
Weakly supervised causal representation learning: Supplementary material

Johann Brehmer*
Qualcomm AI Research[†]
jbrehmer@qti.qualcomm.com

Pim de Haan*
Qualcomm AI Research[†]
QUVA Lab, University of Amsterdam
pim@qti.qualcomm.com

Phillip Lippe
QUVA Lab, University of Amsterdam
p.lippe@uva.nl

Taco Cohen
Qualcomm AI Research[†]
tacos@qti.qualcomm.com

In the following we provide additional results and details that did not fit into our main paper. In Appendix A we provide precise definitions and a complete proof of our identifiability theorem. We then discuss the assumptions underlying this result and their generalization in Appendix B. Appendix C covers implicit latent causal models (ILCMs) and their training, while Appendix D provides details for our experiments. In Appendix E we describe explicit latent causal models (ELCMs) and our experiments with them. Finally, in Appendix F we discuss the potential societal impact of our work.

A Identifiability result

A.1 Definitions

Here we define objects and relations that were not formally defined in the main body of the paper, but are necessary to make Thm. 1 precise and to prove it.

We use the following notation:

- $[n] = \{1, \dots, n\}$
- $\text{pa}_i^{\mathcal{C}} \subseteq [n]$ the set of parent nodes of node i in graph $\mathcal{G}(\mathcal{C})$.
- $\text{desc}_i^{\mathcal{C}} \subseteq [n]$ the set of descendant nodes of node i in graph $\mathcal{G}(\mathcal{C})$, excluding i itself.
- $\text{anc}_i^{\mathcal{C}} \subseteq [n]$ the set of ancestor nodes of node i in graph $\mathcal{G}(\mathcal{C})$, excluding i itself.
- $\text{nonanc}_i^{\mathcal{C}} = [n] \setminus (\text{anc}_i^{\mathcal{C}} \cup \{i\})$ the set of non-ancestor nodes of node i in graph $\mathcal{G}(\mathcal{C})$, excluding i itself.
- Given measure p on space A and measurable function $f : A \rightarrow B$, f_*p is the push-forward measure on B .

We describe causal structure with SCMs.

Definition 4 (Structural causal model (SCM)). *An SCM is a tuple $\mathcal{C} = \langle \mathcal{Z}, \mathcal{E}, F, p_{\mathcal{E}} \rangle$ consisting of the following:*

- domains $\mathcal{Z} = \mathcal{Z}_1 \times \dots \times \mathcal{Z}_n$ of causal (endogenous) variables z_1, \dots, z_n ;
- domains $\mathcal{E} = \mathcal{E}_1 \times \dots \times \mathcal{E}_n$ of noise (exogenous) variables $\epsilon_1, \dots, \epsilon_n$;
- a directed acyclic graph $\mathcal{G}(\mathcal{C})$, whose nodes are the causal variables and edges represent causal relations between the variables;
- causal mechanisms $F = \{f_1, \dots, f_n\}$ with $f_i : \mathcal{E}_i \times \prod_{j \in \text{pa}_i} \mathcal{Z}_j \rightarrow \mathcal{Z}_i$; and

*Equal contribution

[†]Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

- a probability measure $p_{\mathcal{E}}(\epsilon) = p_{\mathcal{E}_1}(\epsilon_1) p_{\mathcal{E}_2}(\epsilon_2) \dots p_{\mathcal{E}_n}(\epsilon_n)$ with full support that admits a continuous density.

Additionally, we assume that $\forall i, \forall z_{\mathbf{pa}_i}, f_i(\cdot, z_{\mathbf{pa}_i}) : \mathcal{E}_i \rightarrow \mathcal{Z}_i$ is a diffeomorphism.

We will need to reason about vectors being “equal up to permutation and elementwise reparameterizations”. We formalize this in the following definition:

Definition 5 (ψ -diagonal). *Let $\psi : [n] \rightarrow [n]$ be a bijection (that is, a permutation). Let $\varphi : \prod_{i=1}^n X_i \rightarrow \prod_{i=1}^n Y_i$ be a function between product spaces. Then φ is ψ -diagonal if there exist functions, called components, $\varphi_i : X_i \rightarrow Y_{\psi(i)}$ such that $\forall i, \forall x, \varphi(x_1, \dots, x_i, \dots, x_n)_{\psi(i)} = \varphi_i(x_i)$.*

This lets us define isomorphisms between SCMs:

Definition 6 (Isomorphism of SCMs). *Let $\mathcal{C} = \langle \mathcal{Z}, \mathcal{E}, F, p_{\mathcal{E}} \rangle$ and $\mathcal{C}' = \langle \mathcal{Z}', \mathcal{E}', F', p'_{\mathcal{E}} \rangle$ be SCMs. An isomorphism $\varphi : \mathcal{C} \rightarrow \mathcal{C}'$ consists of*

1. a graph isomorphism $\psi : \mathcal{G}(\mathcal{C}) \rightarrow \mathcal{G}(\mathcal{C}')$ that tells us how to identify corresponding variables in the two models and which preserves parents: $\mathbf{pa}_{\psi(i)}^{\mathcal{C}'} = \psi(\mathbf{pa}_i^{\mathcal{C}})$ and
2. ψ -diagonal diffeomorphisms for noise and endogenous variables that tell us how to reparameterize them $\varphi_{\mathcal{E}} : \mathcal{E} \rightarrow \mathcal{E}'$ and $\varphi_{\mathcal{Z}} : \mathcal{Z} \rightarrow \mathcal{Z}'$, where $\varphi_{\mathcal{E}}$ must be measure preserving $p_{\mathcal{E}'} = \varphi_{\mathcal{E}*} p_{\mathcal{E}}$. For notational simplicity, we will drop the subscript in $\varphi_{\mathcal{Z}}$ and use the symbol φ to refer both to the SCM isomorphism and the noise isomorphism.

The elementwise diffeomorphisms are required to make the following diagrams commute $\forall i, i' = \psi(i)$:

$$\begin{array}{ccc} \mathcal{Z}_{\mathbf{pa}_i} \times \mathcal{E}_i & \xrightarrow{(\varphi_{\mathbf{pa}_i}, \varphi_{\mathcal{E}, i})} & \mathcal{Z}'_{\mathbf{pa}_{i'}} \times \mathcal{E}'_{i'} \\ \downarrow f_i & & \downarrow f'_{i'} \\ \mathcal{Z}_i & \xrightarrow{\varphi_i} & \mathcal{Z}'_{i'} \end{array} \quad (5)$$

Intuitively, this says that if we apply a causal mechanism f_i and then reparameterize the causal variable i using φ_i , we get the same thing as first reparameterizing the parents and noise variable of variable i , and then applying the causal mechanism $f'_{i'}$.

To reason about interventions, we equip SCMs with intervention distributions in the following definition.

Definition 7 (Intervention structural causal model (ISCM)). *An intervention structural causal model (ISCM) is a tuple $\mathcal{D} = \langle \mathcal{C}, \mathcal{I}, p_{\mathcal{I}} \rangle$ of*

1. an acyclic SCM $\mathcal{C} = \langle \mathcal{Z}, \mathcal{E}, F, p_{\mathcal{E}} \rangle$ that admits a faithful distribution, meaning that conditional independence of causal variables z implies d-separation [1].
2. a set \mathcal{I} of interventions on \mathcal{C} , where each intervention $(I, (\tilde{f}_i)_{i \in I}) \in \mathcal{I}$ consist of
 - (a) a subset $I \subset \{1, \dots, n\}$ of the causal variables, called the intervention target set, and
 - (b) for each $i \in I$, a new causal mechanism $\tilde{f}_i : \mathcal{E}_i \rightarrow \mathcal{Z}_i$ which replaces the original mechanism and which does not depend on the parents.

We define intervention set \mathcal{I} to be atomic if the number of targeted variables is one or zero.
3. a probability measure $p_{\mathcal{I}}$ over \mathcal{I} .

We can extend the notion of isomorphism from SCMs to ISCMs.

Definition 8 (Isomorphism of ISCMs). *Let $\mathcal{D} = \langle \mathcal{C}, \mathcal{I}, p_{\mathcal{I}} \rangle$ and $\mathcal{D}' = \langle \mathcal{C}', \mathcal{I}', p'_{\mathcal{I}} \rangle$ be ISCMs. An ISCM isomorphism is an SCM isomorphism $\varphi : \mathcal{C} \rightarrow \mathcal{C}'$ with underlying graph isomorphism $\psi : \mathcal{G}(\mathcal{C}) \rightarrow \mathcal{G}(\mathcal{C}')$ and a ψ -diagonal diffeomorphism $\tilde{\varphi}_{\mathcal{E}} : \mathcal{E} \rightarrow \mathcal{E}'$ such that*

- the graph isomorphism ψ induces a bijection of intervention sets

$$\psi_{\mathcal{I}} : \mathcal{I} \rightarrow \mathcal{I}' : (I, (\tilde{f}_i)_{i \in I}) \mapsto (\psi(I), (\tilde{f}'_{i'})_{i' \in \psi(I)})$$

- for each intervention $(I, (\tilde{f}_i)_{i \in I}) \in \mathcal{I}$, and each intervened on variable $i \in I$, the following diagram commutes:

$$\begin{array}{ccc}
 \mathcal{E}_i & \xrightarrow{\tilde{\varphi}_{\mathcal{E},i}} & \mathcal{E}'_{\psi(i)} \\
 \downarrow \tilde{f}_i & & \downarrow \tilde{f}'_{\psi(i)} \\
 \mathcal{Z}_i & \xrightarrow{\varphi_i} & \mathcal{Z}'_{\psi(i)}
 \end{array} \tag{6}$$

- $\tilde{\varphi}_{\mathcal{E}}$ is measure preserving, i. e. $p_{\mathcal{E}'} = (\tilde{\varphi}_{\mathcal{E}})_* p_{\mathcal{E}}$.
- the bijection $\psi_{\mathcal{I}} : \mathcal{I} \rightarrow \mathcal{I}'$ preserves the distribution over interventions: $\psi_* p_{\mathcal{I}} = p'_{\mathcal{I}'}$.

Latent Causal Models (LCMs), defined in Def. 1, add a map to the data space to an ILCM. We can lift ISCM isomorphisms to LCM isomorphisms by requiring that these decoders must respect the ISCM isomorphism.

Definition 9 (Isomorphism of LCMs). *Let $\mathcal{M} = \langle \mathcal{C}, \mathcal{X}, g, \mathcal{I}, p_{\mathcal{I}} \rangle$ and $\mathcal{M}' = \langle \mathcal{C}', \mathcal{X}', g', \mathcal{I}', p'_{\mathcal{I}'} \rangle$ be LCMs with identical observation space $\mathcal{X} = \mathcal{X}'$. An LCM isomorphism is an ISCM isomorphism $\varphi : \mathcal{D} \rightarrow \mathcal{D}'$ such that the decoders respect the SCM isomorphism, so this diagram must commute:*

$$\begin{array}{ccc}
 \mathcal{Z} & \xrightarrow{\varphi} & \mathcal{Z}' \\
 \searrow g & & \swarrow g' \\
 & \mathcal{X} &
 \end{array} \tag{7}$$

Remark 1. By defining objects and isomorphisms, we have defined a groupoid of SCMs, a groupoid of ISCMs and a groupoid of LCMs, as the isomorphisms are composed and inverted in an obvious way.

Definition 10 (Equivalence). *We call two SCMs, ISCMs, or LCMs equivalent if an isomorphism exists between them.*

Informally, two SCMs, ISCMs, or LCMs are equivalent if there is a ψ -diagonal map between their causal variables (i. e. the causal variables are equal up to permutation and elementwise diffeomorphisms), there is a ψ -diagonal map between their noise encodings, and all other structure (decoders, intervention sets, intervention distributions) is compatible with these reparameterizations.

Next, we define the solution function of an SCM or ISCM, which maps from noise variables to causal variables by repeatedly applying the causal mechanisms.

Definition 11 (Solution). *Given an ISCM $\mathcal{D} = \langle \mathcal{C}, \mathcal{I}, p_{\mathcal{I}} \rangle$, the solution function $s : \mathcal{E} \rightarrow \mathcal{Z}$ is the unique function such that for all $i \in [n]$, the following diagram commutes [2]*

$$\begin{array}{ccc}
 & \mathcal{E} & \\
 (s_{\text{pa}_i}, \text{id}_{\mathcal{E}_i}) \swarrow & & \searrow s_i \\
 \mathcal{Z}_{\text{pa}_i} \times \mathcal{E}_i & \xrightarrow{f_i} & \mathcal{Z}_i
 \end{array}$$

In equations, we have that $s(\epsilon)_i = f(\epsilon_i; s(\epsilon_{\text{pa}_i}))$. Similarly, intervention $(I, (\tilde{f}_i)_{i \in I}) \in \mathcal{I}$ yields a solution function $\tilde{s}_I : \mathcal{E} \rightarrow \mathcal{Z}$ with the modified causal mechanisms.

For example, with two variables with $z_1 \rightarrow z_2$, the solution is given by:

$$s : \mathcal{E} \rightarrow \mathcal{Z} : \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \end{pmatrix} \mapsto \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} f_1(\epsilon_1) \\ f_2(\epsilon_2, f_1(\epsilon_1)) \end{pmatrix}.$$

Since we require causal mechanisms to be pointwise diffeomorphic, the solution function is a diffeomorphism as well.

