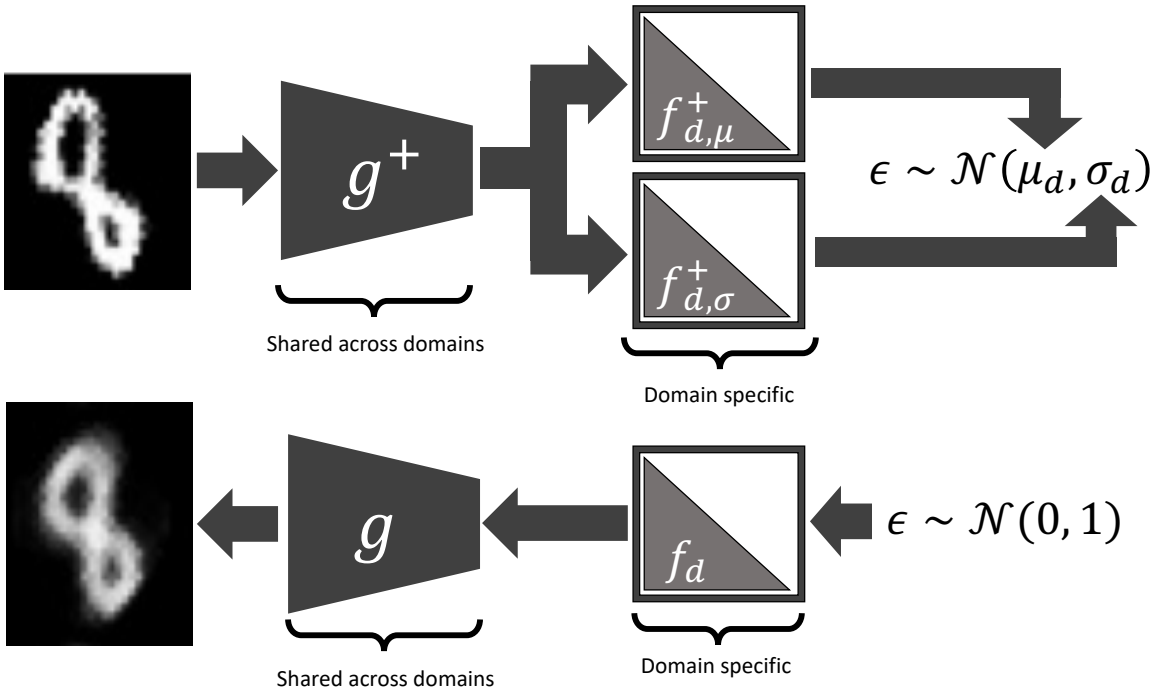


Figure 15: The model structure for the pseudo-invertible ILD model used in RMNIST. The overall structure matches that of a VAE where the encoder (top) first projects to the latent space via g^+ (the pseudoinverse of the observation function g). This latent encoding is then passed to two domain-specific autoregressive models $f^+_{d,\mu}, f^+_{d,\sigma}$ which produce the mean and variance parameters (respectively) of the Gaussian posterior distribution. During training, the exogenous noise variable ϵ is then found via sampling from the posterior distribution ($\epsilon \sim \mathcal{N}(\mu_d, \sigma_d)$) which can be viewed as a stochastic SCM, however, during inference the exogenous variable is set to the mean of the latent posterior distribution (i.e. $\epsilon := \mu_d$) to reduce noise when producing counterfactuals. The decoder (bottom) follows the usual VAE decoder structure, with the exception that the initial linear layer is an autoregressive function of the ϵ input. The structure of all the f models is determined by the type of ILD model used (e.g., dense, canonical, or relaxed canonical) and matches that seen in Fig. 4

reconstruction loss) and latent alignment loss (i.e. the KL-divergence between the latent prior and posterior distributions) and did not specifically include any counterfactual training. Thus, we conjecture the enforcing of sparsity in *ILD-Relax-Can* and *ILD-Can* correctly biased these models in a manner that preserved important non-domain-specific information when performing counterfactuals. Further, we believe the additional flexibility of allowing *ILD-Relax-Can* to learn more shared weights for the non-intervened nodes (i.e. the upper blue triangle in Fig. 4b) allowed for more non-domain-specific information to be used by the *ILD-Relax-Can* model in comparison to *ILD-Can* which has fixed identity weights for the non-intervened nodes.

F Limitations

In this paper, we first prove the existence of distributionally and counterfactually equivalent models. Then we investigate how hard it would be to learn such models in practice when the only objective in the algorithm is to fit the observed distribution. In our extensive simulated experiments and Rotated MNIST experiments, we find that the sparsity constraint inspired by our theory helps the model achieve more accurate counterfactual estimation. From a theoretic side, while our theory proves the existence of canonical ILD models, we have not proven identifiability of the latent causal model or observation function. Indeed, we conjecture that complete identifiability of latent causal models is likely infeasible in our setup except under very strong constraints. A deeper investigation into the conditions for identifiability or proof of non-identifiability would be interesting future directions.

A practical problem we noticed in our simulated experiments is that sometimes the sparse model is harder to fit, i.e., its log-likelihood is worse than the dense model, even if we only consider the cases where the true model is in the model class being optimized (e.g., the sparsity of the model is at least as large as the sparsity of the ground truth

model). We conjecture that this results from a harder loss landscape as we add more constraints to the model. We believe a more careful investigation of the model and algorithm could be an interesting and important future work. For example, if we use a more significantly overparameterized model, there are chances that the training of *ILD-Relax-Can* would become easier. Additionally, the addition of further loss terms could aid in the training of these models, such as, assuming access to some ground truth domain counterfactuals (e.g., the same patient received imaging at multiple hospitals) could be used to penalize our model when it changes latent variables which do not change under the ground truth counterfactuals.

In our experiments, we aimed to test the effects of breaking some of our assumptions (e.g., "what if our model is not strictly invertible"), and while our models still performed better in these cases, there are likely cases where the breaking of our assumptions can cause our models to fail to produce faithful counterfactuals. For example, in a case where there is a very large difference between domains and there is no sparsity in the domain shifts, then it is likely that the constraints constituted by our sparsity assumption will make the sparse models struggle to fit the observed distributions.