Pushing the noise distribution of an SCM through the solution function finally gives us the (observable) distribution entailed by an SCM or ISCM. In an ISCM or LCM we can define several other (observational or interventional) distributions.

$$\begin{array}{ccc}
\epsilon, \tilde{\epsilon} & \xrightarrow{\varphi_{\mathcal{E}}, \tilde{\varphi}_{\mathcal{E}}} & \epsilon', \tilde{\epsilon}' \\
s, \tilde{s}_I \downarrow & & \downarrow s', \tilde{s}'_I \\
z, \tilde{z} & \xrightarrow{\varphi_{\mathcal{Z}}, \varphi_{\mathcal{Z}}} & z', \tilde{z}' \\
g, g \downarrow & & \downarrow g', g' \\
x, \tilde{x} & \xlongequal{\quad\quad\quad} & x, \tilde{x}
\end{array}$$

Figure 7: An illustration of the spaces and maps in our definitions and proof. When LCMs $\mathcal{M}, \mathcal{M}'$ are isomorphic, all squares in the diagram should commute. Additionally, all maps should preserve the weakly supervised distributions on the variables and all horizontal maps should be ψ -diagonal. Note that the latent variables (ϵ, z) can differ up to a diffeomorphism, but the x variables are actually observed, so must be identically equal. From that equality, the other horizontal maps are uniquely defined.

Definition 12 (Distributions). *Given an LCM $\mathcal{M} = \langle \mathcal{C}, \mathcal{X}, g, \mathcal{I}, p_{\mathcal{I}} \rangle$, we have the following generative process:*

$$\begin{aligned}
\epsilon &\sim p_{\mathcal{E}}, & z &= s(\epsilon), & x &= g(z), & e &= s^{-1}(z) \\
I &\sim p_{\mathcal{I}}, & \tilde{\epsilon} &\sim \tilde{p}_{\tilde{\mathcal{E}}}(\tilde{\epsilon} \mid \epsilon, I), & \tilde{z} &= \tilde{s}_I(\tilde{\epsilon}), & \tilde{x} &= g(\tilde{z}), & \tilde{e} &= s^{-1}(\tilde{z}). \quad (8)
\end{aligned}$$

where $p(\tilde{\epsilon}_i \mid \epsilon_i, i \in I) = p_{\mathcal{E}_i}(\tilde{\epsilon}_i)$ and $p(\tilde{\epsilon}_i \mid \epsilon_i, i \notin I) = \delta(\tilde{\epsilon}_i \mid \epsilon_i)$ is the Dirac measure.

Then we define the following weakly supervised distributions:

- The weakly supervised noise distribution with interventions: $p_{\mathcal{C}}^{\mathcal{E}, \mathcal{I}}(\epsilon, \tilde{\epsilon}, I)$.
- The weakly supervised causal distribution with interventions: $p_{\mathcal{C}}^{\mathcal{Z}, \mathcal{I}}(z, \tilde{z}, I)$.
- The weakly supervised observational distribution with interventions: $p_{\mathcal{M}}^{\mathcal{X}, \mathcal{I}}(x, \tilde{x}, I)$.

These distributions are given by appropriate pushforwards of the noise distributions through the transformations in Eq. (8).

By marginalizing over I , we get $p_{\mathcal{C}}^{\mathcal{E}}, p_{\mathcal{C}}^{\mathcal{Z}}, p_{\mathcal{C}}^{\mathcal{E}}, p_{\mathcal{M}}^{\mathcal{X}}$ respectively.

The relationships between all the maps can be found in Fig. 7.

A.2 Identifiability proof

First, we prove two auxiliary lemmata.

Lemma 1. *Let $f : [0, 1] \rightarrow [0, 1]$ be differentiable and Lebesgue measure preserving. Then either $f(x) = x$ or $f(x) = 1 - x$.*

Proof. We follow the celebrated proof from Stack Exchange user zhw [3]. Let λ be the Lebesgue measure. Measure preservation means that for any measurable subset $U \subseteq [0, 1]$, $\lambda(U) = \lambda(f^{-1}(U))$.

First, note that f is surjective, because otherwise the image of f is a proper subinterval $[a, b] \subsetneq [0, 1]$ and $\lambda(f^{-1}([a, b])) = \lambda([0, 1]) = 1 > \lambda([a, b]) = b - a$, which contradicts measure-preservation.

Define the open ball $B(x, r) = \{y \in [0, 1] \mid |y - x| < r\}$. Suppose that $f'(0) = 0$ for some $x \in [0, 1]$. Then there exists an $r > 0$ such that $f(B(x, r)) \subseteq B(f(x), r/4)$, and thus $B(x, r) \subseteq f^{-1}(B(f(x), r/4))$. Therefore, $r \leq \lambda(B(x, r)) \leq \lambda(f^{-1}(B(f(x), r/4)))$, while $\lambda(B(f(x), r/4)) \leq 2 \cdot r/4 = r/2$, contradicting measure preservation. Hence $f'(x) \neq 0$ on $[0, 1]$.

By the Darboux theorem, f' is either strictly positive or strictly negative on the interval and thus f is either strictly increasing or decreasing and thus a bijection. Assume that it is strictly increasing, then $\forall x \in [0, 1], x = \lambda([0, x]) = \lambda(f^{-1}(f([0, x]))) = \lambda(f([0, x])) = f(x) - f(0) = f(x)$. Similarly, if it is strictly decreasing, we find $f(x) = 1 - x$. \square

Lemma 2. *Let $A = C = \mathbb{R}$ and $B = \mathbb{R}^n$. Let $f : A \times B \rightarrow C$ be differentiable. Define differentiable measures p_A on A and p_C on C . Let $\forall b \in B, f(\cdot, b) : A \rightarrow C$ be measure-preserving. Then f is constant in B .*

Proof. Let $P_A : A \rightarrow [0, 1]$, $P_C : C \rightarrow [0, 1]$ be the diffeomorphic cumulative density functions. Then P_A^{-1} and P_C^{-1} are measure-preserving maps from the uniform distribution on $[0, 1]$. Now write $g : [0, 1] \times B \rightarrow [0, 1] : (z, b) \mapsto P_C(f(P_A^{-1}(z), b))$ such that this diagram of measure-preserving differentiable maps commutes:

$$\begin{array}{ccc} A & \xrightarrow{f(\cdot, b)} & C \\ P_A \searrow & & \nearrow P_C^{-1} \\ & [0, 1] \xrightarrow{g(\cdot, b)} [0, 1] & \end{array}$$

Then g is differentiable and $\forall b \in B$ measure-preserving $[0, 1] \rightarrow [0, 1]$. By the previous Lemma 1, the only differentiable measure-preserving functions $[0, 1] \rightarrow [0, 1]$ are id and $1 - \text{id}$. As g is continuous in B , it can not vary between id and $1 - \text{id}$ and thus g , and consequently f are constant in B . \square

We can interpret this lemma in terms of statistical independence. Starting from a product measure on $A \times B$, the requirements of the lemma correspond to $a \perp\!\!\!\perp b$ and $c \perp\!\!\!\perp b$. The lemma thus defines a sense in which for real-valued variables, statistical independence implies functional independence (the converse is always true).

Now in the remainder of this subsection, we prove the main theorem.

Theorem 1 (Identifiability of \mathbb{R} -valued LCMs from weak supervision). *Let $\mathcal{M} = \langle \mathcal{C}, \mathcal{X}, g, \mathcal{I}, p_{\mathcal{I}} \rangle$ and $\mathcal{M}' = \langle \mathcal{C}', \mathcal{X}', g', \mathcal{I}', p'_{\mathcal{I}'} \rangle$ be LCMs with the following properties:*

- *The SCMs \mathcal{C} and \mathcal{C}' both consist of n real-valued endogeneous variables, i. e. $\mathcal{E}_i = \mathcal{Z}_i = \mathcal{Z}'_i = \mathcal{E}'_i = \mathbb{R}$.*
- *The intervention sets \mathcal{I} and \mathcal{I}' consist of the empty intervention and all atomic interventions, $\mathcal{I} = \{\emptyset, \{z_0\}, \dots, \{z_n\}\}$ and similar for \mathcal{I}' .*
- *The intervention distribution $p_{\mathcal{I}}$ and $p'_{\mathcal{I}'}$ have full support.*

Then the following two statements are equivalent:

1. *The weakly supervised distributions entailed by the LCMs are equal, $p_{\mathcal{M}}(x, \tilde{x}) = p_{\mathcal{M}'}(x, \tilde{x})$.*
2. *The LCMs are equivalent, $\mathcal{M} \sim \mathcal{M}'$.*

Proof. “(2) \Rightarrow (1)”: If the LCMs are equivalent, then the fact that $\varphi_{\mathcal{E}}$ and $\tilde{\varphi}_{\mathcal{E}}$ are measure preserving and that diagrams (5) and (6) commute, implies that $p_{\mathcal{C}'}^{\mathcal{Z}'} = (\varphi_{\mathcal{Z}}, \varphi_{\mathcal{Z}})_* p_{\mathcal{C}}^{\mathcal{Z}}$. Then because diagram (7) commutes, the weakly supervised distributions coincide, $p_{\mathcal{M}'}^{\mathcal{X}} = p_{\mathcal{M}}^{\mathcal{X}}$.

“(1) \Rightarrow (2)”: Conversely, if the weakly supervised distributions coincide, $p_{\mathcal{M}'}^{\mathcal{X}} = p_{\mathcal{M}}^{\mathcal{X}}$, the images of $g : \mathcal{Z} \rightarrow \mathcal{X}, g' : \mathcal{Z}' \rightarrow \mathcal{X}$ coincide,

$$\varphi = g'^{-1} \circ g : \mathcal{Z} \rightarrow \mathcal{Z}' \tag{9}$$

is a diffeomorphism, and φ preserves the weakly supervised distribution over causal variables: $p_{\mathcal{C}'}^{\mathcal{Z}'} = (\varphi, \varphi)_* p_{\mathcal{C}}^{\mathcal{Z}}$.

LCM equivalence then follows from showing that $\varphi : \mathcal{D} \rightarrow \mathcal{D}'$ is an ISCM isomorphism, where $\mathcal{D} = \langle \mathcal{C}, \mathcal{I}, p_{\mathcal{I}} \rangle$ and $\mathcal{D}' = \langle \mathcal{C}', \mathcal{I}', p'_{\mathcal{I}'} \rangle$ be the ISCMs inherent to \mathcal{M} and \mathcal{M}' . We show this in the following steps:

1. For each intervention I in \mathcal{D} , there is a corresponding intervention I' in \mathcal{D}' , given by a permutation $\psi : [n] \rightarrow [n]$, such that φ preserves the interventional distribution.
2. The diffeomorphism φ is ψ -diagonal.
3. The permutation ψ preserved the ancestry structure of graphs $\mathcal{G}(\mathcal{C})$ and $\mathcal{G}(\mathcal{C}')$.
4. The diffeomorphism $\varphi_{\mathcal{E}} : \mathcal{E} \rightarrow \mathcal{E}$ of noise variables is ψ -diagonal.
5. The causal mechanisms are compatible with φ .

Step 1: Interventions preserved Remember that the diffeomorphism $\varphi : \mathcal{Z} \rightarrow \mathcal{Z}'$ is such that $p_{\mathcal{C}'}^{\mathcal{Z}'} = (\varphi, \varphi)_* p_{\mathcal{C}}^{\mathcal{Z}}$. For atomic interventions $I \neq J \in \mathcal{I}$, consider the intersection of the

supports of the weakly supervised distribution for interventions on I and J : $U = \text{supp } p_C^{\mathcal{Z}, \mathcal{I}}(z, \tilde{z} | I) \cap \text{supp } p_C^{\mathcal{Z}, \mathcal{I}}(z, \tilde{z} | J) \subset \mathcal{Z} \times \mathcal{Z}$. Note that U has zero measure in $p_C^{\mathcal{Z}, \mathcal{I}}(U | I) = p_C^{\mathcal{Z}, \mathcal{I}}(U | J) = 0$. The distribution is thus a discrete mixture on (z, \tilde{z}) of non-overlapping distributions.

The diffeomorphism (φ, φ) must map between these mixtures. Thus there exists a bijection $\psi : \mathcal{I} \rightarrow \mathcal{I}'$, also inducing a permutation $\psi : [n] \rightarrow [n]$, such that

$$p_{C'}^{\mathcal{Z}', \mathcal{I}'} = (\varphi, \varphi, \psi)_* p_C^{\mathcal{Z}, \mathcal{I}}.$$

Step 2: φ is ψ -diagonal This measure preservation lets us define two equal distributions on $\mathcal{Z} \times \tilde{\mathcal{Z}}' \times \mathcal{I}$, namely $(\text{id}_{\mathcal{Z}}, \varphi, \text{id}_{\mathcal{I}})_* p_C^{\mathcal{Z}, \mathcal{I}}$ and $(\varphi^{-1}, \text{id}_{\tilde{\mathcal{Z}}'}, \psi^{-1})_* p_{C'}^{\mathcal{Z}', \mathcal{I}'}$. In particular, these must then have equal conditionals $p(\tilde{z}' | z, I)$. Thus, for any $U \subseteq \tilde{\mathcal{Z}}'$, $z \in \mathcal{Z}$, $I \in \mathcal{I}$,

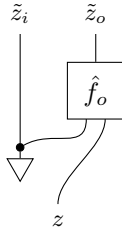
$$p_{C'}^{\mathcal{Z}', \mathcal{I}'}(\tilde{z}' \in U | \varphi(z), \psi(I)) = p_C^{\mathcal{Z}, \mathcal{I}}(\tilde{z} \in \varphi^{-1}(U) | z, I)$$

The conditional probability $p_C^{\mathcal{Z}, \mathcal{I}}(\tilde{z} | z, I)$ can be interpreted as a stochastic map $\mathcal{Z} \rightarrow \tilde{\mathcal{Z}}$. The above relation can then be written as a commuting diagram of stochastic maps, $\forall I \in \mathcal{I}, I' = \psi(I)$:

$$\begin{array}{ccc} \mathcal{Z} & \xrightarrow{p_C^{\mathcal{Z}, \mathcal{I}}(\tilde{z} | z, I)} & \tilde{\mathcal{Z}} \\ \downarrow \varphi & & \downarrow \varphi \\ \mathcal{Z}' & \xrightarrow{p_{C'}^{\mathcal{Z}', \mathcal{I}'}(\tilde{z}' | z', I')} & \tilde{\mathcal{Z}}' \end{array} \quad (10)$$

where we treat $\varphi : \mathcal{Z} \rightarrow \mathcal{Z}'$ as a deterministic stochastic map.

For any variable $i \in [n]$, write the other nodes as $o = [n] \setminus \{i\}$. Let $I = \{i\}$. Then $p_C^{\mathcal{Z}, \mathcal{I}}(\tilde{z} | z, I)$ can be written as a string diagram of stochastic maps:



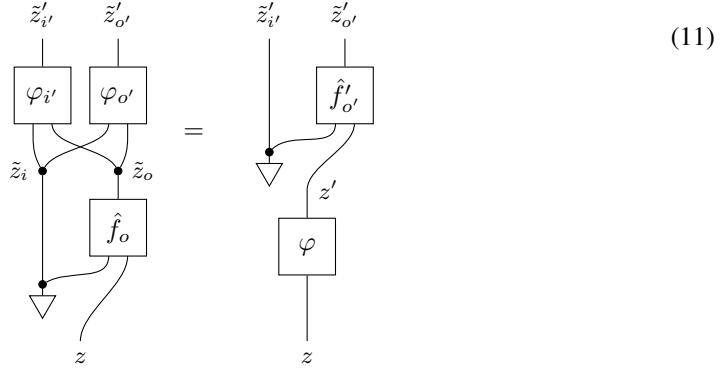
This string diagram represents a conditional probability distribution $p(\tilde{z}_i, \tilde{z}_o | z)$ and is read from the bottom to the top. String diagrams map formally to a generative process [4] and have been used previously in the context of causal models [5]. In this case, the diagram maps to:

$$\tilde{z}_i \sim p(\tilde{z}_i), \quad \tilde{z}_o = \hat{f}_o(\tilde{z}_i, z)$$

where $p(\tilde{z}_i)$ is the interventional distribution and the deterministic map $\hat{f}_o : \tilde{\mathcal{Z}}_i \times \mathcal{Z} \rightarrow \tilde{\mathcal{Z}}_o$ can be constructed from the inverse solution $s^{-1} : \mathcal{Z} \rightarrow \mathcal{E}$ and the causal mechanisms. Each box in a string diagram of stochastic maps denotes a stochastic map and each line to a measurable space. The triangle is the stochastic map $\star \rightarrow \tilde{\mathcal{Z}}_i$ (the star denoting the one-point space; maps from which correspond to probability distributions over the codomain). The \bullet represents copying a variable.

The above commuting diagram (10) can then be written as the equality of the following two string diagrams, where $\psi(I) = I' = \{i'\}$, $o' = [n] \setminus \{i'\}$. We write $\varphi : \mathcal{Z} \rightarrow \mathcal{Z}'$ as the pair $\varphi_{i'} : \mathcal{Z} \rightarrow$

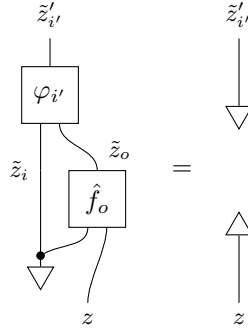
$\mathcal{Z}'_{i'}, \varphi_{o'} : \mathcal{Z} \rightarrow \mathcal{Z}'_{o'}$, obtained by projecting the output of φ to the partition $\mathcal{Z}' = \mathcal{Z}'_{i'} \times \mathcal{Z}'_{o'}$:



This should be read as the equality of the two conditional probability distributions $p(\tilde{z}'_{i'}, \tilde{z}'_{o'} \mid z)$ generated in the following way:

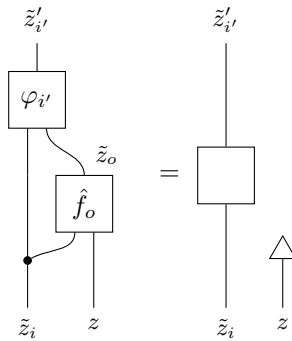
$$\begin{aligned} \text{Left: } & \tilde{z}_i \sim p(\tilde{z}_i), & \tilde{z}_o = \hat{f}_o(\tilde{z}_i, z), & \tilde{z}'_{i'} = \varphi(\tilde{z}_i, \tilde{z}_o)_{i'}, & \tilde{z}'_{o'} = \varphi(\tilde{z}_i, \tilde{z}_o)_{o'}. \\ \text{Right: } & z' = \varphi(z), & \tilde{z}'_{i'} \sim p'(\tilde{z}'_{i'}), & \tilde{z}'_{o'} = \hat{f}'_{o'}(\tilde{z}'_{i'}, z'). \end{aligned}$$

The string diagram equality (11) implies equality when we disregard outputs $\mathcal{Z}'_{o'}$:



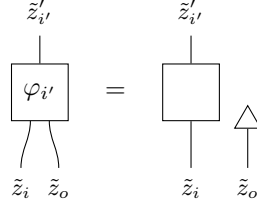
where the upwards pointing triangle represents discarding a variable.

Using Lemma 2, and the fact that $\tilde{\mathcal{Z}}_i = \tilde{\mathcal{Z}}'_{i'} = \mathbb{R}$, the composed differentiable function $\tilde{\mathcal{Z}}_i \times \mathcal{Z} \rightarrow \tilde{\mathcal{Z}}'_{i'}$ is constant in \mathcal{Z} . Thus we have a deterministic function $\tilde{\mathcal{Z}}_i \rightarrow \tilde{\mathcal{Z}}'_{i'}$ such that:



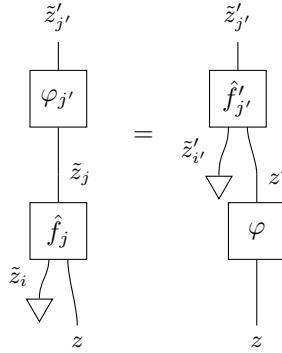
The deterministic function $\tilde{\mathcal{Z}}_i \times \mathcal{Z} \rightarrow \tilde{\mathcal{Z}}_i \times \tilde{\mathcal{Z}}_o$ is surjective and both the left- and right-hand side can be seen as first applying this function (though the output is discarded on the right hand side), which

implies there exists a function $\tilde{\mathcal{Z}}_i \rightarrow \tilde{\mathcal{Z}}'_{i'}$ such that



In words, the function $\varphi_{i'} : \mathcal{Z}_i \times \mathcal{Z}_o \rightarrow \mathcal{Z}'_{i'}$ is constant in \mathcal{Z}_o . This holds for all i and thus φ is ψ -diagonal.

Step 3: Ancestry preserved Let $i \neq j \in [n]$, $i' = \psi(i)$, $j' = \psi(j)$, and $I = \{i\}$. Writing φ as ψ -diagonal, the commuting diagram (10) for the j' component of \tilde{z}' , can be written as the following string diagram:



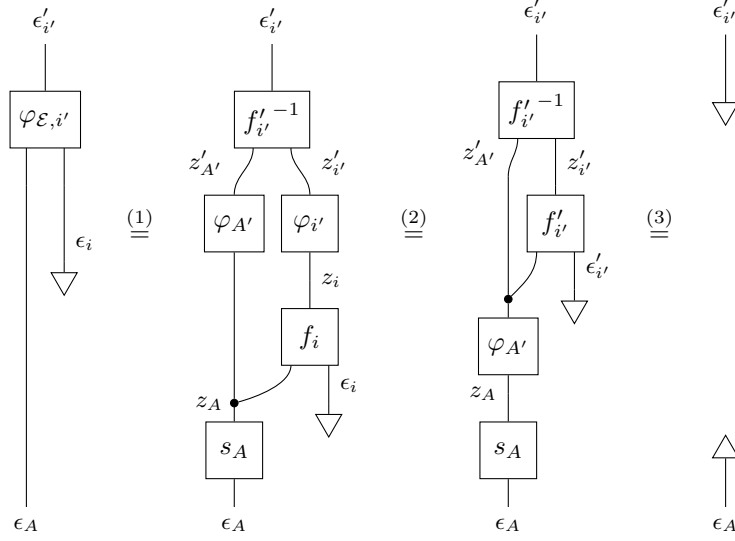
The left hand side is a deterministic map $\mathcal{Z} \rightarrow \tilde{\mathcal{Z}}'_{j'}$, if and only if \hat{f}_j is constant in $\tilde{\mathcal{Z}}_i$ which by faithfulness is the case if and only if $i \notin \mathbf{anc}_j$. The same holds on the right hand side, so $\forall i \neq j \in [n]$, $i \in \mathbf{anc}_j^C \iff \psi(i) \in \mathbf{anc}_{\psi(j)}^C$.

Step 4: Noise map diagonal Define $\varphi_{\mathcal{E}} = s'^{-1} \circ \varphi \circ s : \mathcal{E} \rightarrow \mathcal{E}'$. Note that $\varphi_{\mathcal{E}}(\epsilon)_{i'}$ only depends on ϵ_i and $\epsilon_{\mathbf{anc}_i}$, because $s(\epsilon)_{\mathbf{anc}_i, i}$ and $s'^{-1}(z')_{i'}$ only depend on ancestors, φ is ψ -diagonal and ψ preserves ancestry.

The map φ is measure-preserving. Thus $\forall i$ and writing $A = \mathbf{anc}_i$, the conditional $p(z_i | z_A) = p(z_i | z_{\mathbf{pa}_i})$, interpreted as a stochastic map, is preserved by φ . We can express this as another commuting diagram, in which the two paths from \mathcal{E}_A to $\mathcal{E}'_{i'}$ must be equal:

$$\begin{array}{ccccc}
 \mathcal{E}_A & \xrightarrow{s_A} & \mathcal{Z}_A & \xrightarrow{p(z_i | z_{\mathbf{pa}_i})} & \mathcal{Z}_{A,i} \\
 & & \varphi_A \downarrow & & \downarrow \varphi_{A,i} \\
 & & \mathcal{Z}'_{A'} & \xrightarrow{p(z'_{i'} | z'_{\mathbf{pa}_{i'}})} & \mathcal{Z}'_{A',i'} \xrightarrow{f'_{i'}^{-1}} & \mathcal{E}'_{i'}
 \end{array}$$

where $f'_{i'}^{-1}(z') = f(z'_{\text{pa}_{i'}}, \cdot)^{-1}(z'_{i'})$. Then we have:



where the first equality follows from the definition of $\varphi_{\mathcal{E}, i'}$, the second equality from the commuting diagram above and the third equality from the fact that $f'_{i'}$ and $f'_{i'}^{-1}$ cancel. Then, again using Lemma 2, the map on the left hand side must be constant in ϵ_A . The noise encoding is thus also ψ -diagonal.

Step 5: Equivalence Consider for a variable i and with $i' = \psi(i)$ the following commuting diagram of deterministic maps. Note that we write the causal mechanism f_i as a function of all ancestors, not just the parents, so it is constant in the non-parents. Because of faithfulness, it is non-constant in the parents. Since ψ preserves ancestors, $f'_{i'}$ is well-typed.

$$\begin{array}{ccc}
 \mathcal{E} & \xrightarrow{\varphi_{\mathcal{E}}} & \mathcal{E}' \\
 \downarrow (s_{\text{anc}_i}, \text{id}_{\mathcal{E}_i}) & & \downarrow (s'_{\text{anc}_{i'}}, \text{id}_{\mathcal{E}'_{i'}}) \\
 \mathcal{Z}_{\text{anc}_i} \times \mathcal{E}_i & \xrightarrow{(\varphi_{\mathcal{Z}, \text{anc}_i}, \varphi_{\mathcal{E}, i})} & \mathcal{Z}'_{\text{anc}_{i'}} \times \mathcal{E}'_{i'} \\
 \downarrow f_i & & \downarrow f'_{i'} \\
 \mathcal{Z}_i & \xrightarrow{\varphi_{\mathcal{Z}, i}} & \mathcal{Z}'_{i'}
 \end{array}$$

The composition of the left vertical maps is equal to s_i , the composition of the right vertical maps to $s'_{i'}$. Therefore and because of the definition of $\varphi_{\mathcal{E}}$, the outer and the top square commute. Then, because $(s_{\text{anc}_i}, \text{id}_{\mathcal{E}_i})$ is surjective, the bottom square also commutes [6, Lemma 1.6.21].

Then for $z_j \in \text{anc}_i$, we have that

$$z_j \in \text{pa}_i^{\mathcal{C}} \iff f_i \text{ not constant in } z_j \iff f'_{i'} \text{ not constant in } z'_{j'} \iff z'_{j'} \in \text{pa}_{i'}^{\mathcal{C}'}$$

And thus ψ not only preserves ancestry, but also parenthood and is thus a graph isomorphism $\psi : \mathcal{G}(\mathcal{C}) \rightarrow \mathcal{G}(\mathcal{C}')$. Diagram (5) commutes, and we have established an SCM isomorphism $\varphi : \mathcal{C} \rightarrow \mathcal{C}'$.

To have this also be an ISCM isomorphism, we need diagram (6) to commute and the distribution over interventions to be preserved. For the first, use the fact that all maps in (6) are isomorphisms to simply define $\tilde{\varphi}_{\mathcal{E}}$ so that the diagram commutes. The second follows directly from the assumptions. Hence $\varphi : \mathcal{D} \rightarrow \mathcal{D}'$ is an ISCM isomorphism, $\mathcal{D} \sim \mathcal{D}'$, and—together with the arguments in the beginning of this proof—finally $\mathcal{M} \sim \mathcal{M}'$. \square

B Limitations & generalization

Our identifiability result relies on a few assumptions. Here we discuss some key requirements of Thm. 1 and whether they can be relaxed.

Diffeomorphic causal mechanisms In Def. 4, we require causal mechanisms to be pointwise diffeomorphisms from noise variables to causal variables. Under some mild smoothness assumptions, any SCM can be brought into this form by elementwise redefinitions of the variables, without affecting the observational or interventional distributions. However, such a redefinition may change counterfactual / weakly supervised distributions.

All interventions observed To guarantee identifiability, we require that the intervention distribution has support for any atomic intervention: datasets need to contain data pairs generated from interventions on any causal variable. However, in many systems not all variables can be intervened upon, for instance because some variables are fundamentally immutable or for safety reasons. In that case, the LCM may not be fully identifiable, but there may still be partial identifiability. Interventions on child variables, for instance, can guarantee the identifiability of the parents [7]. We leave a precise characterization of the equivalence classes under such partial weak supervision for future work.

Perfect interventions Our proof of Thm. 1 requires perfect interventions, i. e. that intervened-upon mechanisms do not depend on any causal variables. This is arguably the biggest mismatch between our assumptions and many real-world systems.

If we try to generalize this assumption and allow the intervention mechanisms to depend on the parent variables, identifiability is lost. A simple counterexample is the following.

Example 1 (Non-identifiable ISMCs under imperfect interventions). *Consider two inequivalent ISMCs $\mathcal{D}, \mathcal{D}'$, each with two variables and graph $z_1 \rightarrow z_2$. Let the mechanisms be equal, except that $f_2'(\epsilon_2; z_1) = f_2(\epsilon_2; z_1) + z_1$ and $\tilde{f}_2'(\tilde{\epsilon}_2; z_1) = \tilde{f}_2(\tilde{\epsilon}_2; z_1) + z_1$. Then it is easy to see that a ϕ that is the identity on the first variable and on the second variable is $\phi_2(z_1, z_2) = z_1 + z_2$, preserves the weakly supervised distribution, but is not an ISCM isomorphism.*

Diffeomorphic decoder Definition 1 and Thm. 1 assume that the map from causal variables to observed data is given by a deterministic, diffeomorphic decoder. However, our practical implementation in a VAE uses a stochastic decoder and allows for noisy data. Our experiments provide empirical evidence for identifiability in this setting. We believe that it may be possible to extend Thm. 1 to stochastic decoders, similarly to Khemakhem et al. [8]. We plan to study this extension in future work.

Independent of whether the decoder is deterministic or stochastic, however, is the requirement that the value of all causal variables need to be reflected in the observed low-level data. Our approach (and as far as we are aware all other methods for causal representation learning) is not able to detect causal variables that do not directly influence the low-level data. Latent confounders, present in many real-world problems, provide an as-of-now unsolved hurdle to causal representation learning.

Real-valued causal variables Theorem 1 assumes real-valued causal and noise variables, $\mathcal{Z}_i = \mathcal{E}_i = \mathbb{R}$. We can easily extend this to intervals $(a, b) \in \mathbb{R}$, as these are isomorphic to \mathbb{R} . However, the extension to arbitrary continuous spaces or \mathbb{R}^n is not straightforward. The main reason is that our proof relies on Lemma 2, which does not generalize.

Let us provide a counterexample for identifiability with circle S^1 -valued causal variables.

Example 2 (S^1 -valued non-identifiable LCMs). *Consider an LCM $\mathcal{M} = \langle \mathcal{C}, \mathcal{X}, g, \mathcal{I}, p_{\mathcal{I}} \rangle$ with the following components:*

- The SCM \mathcal{C} consists of two circle-valued variables $z_1, z_2 \in S^1$ with noise variables $\epsilon_1, \epsilon_2 \in S^1$. We parameterize S^1 as $[0, 2\pi)$ with addition defined modulo 2π .
- The causal graph is $z_1 \rightarrow z_2$.
- The causal mechanisms are $f_1(\epsilon_1) = \epsilon_1$ and $f_2(\epsilon_2; z_1) = \epsilon_2 + z_1$.
- The solution function is $s(\epsilon_1, \epsilon_2) = (\epsilon_1, \epsilon_2 + \epsilon_1)$.
- The noise variables are distributed as $\epsilon_1 \sim \mathcal{U}$, uniformly, and $\epsilon_2 \sim q$, which we require to not be invariant under translations (so in particular not uniform). For example, one can take the von Mises distribution $\log q(\epsilon_2) = \cos(\epsilon_2) + \text{const}$.
- The observation space is \mathcal{X} and the decoder $g : S^1 \times S^1 \rightarrow \mathcal{X}$ is diffeomorphic.
- The intervention set \mathcal{I} consists of the empty intervention, atomic interventions on z_1 with $\tilde{z}_1 \sim \mathcal{U}$, and atomic interventions on z_2 with $\tilde{z}_2 \sim \mathcal{U}$. Each of these interventions has probability $\frac{1}{3}$ in $p_{\mathcal{I}}$.

Note that the SCM is faithful, as $z_1 \not\perp\!\!\!\perp z_2$ in the observational distribution, because q is not translationally invariant. The LCM entails the weakly supervised causal distribution

$$p_{\mathcal{C}}^{\tilde{\mathcal{Z}}}(z, \tilde{z}) = \mathcal{U}(z_1) q(z_2 - z_1) \left[\frac{1}{3} \delta(\tilde{z}_1 - z_1) \delta(\tilde{z}_2 - z_2) + \frac{1}{3} \mathcal{U}(\tilde{z}_1) \delta(\tilde{z}_2 - z_2 - \tilde{z}_1 + z_1) + \frac{1}{3} \delta(\tilde{z}_1 - z_1) \mathcal{U}(\tilde{z}_2) \right] \quad (12)$$

with Dirac delta δ . The weakly supervised data distribution is then given by $p_{\mathcal{M}}^{\mathcal{X}} = (g_*, g_*) p_{\mathcal{C}}^{\tilde{\mathcal{Z}}}$.

Now consider a second LCM $\mathcal{M}' = \langle \mathcal{C}', \mathcal{X}, g', \mathcal{I}', p_{\mathcal{I}'} \rangle$:

- The SCM \mathcal{C}' consists of two circle-valued variables $z'_1, z'_2 \in S^1$ with noise variables $\epsilon'_1, \epsilon'_2 \in S^1$.
- The causal graph is trivial and the causal mechanisms are given by the identity, $f'_i(\epsilon'_i) = \epsilon'_i$.
- The noise variables are distributed as $\epsilon'_1 \sim \mathcal{U}$ and $\epsilon'_2 \sim q$.
- The observation space is \mathcal{X} and the decoder $g' : S^1 \times S^1 \rightarrow \mathcal{X}$ is given by the diffeomorphism $g'(z') = g \circ s(z')$, where s is the solution function of \mathcal{C} .
- The intervention set \mathcal{I}' consists of empty interventions, atomic interventions on z'_1 with $\tilde{z}'_1 \sim \mathcal{U}$, and atomic interventions on z'_2 with $\tilde{z}'_2 \sim \mathcal{U}$. Each of these interventions has probability $\frac{1}{3}$ in $p_{\mathcal{I}'}$.

We find a weakly supervised causal distribution

$$p_{\mathcal{C}'}^{\tilde{\mathcal{Z}}'}(z', \tilde{z}') = \mathcal{U}(z'_1) q(z'_2) \left[\frac{1}{3} \delta(\tilde{z}'_1 - z'_1) \delta(\tilde{z}'_2 - z'_2) + \frac{1}{3} \mathcal{U}(\tilde{z}'_1) \delta(\tilde{z}'_2 - z'_2) + \frac{1}{3} \delta(\tilde{z}'_1 - z'_1) \mathcal{U}(\tilde{z}'_2) \right]. \quad (13)$$

Clearly, two LCMs are not equivalent, because their graphs are non-isomorphic. Yet, if we define

$$\varphi : \mathcal{Z} \rightarrow \mathcal{Z}' : (z_1, z_2) \mapsto (z_1, z_2 - z_1)$$

then the weakly supervised distribution of the causal variables is preserved:

$$\begin{aligned} ((\varphi, \varphi)_* p_{\mathcal{C}}^{\tilde{\mathcal{Z}}})(z', \tilde{z}') &= p_{\mathcal{C}'}^{\tilde{\mathcal{Z}}'}((z'_1, z'_2 + z'_1), (\tilde{z}'_1, \tilde{z}'_2 + \tilde{z}'_1)) \\ &= \mathcal{U}(z'_1) q(z'_2 + z'_1 - z'_1) \left[\frac{1}{3} \delta(\tilde{z}'_1 - z'_1) \delta(\tilde{z}'_2 + \tilde{z}'_1 - (z'_2 + z'_1)) + \frac{1}{3} \mathcal{U}(\tilde{z}'_1) \delta(\tilde{z}'_2 + \tilde{z}'_1 - (z'_2 + z'_1) - \tilde{z}'_1 + z'_1) + \frac{1}{3} \delta(\tilde{z}'_1 - z'_1) \mathcal{U}(\tilde{z}'_2 + \tilde{z}'_1) \right] \\ &= \mathcal{U}(z'_1) q(z'_2) \left[\frac{1}{3} \delta(\tilde{z}'_1 - z'_1) \delta(\tilde{z}'_2 - z'_2) + \frac{1}{3} \mathcal{U}(\tilde{z}'_1) \delta(\tilde{z}'_2 - z'_2) + \frac{1}{3} \delta(\tilde{z}'_1 - z'_1) \mathcal{U}(\tilde{z}'_2) \right] \\ &= p_{\mathcal{C}'}^{\tilde{\mathcal{Z}}'}(z', \tilde{z}') \end{aligned}$$

where we use that the density \mathcal{U} is constant. Also, because $\varphi = s^{-1}$ and $g'(z') = g \circ s(z')$, we have that $p_{\mathcal{M}}^{\mathcal{X}} = p_{\mathcal{M}'}^{\mathcal{X}}$.

So these two models with their non-isomorphic graph structures have identical weakly-supervised distributions on the observables x, \tilde{x} . They therefore provide a counter-example for a straightforward generalization of Thm. 1 to causal variables with arbitrary continuous domains.

The key issue here is that the interventional distribution on \tilde{z}_2 has many symmetries or automorphisms: diffeomorphic maps $\mathcal{Z}_2 \rightarrow \mathcal{Z}_2$ that preserve $p(\tilde{z}_2)$ — in this case these are the cyclic translations. In general, for any causal model \mathcal{D} with two variables $z_1 \rightarrow z_2$, we can construct a map $\phi(z_1, z_2) = (z_1, \Gamma(z_1)(z_2))$ where for all z_1 , $\Gamma(z_1) : \mathcal{Z}_2 \rightarrow \mathcal{Z}_2$ is a differentiable map from \mathcal{Z}_1 to a diffeomorphism on \mathcal{Z}_2 that preserves the interventional distribution $p(\tilde{z}_2)$. This map ϕ preserves the weakly supervised distribution from \mathcal{D} to a unique model \mathcal{D}' , whose causal mechanisms are:

$f'_1(\epsilon_1) = f_1(\epsilon_1)$ and $f'_2(z_1, \epsilon_1) = \Gamma(z_1)(f_2(z_1, \epsilon_2))$. However, ϕ is only an ISCM morphism if it is also diagonal, and thus Γ must be constant in z_1 .

For the \mathbb{R} -valued variables of the main paper, any smooth distribution on \mathbb{R} has exactly two automorphisms, related to the automorphisms of the univariate Gaussian distribution $x \mapsto x$ and $x \mapsto -x$. Γ can not smoothly switch between these and thus must be constant, making ϕ diagonal and an ISCM morphism. However, any multi-dimensional distribution has many automorphisms. One class of these is related to the orthogonal transformations of a standard multivariate Gaussian. Another, much larger, class is related to the flows generated by divergence-free vector fields on the unit ball. Γ can smoothly choose different such automorphisms for different values of z_1 , making ϕ not diagonal and thus not an ISCM morphism. In conclusion, this smooth space of automorphisms make the multivariate case unidentifiable from weak supervision.

C Implicit latent causal models

C.1 Behaviour of noise encodings under interventions

We first prove a key property of ISCMs, which motivate the construction of ILCMs:

Lemma 3. *Let $\mathcal{D} = \langle \mathcal{C}, \mathcal{I}, p_{\mathcal{I}} \rangle$ be an ISCM with n real-valued causal variables. Sample from the weakly-supervised distribution over noise encodings $(e, \tilde{e}) \sim p(e, \tilde{e}|I)$. Then $\forall i \notin I$, almost surely $e_i = \tilde{e}_i$.*

Proof. First, note that $z = s(\epsilon)$ and $e = s^{-1}(z)$, so $\epsilon = e$. If $i \notin I$, then the unintervened noise equals the intervened noise $\tilde{e}_i = \epsilon_i$, so

$$\begin{aligned} \tilde{e}_i &= s^{-1}(\tilde{z})_i \\ &= f_i(\cdot, \tilde{z}_{\text{pa}_i})^{-1}(\tilde{z}_i) \\ &= f_i(\cdot, \tilde{z}_{\text{pa}_i})^{-1}(f_i(\tilde{e}_i, \tilde{z}_{\text{pa}_i})) \\ &= \tilde{e}_i = \epsilon_i = e_i \end{aligned} \quad \square$$

C.2 Model specification

An implicit latent causal model for a system of n causal variables consists of the following components:

- a Gaussian noise encoder $q(e|x)$ with mean $\mu_e(x)$ and standard deviation $\sigma_e(x)$ implemented as neural networks;
- a Gaussian noise decoder $p(x|e)$ with mean $\mu_x(e)$ implemented as neural network and fixed, constant standard deviation;
- an intervention encoder $q(I|x, \tilde{x})$ defined as

$$\log q(i \in I|x, \tilde{x}) = \frac{1}{Z} (a + b |\mu_e(x)_i - \mu_e(\tilde{x})_i| + c |\mu_e(x)_i - \mu_e(\tilde{x})_i|^2),$$

where $a < 0$ and $b, c > 0$ are learnable parameters and where the normalization constant Z is defined such that $\sum_I q(i \in I|x, \tilde{x}) = 1$;

- solution functions $s_i(e_i; e_{\setminus i})$ for $i = 1, \dots, n$ implemented as invertible affine transformations, where the offset and slope are functions of $e_{\setminus i}$ implemented with neural networks;
- noise priors $p_i(e_i)$, which we choose to be standard Gaussian;
- an post-intervention causal-variable prior $\tilde{\pi}(\tilde{z}_i)$, which we choose to be standard Gaussian; and
- an intervention-target prior $p(I)$, which we choose to be uniform.

Encoding a data pair (x, \tilde{x}) during training consists of the following steps:

$$\begin{aligned} I &\sim q(I|x, \tilde{x}) \\ e_{\text{preliminary}} &\sim q(e|x) & \tilde{e}_{\text{preliminary}} &\sim q(\tilde{e}|\tilde{x}) \\ \forall i, \lambda_i &\sim \text{Uniform}(0, 1) & e_{\text{average } i} &= \lambda_i e_{\text{preliminary } i} + (1 - \lambda_i) \tilde{e}_{\text{preliminary } i} \\ e_i &= \begin{cases} e_{\text{preliminary } i} & i \in I \\ e_{\text{average } i} & i \notin I \end{cases} & \tilde{e}_i &= \begin{cases} \tilde{e}_{\text{preliminary } i} & i \in I \\ e_{\text{average } i} & i \notin I \end{cases} \end{aligned} \quad (14)$$

Algorithm 1 Schematic ILCM training.

Require: Training data $p_{\text{data}}(x, \tilde{x})$

Require: ILCM with encoder $q(e, \tilde{e}, I|x, \tilde{x})$, decoder $p(x|e)$, solution functions $s_i(e_i; e_{\setminus i})$.

Require: Optimizer

Require: Loss weights α, β , optimizer hyperparameters, ILCM initialization

- 1: Randomly initialize ILCM
 - 2: **while** not converged **do**
 - 3: Sample data batch $x, \tilde{x} \sim p_{\text{data}}(x, \tilde{x})$
 - 4: Encode $e \sim q^{\text{phase 1}}(x), \tilde{e} \sim q^{\text{phase 1}}(\tilde{x})$
 - 5: Compute loss $L \leftarrow \mathcal{L}^{\text{phase 1}}(x, \tilde{x})$ (Eq. (17))
 - 6: Train ILCM on L with optimizer
 - 7: **end while**
 - 8: **while** not converged **do**
 - 9: Sample data batch $x, \tilde{x} \sim p_{\text{data}}(x, \tilde{x})$
 - 10: Encode $e, \tilde{e}, I \sim q(e, \tilde{e}, I|x, \tilde{x})$ (Eq. (14))
 - 11: Compute loss $L \leftarrow \mathcal{L}^{\text{phase 2}}(x, \tilde{x})$ (Eq. (18))
 - 12: Train ILCM on L with optimizer
 - 13: **end while**
 - 14: Freeze convolutional layers in $q(e, \tilde{e}, I|x, \tilde{x})$
 - 15: **while** not converged **do**
 - 16: Sample data batch $x, \tilde{x} \sim p_{\text{data}}(x, \tilde{x})$
 - 17: Encode $e, \tilde{e}, I \sim q(e, \tilde{e}, I|x, \tilde{x})$ (Eq. (14))
 - 18: Compute loss $L \leftarrow \mathcal{L}(x, \tilde{x})$ (Eq. (16))
 - 19: Train ILCM on L with optimizer
 - 20: **end while**
 - 21: Determine top. order $o \leftarrow \text{TopoOrderHeuristic}(\text{ILCM}, p_{\text{data}}(x, \tilde{x}))$ (described in Sec. C.5)
 - 22: Modify s_i to only depend on $s_i(e_i; e_{\text{anc}_i})$ according to o
 - 23: **while** not converged **do**
 - 24: Sample data batch $x, \tilde{x} \sim p_{\text{data}}(x, \tilde{x})$
 - 25: Encode $e, \tilde{e}, I \sim q^{\text{phase 4}}(e, \tilde{e}, I|x, \tilde{x})$ (Eq. (20))
 - 26: Compute loss $L \leftarrow \mathcal{L}(x, \tilde{x})$ (Eq. (16))
 - 27: Train ILCM on L with optimizer
 - 28: **end while**
-

In the first line, we encode to an intervention target (either by sampling or by enumerating all possibilities and summing up the corresponding loss terms). In the second line, the data is encoded to the noise-encoding space. The third and fourth line project these noise encodings such that for those components e_i that are not intervened upon, $i \notin I$, we have that the pre-intervention noise encoding and post-intervention noise encoding are equal, $e_i = \tilde{e}_i$. This makes sure that the latents are consistent with the weakly supervised structure, and punishes deviations from this structure through the reconstruction error (likelihood). We use the symbol $e, \tilde{e} \sim q(e, \tilde{e}|x, \tilde{x}, I)$ to refer to the second to last line together.

The prior is given by

$$p(e, \tilde{e}_I, I) = p(I) \prod_i p_i(e_i) \prod_{i \in I} p(\tilde{e}_i|e)$$

with, for $i \in I$,

$$p(\tilde{e}_i|e) = \tilde{p}(s_i(\tilde{e}_i|e_{\setminus i})) \left| \frac{\partial \bar{s}_i(\tilde{e}_i; e_{\setminus i})}{\partial \tilde{e}_i} \right|. \quad (15)$$

C.3 Training

Loss We train ILCMs by minimizing the β -VAE loss

$$\begin{aligned} \mathcal{L}_{\text{ILCM}} = \mathbb{E}_{x, \tilde{x}} \mathbb{E}_{I \sim q(I|x, \tilde{x})} \mathbb{E}_{e, \tilde{e} \sim q(e, \tilde{e}|x, \tilde{x}, I)} & \left[\log p(x|e) + \log p(\tilde{x}|\tilde{e}) \right. \\ & \left. + \beta \left\{ \log p(e, \tilde{e}_I, I) - \log q(I|x, \tilde{x}) - \log q(e, \tilde{e}_I|x, \tilde{x}, I) \right\} \right], \end{aligned}$$

where β is a hyperparameter.

Two additional loss terms are used as regularizers during training. The first is a plain reconstruction error (or log likelihood term) that does not use the projections given in Eq. (14) to encourage consistency between the noise encoder and noise decoder:

$$\mathcal{L}_{\text{reco}} = \mathbb{E}_{x, \tilde{x}} \mathbb{E}_{e \sim q(e|x)} \mathbb{E}_{\tilde{e} \sim q(\tilde{e}|\tilde{x})} \left[\log p(x|e) + \log p(\tilde{x}|\tilde{e}) \right].$$

Throughout training, we also add the negative entropy of the batch-aggregate intervention posterior $q_I^{\text{batch}}(I) = \mathbb{E}_{x, \tilde{x} \in \text{batch}} [q(I|x, \tilde{x})]$:

$$\mathcal{L}_{\text{entropy}} = \mathbb{E}_{\text{batches}} \left[- \sum_I q_I^{\text{batch}}(I) \log q_I^{\text{batch}}(I) \right].$$

This helps avoid a collapse of the latent space to a lower-dimensional subspace. The overall loss is then given by

$$\mathcal{L} = \mathcal{L}_{\text{ILCM}} + \alpha \mathcal{L}_{\text{reco}} + \gamma \mathcal{L}_{\text{entropy}} \quad (16)$$

with hyperparameters $\alpha, \gamma \geq 0$.

Training phases We train ILCM models in four phases:

1. We begin with a short pre-training phase, in which the noise encoder and noise decoder are trained on a plain β -VAE loss with a standard Gaussian prior,

$$\begin{aligned} \mathcal{L}^{\text{phase 1}} = \mathbb{E}_{x, \tilde{x}} \mathbb{E}_{e, \tilde{e} \sim q(e|x)q(\tilde{e}|\tilde{x})} & \left[\log p(x|e) + \log p(\tilde{x}|\tilde{e}) \right. \\ & \left. + \beta \left\{ \log \mathcal{N}(e) + \log \mathcal{N}(\tilde{e}) - \log q(e|x) - \log q(\tilde{e}|\tilde{x}) \right\} \right]. \quad (17) \end{aligned}$$

This provides a good starting point for the remainder of the training, in which the encoders and decoders have already learned some patterns in the data space.

2. Next, we train the noise encoder, noise decoder, and intervention encoder. In this phase we do not yet use the neural solution functions and instead model $p(\tilde{e}_i|e)$ with a uniform probability density. Ignoring irrelevant constants, the total loss is given by

$$\begin{aligned} \mathcal{L}^{\text{phase 2}} = \mathbb{E}_{x, \tilde{x}} \mathbb{E}_{I \sim q(I|x, \tilde{x})} \mathbb{E}_{e, \tilde{e} \sim q(e, \tilde{e}|x, \tilde{x}, I)} & \left[\log p(x|e) + \log p(\tilde{x}|\tilde{e}) + \beta \left\{ \log p(I) \right. \right. \\ & \left. \left. + \sum_i \log p_i(e_i) - \log q(I|x, \tilde{x}) - \log q(e, \tilde{e}_I|x, \tilde{x}, I) \right\} \right] + \alpha \mathcal{L}_{\text{reco}} + \gamma \mathcal{L}_{\text{entropy}}. \quad (18) \end{aligned}$$

This avoids harmful feedback from the randomly initialized solution functions influencing the training of the encoder and decoders, stabilizing the learning of good latent representations.

3. We then “switch on” the solution functions and model the intervention targets with the density $p(\tilde{e}_i|e)$ as given above and train on the combined loss \mathcal{L} .
On image datasets, we freeze the convolutional layers in the encoder in this stage and only continue training the final layers of the encoder together with the solution functions.
4. For a final fine-tuning phase, we change the setup in two more ways. First, we analyze the learned solution functions to infer the most likely topological order. For this we use the step 1 of the heuristic algorithm for graph inference, which we will define in Sec. C.5 below.

Then the solution functions are modified such that s_i only depends on the ancestors e_{anc_i} according to the inferred topological order:

$$s_i(e_i; e_{\setminus i}) \rightarrow s_i(e_i; e_{\text{anc}_i}). \quad (19)$$

This is implemented with a suitable masking layer in the neural network implementation of s_i . We find that this form of inductive bias in the prior helps with learning cleanly disentangled representations.

Second, we fix the intervention encoders to the deterministic,

$$q^{\text{deterministic}}(I|x, \tilde{x}) = \begin{cases} 1 & I = \arg \max_I q(I|x, \tilde{x}) \\ 0 & \text{else,} \end{cases}$$

$$q(e, \tilde{e}, I|x, \tilde{x}) = q^{\text{deterministic}}(I|x, \tilde{x})q(e, \tilde{e}|x, \tilde{x}, I). \quad (20)$$

This further improves the training efficiency.

We summarize the whole ILCM training procedure in Alg. 1. The separation of phase 2 and 3—first training the encoder with a simplified prior, then training the solution functions—improved the success of our method substantially. Adding the pre-training and fine-tuning phases 1 and 4 slightly improved the efficiency of the training, but is not critical.

We use the Adam optimizer [9] with a cosine annealing schedule [10], which is restarted at the beginning of training phases 3 and 4. The hyperparameters differ slightly between experiments and will be given in Sec. D.

C.4 Identifiability

We will now show that our identifiability result extends to implicit latent causal models. We construct an equivalence $\Omega : \text{ILCM} \rightarrow \text{ELCM}$ between ILCMs and ELCMs, which preserves the weakly supervised distribution $p(x, \tilde{x})$. This gives an equivalence relation on ILCMs, namely if their corresponding ELCMs are isomorphic according to Def. 9. A direct corollary of Thm. 1 is then that the weakly supervised distributions of ILCMs $\mathcal{N}, \mathcal{N}'$ are equal, $p_{\mathcal{N}}(x, \tilde{x}) = p_{\mathcal{N}'}(x, \tilde{x})$, if and only if they are equivalent, $\mathcal{N} \sim \mathcal{N}'$.

To construct the map Ω between an ILCM \mathcal{N} and ELCM \mathcal{M} , let both have equal: solution function $s : \mathcal{E} \rightarrow \mathcal{Z}$, noise distribution $p(\epsilon)$, intervention distribution $p(I)$, intervention encoder. Let the encoder $q(e|x)$ of \mathcal{N} be equal to the encoder $q(z|x)$ of \mathcal{M} , post-composed with the inverse solution function $s^{-1} : \mathcal{Z} \rightarrow \mathcal{E}$, and the decoder be pre-composed with the solution function. Let the explicit graph in \mathcal{M} equal the graph induced by the solution function of \mathcal{N} , with $\forall i \neq j$, the edge $i \rightarrow j$ exists if $\exists \epsilon$ such that $\partial s(\epsilon)_j / \partial \epsilon_i \neq 0$.

To map from an ILCM \mathcal{N} to an ELCM \mathcal{M} , we pick the intervened causal mechanism \tilde{f}_i as any map such that the causal distribution $p(\tilde{z}_i) = (\tilde{f}_i)_* p(e_i)$ of \mathcal{M} equals the post-intervention causal-variable prior $\tilde{\pi}(\tilde{z}_i)$ in \mathcal{N} . For \mathbb{R} -valued variables and fully supported distributions with smooth densities, there are exactly two possible choices for \tilde{f}_i . To map from an ELCM \mathcal{M} to an ILCM \mathcal{N} , we set $\tilde{\pi}(\tilde{z}_i) := (\tilde{f}_i)_* p(e_i)$.

Following the definitions of the weakly supervised distributions, it is easy to see that Ω preserves weakly supervised distribution, so that if $\mathcal{M} = \Omega(\mathcal{N})$, then $p_{\mathcal{M}}(x, \tilde{x}) = p_{\mathcal{N}}(x, \tilde{x})$, and if $\mathcal{N} = \Omega^{-1}(\mathcal{M})$, then $p_{\mathcal{M}}(x, \tilde{x}) = p_{\mathcal{N}}(x, \tilde{x})$. This proves that our identifiability result of ELCMs also applies to ILCMs.

C.5 Graph inference

ILCMs do not contain an explicit graph representation, but the causal structure is implicitly represented by the learned solution functions. We propose two algorithms for causal discovery based on a trained ILCM model.

ILCM-E We can perform causal discovery in a two-stage procedure. After training an ILCM to learn the causal representations, we use an off-the-shelf method for causal discovery on the learned representations. Since the ILCM allows us to infer intervention targets, we can use intervention-based algorithms. In this paper, we use ENCO [11], a recent differentiable causal discovery method

that exploits interventions to obtain acyclic graphs without requiring constrained optimization. Alternatives to ENCO include DCDI [12] and GIES [13].

ILCM-H Alternatively, we can unearth the causal structure encoded in the learned solution functions s_i , which map noise variables to causal variables. We introduce a heuristic algorithm that can find the causal graph \mathcal{G} and the causal mechanisms f_i from a trained ILCM. It proceeds in three steps:

1. *Computing the topological order:* To determine the topological order of the causal graph, we use the property that after convergence, the solution function $z_i = s_i(e_i; e)$ will only depend on the ancestors of e_i . This allows us to define a heuristic that determines whether z_i is an ancestor of z_j . For each pair (i, j) , we compute

$$\text{ancestry}(i, j) = d\left(s_j(e_j; e), s_j(e_j; \text{mask}_i[e])\right). \quad (21)$$

Here $d(\cdot, \cdot)$ is a distance measure between two functions; in practice, we use the expected MSE over a validation dataset. The function $\text{mask}_i[e]$ replaces the i -th component of e with uninformative input, for instance median of that component computed over the training dataset.

After computing these ancestry scores, we can compute a topological order by sorting the variables such that likely ancestors appear before their likely descendants according to this heuristic. We do this with a greedy algorithm.

2. *Extracting the causal mechanisms:* Next, we compute the causal mechanisms f_i such that $z_i = f_i(e_i; z_{\text{anc}_i})$. We begin by setting i to the root node (the first variable in the topological order computed in step 1) and proceed in topological order. At every step we set

$$f_i(e_i; z_{\text{anc}_i}) = s_i(e_i; \hat{e}) \quad \text{with} \quad \hat{e}_j = \begin{cases} f_j^{-1}(z_j; z_{\text{anc}_j}) & j \in \text{anc}_i \\ \text{mask}[e_j] & \text{otherwise.} \end{cases} \quad (22)$$

3. *Finding causal parents:* Finally, we check whether an ancestor z_i is a parent of a node z_j by testing whether f_j explicitly depends on z_i . Again, we use a heuristic measure of functional dependence:

$$\text{paternity}(i, j) = d\left(f_j(e_j; z_{\text{anc}_j}), f_j(e_j; \text{mask}_i[z_{\text{anc}_j}])\right). \quad (23)$$

We then construct the causal graph by thresholding on this heuristic. This gives us the inferred adjacency matrix

$$\hat{A}_{ij} = \begin{cases} 1 & i \in \text{anc}_j \quad \text{and} \quad \text{paternity}(i, j) > p_{\min} \\ 0 & \text{else} \end{cases} \quad (24)$$

where $p_{\min} > 0$ is a hyperparameter.

The heuristic algorithm (ILCM-H) does not require any optimization and is thus computationally more efficient, but arguably less principled than the likelihood-based ENCO approach (ILCM-E). In our experiments, the ILCM-H approach finds the correct graph in 7 out of the 8 datasets, while ILCM-E always yields the correct causal graph.

D Experiments

D.1 General setup

Baselines We compare our ILCM-E and ILCM-H results (which differ only by the graph inference algorithm, as described above) to three baseline methods. The *disentanglement VAE* (dVAE) method is a VAE for paired data with individual latent components changing between the pre-intervention latents and the post-intervention latents, but without causal structure. We implement it by using an ILCM, but enforce a trivial causal graph by not allowing the solution functions $s_i(\cdot; e)$ to depend on e .

Our second baseline is an unstructured β -VAE, which treats pre-intervention and post-intervention data as i. i. d. and models both with a standard Gaussian prior.

Finally, we include a slot attention baseline. We use as many slots as there are latents. We break the symmetry between the slots by initialising the slots not with a random vector, but with a different learned vector per slot, as is done in Ref. [14]. We choose a six-dimensional latent for each slot.

Metrics We evaluate the disentanglement of the learned causal variables by computing the DCI disentanglement, completeness, and informativeness scores [15]. They are based on a feature importance matrix that quantifies how important each model latent is for predicting each ground-truth causal factor; we compute the feature importance matrix with gradient boosted trees in scikit-learn’s implementation with default parameters [16]. There are many other disentanglement metrics, but empirically these tend to be highly correlated with the DCI disentanglement score [17], so we omit them here for simplicity. For the slot attention models, we add up the contribution of the latent dimensions of each slot, to get a importance matrix between slots and ground truth causal variables.

The quality of intervention inference is evaluated with the accuracy of the intervention encoder. Since we can only identify causal variables and intervention targets up to a permutation, we compute this accuracy for any possible permutation of the causal variables and then report the best result.

Finally, we evaluate the quality of the inferred causal graphs. We identify the ground-truth variables with the corresponding learned causal variables based on the importance matrix computed for the DCI disentanglement score [15]. We then compute the structural Hamming distance (or graph edit distance) between the learned graph and the true graph. As an example, consider the case of two causal variables, where the ground-truth graph is $z_1 \rightarrow z_2$, the ILCM graph is $z'_1 \rightarrow z'_2$, and the ground-truth and learned variables are mapped to each other as $z_1 \leftrightarrow z'_2$ and $z_2 \leftrightarrow z'_1$. Then the structural Hamming distance will be 1, as the cause and effect are flipped in the learned model.

D.2 2D toy experiment

Dataset We first demonstrate LCMs in a pedagogical toy experiment with $\mathcal{X} = \mathcal{Z} = \mathbb{R}^2$. Training data is generated from a nonlinear SCM with the graph $z_1 \rightarrow z_2$ and mapped to the data space through a randomly initialized normalizing flow.







In particular, we have that $z_1 \sim \mathcal{N}(z_1; 0, 1^2)$ and $z_2 \sim \mathcal{N}(z_2; 0.3z_1^2 + 0.6z_1, 0.8^2)$. This latent data is mapped to the data space $\mathcal{X} = \mathbb{R}^2$ with a randomly initialized coupling flow with five affine coupling layers interspersed with random permutations of the dimensions. For the weakly supervised setting we use a uniform intervention prior over $\{\emptyset, \{z_1\}, \{z_2\}\}$. We generate 10^5 training samples, 10^4 validation samples, and 10^4 evaluation samples (where each sample is one pair (x, \tilde{x}) of pre- and post-intervention data).

Architecture The noise encoder and noise decoder are Gaussian, with mean and standard deviation computed by fully connected networks. The solution functions are implemented as affine transformations with slope and offset computed as a function of the pre-intervention noise encodings, also implemented with fully connected networks. For each MLP, we use two hidden layers with 100 units each and ReLU activations.

Training Models are trained using the procedure described in Sec. C.3. We train for $9 \cdot 10^4$ steps using a batch size of 100 and an initial learning rate of 10^{-3} . The weights of the different loss terms and regularizers are as follows: β is initially set to 0 and increased to its final value of 1 during training, $\alpha = 10^{-2}$, and $\gamma = 0$ throughout training. For each method, we train models with three random seeds and in the end select the median run according to the validation loss.

D.3 Causal3DIdent experiments

Dataset In the Causal3DIdent experiments we consider six different datasets, each generated from a different causal graph, SCM, and decoder. The six causal graphs we consider are:

- the trivial graph 
- single edge 
- the chain 
- the fork 
- the collider  and
- the full graph 

For each of these subsets, we randomly generate a nonlinear SCM with heteroskedastic noise: for each causal mechanism, we randomly initialize an MLP that outputs the scale and shift of an affine

Table 2: Detailed experiment results. We compare our ILCM-E (using ENCO for graph inference) and ILCM-H (with a heuristic for graph inference) to disentanglement VAE (dVAE-E), unstructured β -VAE, and slot attention baselines. We show the DCI scores (disentanglement D , completeness C , informativeness I), the accuracy of intervention inference, the learned graph, and the structural Hamming distance (SHD) between learned and true graph. Best results in bold.

































Dataset	True graph	Method	D	C	I	Int. accuracy	Learned graph	SHD
2D toy data		ILCM-E (ours)	0.99	0.99	0.00	0.96		0
		ILCM-H (ours)	0.99	0.99	0.00	0.96		0
		dVAE	0.35	0.50	0.01	0.96		1
		β -VAE	0.52	0.53	0.00	–	–	–
Causal3DIdent		ILCM-E (ours)	0.99	0.99	0.00	0.98		0
		ILCM-H (ours)	0.99	0.99	0.00	0.98		0
		dVAE	1.00	1.00	0.00	0.98		0
		β -VAE	0.94	0.94	0.00	–	–	–
		Slot attention	0.90	0.90	0.01	–	–	–
		ILCM-E (ours)	1.00	1.00	0.00	0.98		0
		ILCM-H (ours)	1.00	1.00	0.00	0.98		0
		dVAE	0.91	0.91	0.00	0.98		1
		β -VAE	0.92	0.92	0.00	–	–	–
		Slot attention	0.56	0.84	0.02	–	–	–
		ILCM-E (ours)	0.99	0.99	0.00	0.98		0
		ILCM-H (ours)	0.99	0.99	0.00	0.98		0
		dVAE	0.83	0.83	0.00	0.98		2
		β -VAE	0.63	0.71	0.00	–	–	–
		Slot attention	0.42	0.59	0.02	–	–	–
		ILCM-E (ours)	0.99	0.99	0.00	0.98		0
		ILCM-H (ours)	0.99	0.99	0.00	0.98		1
		dVAE	0.79	0.81	0.00	0.98		2
		β -VAE	0.63	0.68	0.01	–	–	–
		Slot attention	0.87	0.87	0.03	–	–	–
		ILCM-E (ours)	0.99	0.99	0.00	0.98		0
		ILCM-H (ours)	0.99	0.99	0.00	0.98		0
		dVAE	0.80	0.81	0.01	0.98		2
		β -VAE	0.28	0.52	0.16	–	–	–
Slot attention		0.32	0.35	0.04	–	–	–	
	ILCM-E (ours)	0.99	0.99	0.00	0.98		0	
	ILCM-H (ours)	0.99	0.99	0.00	0.98		0	
	dVAE	0.60	0.64	0.00	0.98		3	
	β -VAE	0.57	0.61	0.01	–	–	–	
	Slot attention	0.53	0.67	0.01	–	–	–	
CausalCircuit		ILCM-E (ours)	0.97	0.97	0.00	1.00		0
		ILCM-H (ours)	0.97	0.97	0.00	1.00		0
		dVAE-E	0.34	0.55	0.00	1.00		5
		β -VAE	0.39	0.43	0.00	–	–	–
		Slot attention	0.39	0.82	0.00	–	–	–



Figure 8: Effect of varying the learned causal factors on the image in the Causal3DIdent dataset. We encode a single test images (middle column) into the three learned causal variables. We then vary each of these causal factors in isolation (without performing interventions, that is, without including the causal effects on other variables) and show the reconstructed images. The ILCM (top) learns a representation that is quite disentangled: z_1 largely corresponds to the spotlight color, z_2 to the spotlight position, and z_3 to the object color. In contrast, the acausal dVAE baseline entangles the object color and spotlight color in its learned representation z_1 .

transformation as a function of the causal parents. We choose an MLP initialization scheme that emphasizes nontrivial, nonlinear causal effects. We then identify a random permutation of the three causal variables with three high-level concepts in the Causal3DIdent dataset: the object hue, the spotlight hue, and the spotlight position. We use the following causal graphs:

- single edge: object hue \rightarrow spotlight position;
- chain: spotlight position \rightarrow spotlight hue \rightarrow object hue;
- fork: spotlight hue \rightarrow spotlight position, object hue;
- collider: spotlight hue \rightarrow object hue \leftarrow spotlight position;
- full graph: spotlight hue \rightarrow object hue \rightarrow spotlight position, spotlight hue \rightarrow spotlight position.

Since all of these properties are defined on a range $[0, 2\pi)$, we apply an elementwise arctanh transform and rescaling to our variables such that they populate a subset of $[0, 2\pi)$. This also avoids topological issues. Next, we generate images in 64×64 resolution following the procedure described in Ref. [7]. We use Blender [18] to generate 3D rendered images based on the previously defined causal variables. To increase diversity of the six datasets, we render each dataset with a different object: Teapot [19], Armadillo [20], Hare [21], Cow [22], Dragon [23], and Horse [24]. We generate 10^5 training samples, 10^4 validation samples, and 10^4 evaluation samples.

Architecture For the noise encoder and noise decoder we use a convolutional architecture with four residual blocks, using downsampling via average-pooling and bilinear upsampling, respectively. We do not use BatchNorm, as we found that that can lead to practical issues when images in a batch are very similar. The output of the convolutional layers is then fed through a fully connected network with two hidden layers, 64 units each, and ReLU activations. For each of the three latents output by the encoder, we apply an additional elementwise MLP with one hidden layers, 16 units each, and ReLU activations. For the solution functions we use the same architecture as in the 2D toy data.

Training Models are trained using the procedure described in Sec. C.3. We train for $2.5 \cdot 10^5$ steps using a batch size of 64 and an initial learning rate of $8 \cdot 10^{-5}$. The weights of the different loss terms and regularizers are as follows: β is initially set to 0 and increased to its final value of 0.05 during training, $\alpha = 10^{-2}$, and $\gamma = 5$ throughout training. For each method, we train models with three random seeds and in the end select the median run according to the validation loss.

Results In the main paper, we report aggregate metrics averaged over the subsets in Tbl. 1 and demonstrate intervention inference in Fig. 5. In the latter, we infer intervention targets the intervention encoder $q(I|x, \tilde{x})$, find the pre- and post-intervention noise encodings with the noise encoder, $e, \tilde{e} \sim q(e, \tilde{e}|x, \tilde{x}, I)$, intervene in the latent space by constructing a new noise encoding consisting of $e_{\setminus I}$ and \tilde{e}_I , and push to the data space with the decoder.

In addition to the results shown in the main paper, we show metrics for each separate subset in in Tbl. 2 and visualize the disentanglement properties of the learned representations in Fig. 8.

D.4 CausalCircuit experiments

Dataset We introduce the new CausalCircuit dataset. This environment is built in the MuJoCo simulator [25], using a model of the TriFinger robotic platform [26], of which we use a single finger. There are four causal variables: the three lights and the robot arm. The arm state is the position along an arc that goes over three buttons. The position on the arc is translated into actuator angles using an inverse kinematic model. To obtain a sample, the arm is moved away from the buttons, then lowered to the desired position, after which a rendering is recorded. Each red, green and blue button then has a pressed state b_R, b_G, b_B that depends on how far the button is touched from the center and scales linearly in the radial distance from 0 to 1. The causal model for the red, green and blue light variables and the arm z_A then is:

$$\begin{aligned} v_R &= 0.2 + 0.6 * \text{clip}(z_G + z_B + b_R, 0, 1) \\ v_G &= 0.2 + 0.6 * b_G \\ v_B &= 0.2 + 0.6 * b_B \\ z_R &\sim \text{Beta}(5v_R, 5 * (1 - v_R)) \\ z_G &\sim \text{Beta}(5v_G, 5 * (1 - v_G)) \\ z_B &\sim \text{Beta}(5v_B, 5 * (1 - v_B)) \\ z_A &\sim \text{Uniform}(0, 1) \end{aligned}$$

The scene is rendered in 512×512 pixels using the MuJoCO renderer.

Dimensionality reduction Rather than training directly on images, we found it beneficial for fast experimentation to first condense the image datasets into a lower-dimensional representation and then train both ILCMs and baselines (excluding slot attention) on that data. We used a β -VAE with a standard Gaussian prior with 16 latent dimensions; the encoder and decoder follow the same architecture as for the Causal3DIdent dataset. This VAE was trained for $8.5 \cdot 10^4$ steps using the Adam optimizer, an initial learning rate of $3 \cdot 10^{-4}$, cosine annealing, and batchsize 128.

For slot attention, we reduced the resolution to 64×64 , as we found it difficult to train the model on the full resolution 512×512 .

Architecture On this dimensionality-reduced, 16-dimensional data, we use fully connected networks for noise encoder and noise decoder, each with five hidden layers with 64 units each and ReLU activations. The solution functions have the same architecture as in the other experiments.

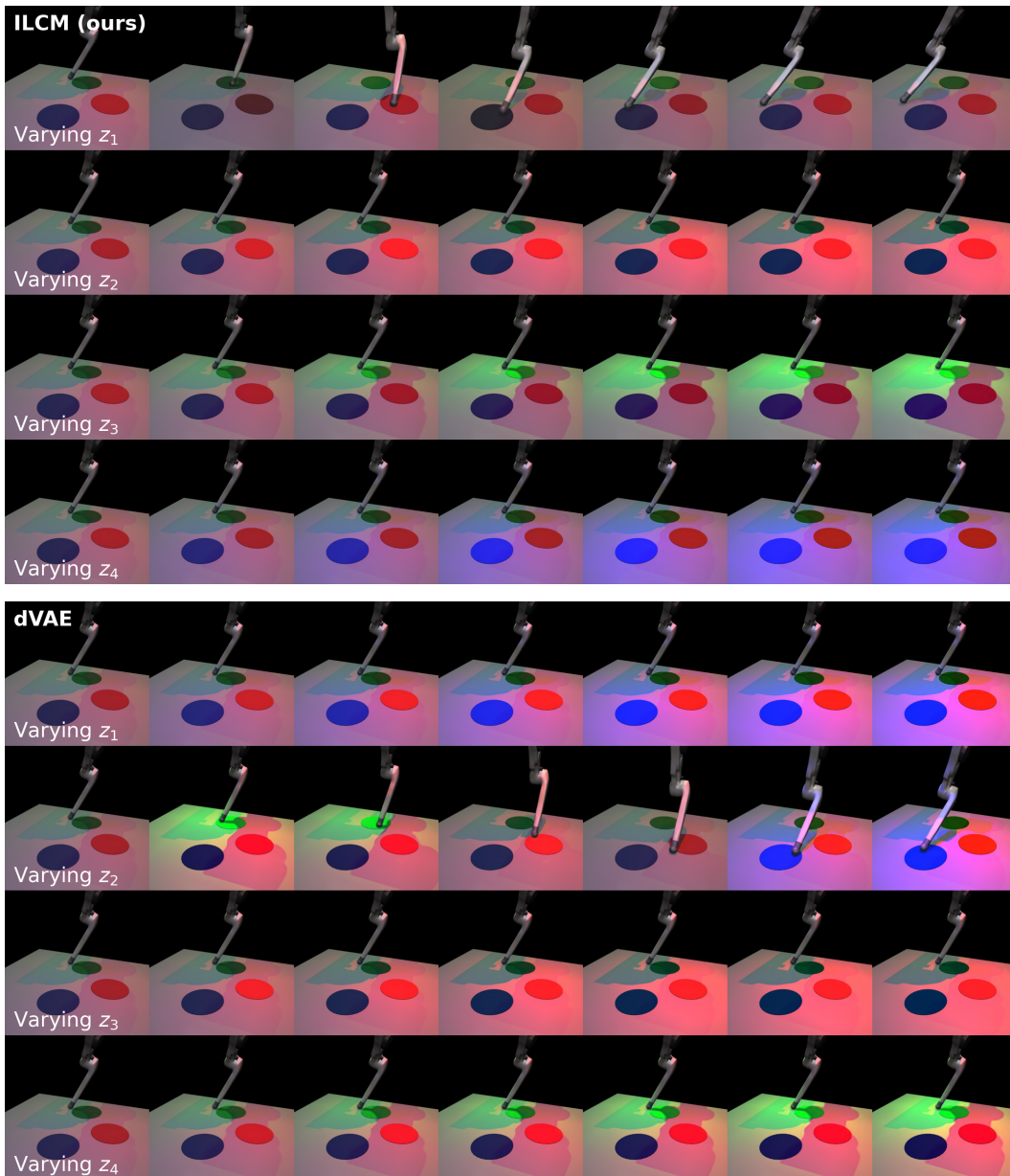


Figure 9: Effect of varying the learned causal factors on the image in the CausalCircuit dataset. We encode a single test images (left column) into the four learned causal variables. We then vary each of these causal factors in isolation (without performing interventions, that is, without including the causal effects on other variables) and show the reconstructed images. The ILCM (top) learns a representation that is quite disentangled: z_1 corresponds to the blue light, z_2 to the green light, z_3 to the robot arm position, and z_4 to the red light. In contrast, the acausal dVAE baseline entangles the different lights and the robot arm position in its learned latent factors.

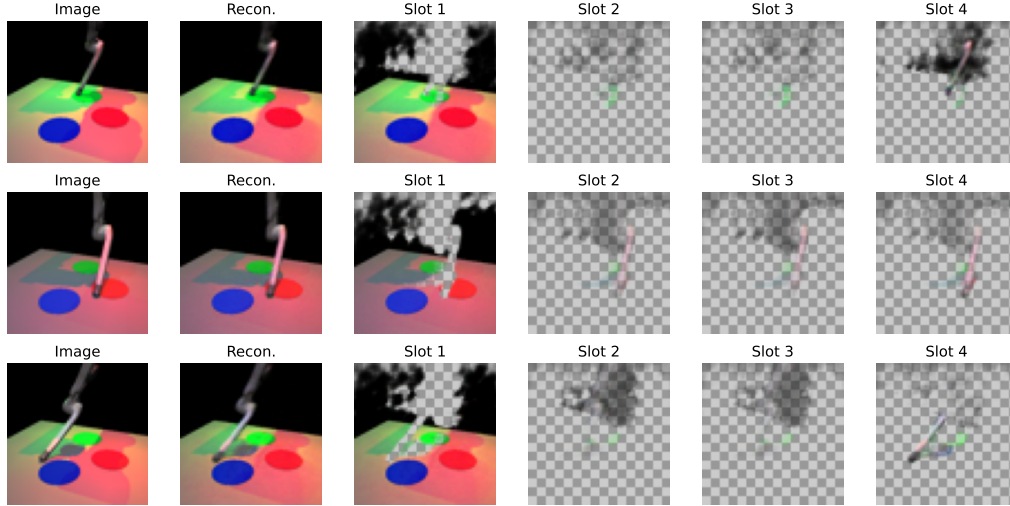


Figure 10: The slots found by slot attention. We see that the slots do not correspond to the causal variables. Only the arm is disentangled from the buttons and lights.

Training Models are trained using the procedure described in Sec. C.3. We train for $9.4 \cdot 10^4$ steps using a batch size of 64 and an initial learning rate of $3 \cdot 10^{-4}$. The weights of the different loss terms and regularizers are as follows: β is initially set to 0 and increased to its final value of $3 \cdot 10^{-4}$ during training, $\alpha = 10^{-2}$, and $\gamma = 10$ throughout training. For each method, we train models with three random seeds and in the end select the median run according to the validation loss.

Results In addition to the results in the main paper, Fig. 9 shows that our ILCM model successfully disentangled the causal factors, while the dVAE baseline failed at that task. Similarly, in Fig. 10 we see that the slot attention model fails to assign the causal variables into separate slots. We presume this is because the lights blend into each other, making them only describable by a single slot.

D.5 Graph scaling

Dataset Finally, we study the scaling of LCMs with the size of the causal graph. We generate synthetic datasets with $n \in \{2, 4, 6, 8, 10, 12, 15, 20\}$ causal variables. For each dimensionality, we generate three different datasets with different SCMs and maps to the data space. The causal graphs are generated by fixing a topological order, then for each edge consistent with the topological order we draw a Bernoulli random variable with probability $\frac{1}{2}$ to determine whether the edge exists. The SCMs are additive noise models with linear causal effects, $z_j = \epsilon_j + \sum_{i \in \text{pa}_j} a_{ij} z_i$, where $\epsilon_j \sim \mathcal{N}(0, 1)$ and the coefficients a_{ij} are drawn from a Gaussian mixture model with two equally likely components with means ± 1 and standard deviation 0.3. (Drawing the strengths of the causal effects from such a mixture model makes it unlikely to draw a causal effect very close to 0, which corresponds to a faithfulness violation.) Causal variables are mapped to the data space \mathbb{R}^n by a randomly sampled $SO(n)$ transformation.

Architecture We use fully connected networks for noise encoder and noise decoder, each with two hidden layers with 64 units each and ReLU activations. The solution functions have the same architecture as in the other experiments.

Training Models are trained using the procedure described in Sec. C.3. We train for $1.4 \cdot 10^5$ steps using a batch size of 64 and an initial learning rate of $3 \cdot 10^{-4}$. The weights of the different loss terms and regularizers are as follows: β is initially set to 0 and increased to its final value of 1 during training, $\alpha = 10^{-2}$, and $\gamma = 1$ throughout training. We find it beneficial to not fix the topological order during training. For each dataset and each method, we train models with three random seeds; in the end we take the mean over all datasets and seeds.

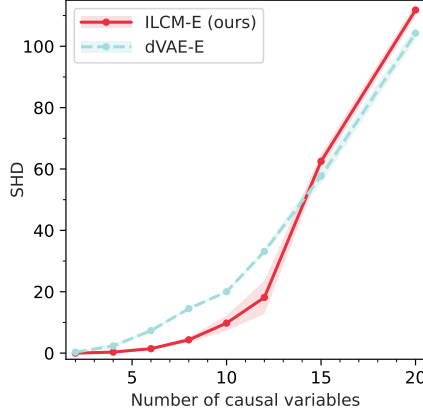


Figure 11: Scaling with graph size. We show the mean SHD between the learned and true causal graph (lower is better) and the standard error of the mean. The graphs found with ILCM-E are of a high quality for up to around 10 causal variables, but ILCM does not yet scale to even larger systems.

Results In addition to the results in the main paper, Fig. 11 shows the accuracy of the learned causal graphs as a function of the size of the system inferred with ENCO. For up to around 10 causal variables, our method lets us disentangle the causal variables reliably, and the graphs found by ILCM are more accurate than those found by the baseline. Scaling ILCMs to even larger system will require additional research.

E Explicit latent causal models

E.1 Setup

Explicit latent causal models (ELCMs) are variational autoencoders, in which the latent variables are the causal variables of an SCM. They consist of a causal encoder $q(z|x)$, decoder $p(x|z)$, and a prior $p(z, \tilde{z})$ that encodes the causal structure.

Intervention targets can either be inferred with an intervention encoder $q(I|x, \tilde{x})$, similar to our ILCM model, or be marginalized over explicitly as $p(z, \tilde{z}) = \sum_I p(I)p(z, \tilde{z}|I)$. We have experimented with both settings, for concreteness we here focus on the second, simpler setup.

The conditional prior $p(z, \tilde{z}|I)$ is the weakly supervised distribution of an SCM. They are parameterized through the causal graph \mathcal{G} , neural causal mechanisms $f_i(\epsilon_i; z_{\text{pa}_i})$, and noise base distributions. To learn the graph, the simplest option is to instantiate one LCM per graph equivalence class, train them, and select the model with the lowest validation loss. Alternatively, we can parameterize the graph in a differentiable way [11, 12, 27, 28] and learn the graph together with the other components through gradient descent.

We contrast the VAE setup of ILCMs and ELCMs in Tbl. 3.

Like ILCMs, ELCMs are trained on a VAE loss corresponding to a variational bound on $\log p(x, \tilde{x})$. Following common practice in causal discovery [11, 12, 29], we incentivize learning the sparsest graph compatible with the data distribution by adding a regularization term proportional to the number of edges in the graph to the loss.

E.2 Experiments

Dataset We experiment with ELCMs in similar datasets as we did in the main paper with ILCMs. In particular, we report results on six Causal3DIdent variations. However, we performed these datasets on an earlier iteration of these datasets: while main parameters of the scenes and the causal graphs are the same as in the experiments reported in the main paper, the ground-truth causal mechanisms are different. The metrics reported here are therefore not directly comparable to the ILCM results in the main paper.

Table 3: Differences between explicit and implicit latent causal models (ELCMs and ILCMs). Optional learnable components are shown in parentheses.

¹For simplicity, in this section we describe an ELCM implementation without intervention encoder. ²We achieved the best results when inferring and enforcing a topological order only for the last phase of ELCM training, see Sec. C.3.

	Explicit latent causal model	Implicit latent causal model
Latent variables	causal variables (z, \tilde{z}) intervention targets I	noise encodings (e, \tilde{e}) intervention targets I
Learnable components	encoder $q(z x)$ decoder $p(x z)$ (intervention encoder $q(I x, \tilde{x})$) ¹ graph \mathcal{G} causal mechanisms $f_i(\epsilon_i; z_{\text{pa}_i})$	encoder $q(e x)$ decoder $p(x e)$ (intervention encoder $q(I x, \tilde{x})$) (topological order) ² solution functions $s_i(e_i; e_{\setminus i})$

Table 4: ELCM experiments on Causal3DIdent datasets. We show the learned causal graph, the structural Hamming distance SHD between the learned and the true graph and the DCI disentanglement score (D). The datasets differ slightly from the ones used in our main experiments, so metrics are not directly comparable.

True graph	D	Learned graph	SHD
	1.00		0
	0.99		0
	0.45		3
	0.98		0
	0.98		0
	0.43		2

Hyperparameters Our ELCM architecture and training follows similar hyperparameters to our ILCM experiments. We experimented with various graph parameterizations and sampling procedures, including directed edge existence probabilities with Gumbel-Softmax sampling [12, 27], undirected edge existence probabilities and edge orientation probabilities [11], and the parameterization through edge existence probabilities and a distribution over permutations [28]. While our implementation all of these methods were able to successfully learn causal graphs given the true causal variables, we were not able to reliably learn the representations and the graph jointly. We observed a higher success rate when training separate models for different fixed DAGs and then selecting the best graph based on the validation loss. The results reported below were generated with this exhaustive graph search strategy. Again we show the median run out of three random seeds according to the validation loss.

Results In Tbl. 4 we show disentanglement scores and learned graphs. The results are mixed: in some of the datasets the graph was correctly identified and the variables are disentangled, while in others the model failed at both tasks. Notably, we find that the results strongly vary with the initialization (i. e. the random seed). In Tbl. 4 we only show the median result out of three runs, but in almost all datasets there is one random seed that lead to successful disentanglement (always with the best training and validation loss) and one random seed that led to failed disentanglement (with a worse training and validation loss). ELCM training is thus much less robust than our ILCM experiments, where the results were largely stable across random seeds.

Discussion This result hints at the presence of local minima in the loss landscape that models can get stuck in when starting from an unlucky initialization. By manually analyzing the trained ELCMs, we find that one common failure mode is that models learn variables that are to some extent disentangled and the graph has the right skeleton, but some of the causal effects are wrongly oriented (swapping cause and effect). Such graphs are often in the same Markov equivalence class as the correct graph, which is why such a model can minimize the observational contribution $-\log p(z)$ to the overall loss. Smoothly changing the representations would take the model out of the Markov equivalence class, increase this term and thus the overall loss; this configuration thus presents a

local loss minimum. The same phenomenon occurs in our experiments with differentiable graph parameterizations.

F Potential societal impact

Although we expect the immediate societal impact of this work to be negligible, more generally causal representation learning may have significant impact in the longer run. It will allow for the discovery of potential causal relationships in unstructured human-centric data. This may be beneficial, for example as it allows one to inspect if a model has learned sensible or fair causal relationships. A potential risk is that, for example because certain confounding variables are not discovered from the data, the algorithm may conclude erroneously that sensitive variables are causes of relevant outcomes. Users of the algorithms should be cautious of that.

References

- [1] Judea Pearl. *Causality : models, reasoning, and inference*. Cambridge University Press, Cambridge, U.K. New York, 2000. ISBN 978-0521895606.
- [2] Stephan Bongers, Patrick Forré, Jonas Peters, and Joris M. Mooij. Foundations of Structural Causal Models with Cycles and Latent Variables. *Annals of Statistics*, 49(5):2885–2915, 2021. doi: 10.1214/21-AOS2064.
- [3] Stack Exchange user zhwh. Lebesgue measure-preserving differentiable function. Mathematics Stack Exchange, 2016. URL <https://math.stackexchange.com/q/1755585>. (version: 2016-04-23).
- [4] Tobias Fritz. A synthetic approach to markov kernels, conditional independence and theorems on sufficient statistics. *Advances in Mathematics*, 370:107239, 2020. ISSN 0001-8708. doi: <https://doi.org/10.1016/j.aim.2020.107239>. URL <https://www.sciencedirect.com/science/article/pii/S0001870820302656>.
- [5] Brendan Fong. Causal theories: A categorical perspective on bayesian networks. *arXiv preprint arXiv:1301.6201*, 2013.
- [6] Emily Riehl. *Category Theory in Context*. Aurora: Dover Modern Math Originals. Dover Publications, 2017. ISBN 9780486820804.
- [7] 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. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.
- [8] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational Autoencoders and Nonlinear ICA: A Unifying Framework. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 2207–2217. PMLR, 2020.
- [9] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.
- [10] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. August 2016.
- [11] Phillip Lippe, Taco Cohen, and Efstratios Gavves. Efficient Neural Causal Discovery without Acyclicity Constraints. In *International Conference on Learning Representations*, 2022.
- [12] Philippe Brouillard, Sébastien Lachapelle, Alexandre Lacoste, Simon Lacoste-Julien, and Alexandre Drouin. Differentiable Causal Discovery from Interventional Data. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.
- [13] Alain Hauser and Peter Bühlmann. Characterization and Greedy Learning of Interventional Markov Equivalence Classes of Directed Acyclic Graphs. *Journal of Machine Learning Research*, 13(1):2409–2464, 2012. ISSN 1532-4435.
- [14] Thomas Kipf, Gamaleldin F Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. *arXiv preprint arXiv:2111.12594*, 2021.
- [15] Cian Eastwood and Christopher K I Williams. A framework for the quantitative evaluation of disentangled representations. *International Conference on Learning Representations*, February 2018.
- [16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [17] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning

- of disentangled representations. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4114–4124. PMLR, 2019.
- [18] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2021. <http://www.blender.org>.
- [19] Martin Edward Newell. *The Utilization of Procedure Models in Digital Image Synthesis*. PhD thesis, The University of Utah, 1975. AAI7529894.
- [20] Venkat Krishnamurthy and Marc Levoy. Fitting Smooth Surfaces to Dense Polygon Meshes. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '96*, page 313–324, New York, NY, USA, 1996. Association for Computing Machinery. ISBN 0897917464. doi: 10.1145/237170.237270.
- [21] Greg Turk and Marc Levoy. Zippered Polygon Meshes from Range Images. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '94*, page 311–318, New York, NY, USA, 1994. Association for Computing Machinery. ISBN 0897916670. doi: 10.1145/192161.192241.
- [22] Keenan Crane. Keenan’s 3D Model Repository. <https://www.cs.cmu.edu/~kmc Crane/Projects/ModelRepository/>, 2021.
- [23] Brian Curless and Marc Levoy. A Volumetric Method for Building Complex Models from Range Images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '96*, page 303–312, New York, NY, USA, 1996. Association for Computing Machinery. ISBN 0897917464. doi: 10.1145/237170.237269.
- [24] Emil Praun, Adam Finkelstein, and Hugues Hoppe. Lapped Textures. In *Proceedings of ACM SIGGRAPH 2000*, pages 465–470, July 2000.
- [25] Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, October 2012.
- [26] Manuel Wüthrich, Felix Widmaier, Felix Grimminger, Joel Akpo, Shruti Joshi, Vaibhav Agrawal, Bilal Hammoud, Majid Khadiv, Miroslav Bogdanovic, Vincent Berenz, et al. Trifinger: An open-source robot for learning dexterity. *arXiv preprint arXiv:2008.03596*, 2020.
- [27] Eric Jang, Shixiang Gu, and Ben Poole. Categorical Reparameterization with Gumbel-Softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.
- [28] Bertrand Charpentier, Simon Kibler, and Stephan Günnemann. Differentiable DAG sampling. March 2022.
- [29] Xun Zheng, Bryon Aragam, Pradeep Ravikumar, and Eric P. Xing. DAGs with NO TEARS: Continuous Optimization for Structure Learning. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9492–9503, 2018.