Compendium of Neuro-Symbolic Artificial Intelligence

Pascal Hitzler[a], Md Kamruzzaman Sarker[b], and Aaron Eberhart[a]
[a] *Kansas State University*
[b] *Kansas State University*

Contents

Chapter 1

Chapter Title: Neuro-Causal Models

Bryon Aragam, University of Chicago

Pradeep Ravikumar, Carnegie Mellon University

We describe a novel neuro-symbolic model architecture we term "neuro-causal models," that uses a synthesis of deep generative models and causal graphical models to automatically infer higher level symbolic information from lower level "raw features", while also allowing for rich relationships among the symbolic variables.

A key advance over the past decade and a half within artificial intelligence and machine learning has been the development of approaches to learn higher level representations from lower level raw input features such as image pixel intensities and word sequences [1, 2, 3, 4, 5, 6, 7]. A key advantage of these higher level representations is that they capture richer semantics with fewer variables, and accordingly, on top of which we can then learn statistically efficient models for a variety of downstream tasks such as prediction, classification, and clustering. The critical advance in recent years has been the *learning* of these representations, rather than the use of traditional handcrafted features that can be difficult to specify adequately and correctly. This has led to notable applications such as DALL-E, StableDiffusion, ChatGPT, and AudioLM, among many others.

In practice, we typically learn such representations using black-box deep neural networks, and which have led to considerable recent empirical successes. Deep generative models (DGMs) such as variational autoencoders (VAEs) [2, 3] are a prominent example of such neural approaches. These empirical successes notwithstanding, there are two caveats with such black-box neural approaches. The first is that the training of these DGMs and neural models is an intricate task: They are susceptible to posterior collapse and poor local minima [8, 9, 10, 11]. The second caveat is that it is a difficult and open problem to provide guarantees on what features will (or won't) be learned [12, 13], and in general to characterize the latent space of DGMs [14, 15]. For example, does the latent space represent semantically meaningful or practically useful features? Are the learned representations stable, or are they simply artifacts of peculiar choices of hyperparameters? These questions have been the subject of numerous studies in recent years [16, 17, 13, 18, 19, 20]. These caveats become problematic when these methods are used in high stakes settings such as medicine, health care, law, and finance, where accountability and transparency are not just desirable but often legally required.

It has thus become necessary to place representation learning on a more rigorous scientific footing. In order to do this, it is crucial to be able to discuss *ideal, target features* and the underlying representations that define these features. In other words, given the distribution over low-level raw inputs, does there exist a reproducible set of feature representations that we can recover? This is what is referred to as *identifiability*. Recently, the ML literature has turned its attention to fundamental identifiability questions [21, 22, 11], in order to move beyond consideration solely of ill-specified downstream tasks (e.g. classification, prediction, sampling, etc.).

In addition to simply extracting higher-level, possibly even conceptual, feature representations, we might also wish to understand the relationships between the objects and/or concepts underlying the feature representations, which form a core component of human reasoning, and by extension, a core component of artificial intelligence [23, 24]. Crucially, for high-dimensional data such as images, videos, and audio, the dependence between raw input features (e.g. pixels in an image) is much less relevant than the dependence between high-level, latent features (e.g. concepts or objects). Moreover, an important desideratum in high-stakes settings discussed earlier is that these relationships also be causal [25, 26]. Causal relationships are robust to perturbations, encode invariances in a system, and enable agents to reason effectively about the effects of their actions in an environment. Although deep learning—and in particular, deep generative modeling—is popular for learning latent representations, adapting deep architectures to also extract causal relationships has remained an outstanding challenge. A key hurdle is the aforementioned identifiability problem, which is an important prerequisite for interpreting learned representations causally. Although identifiability is a foundational concept in causal inference, it has only recently gained traction in the wider literature on deep generative models and representation learning. In order to interpret learned features causally, identifiability is critical.

We are thus faced with a two-fold challenge: 1) The extraction of high-level causal features from raw data, where the latent features may have general, potentially nonlinear relationships with raw input features, and 2) The inference of causal relationships between these high-level features, that in turn may have complex non-linear dependencies among them. Given such a model, we could adapt it for various causal and AI reasoning tasks, ranging over estimating the magnitude of causal effects, the effect of interventions, reasoning about counterfactuals, etc. The application of deep architectures to these problems has exploded in recent years: For estimating causal effects, see [27, 28, 29]; for causal discovery, see [30, 31, 32, 33].

A natural framework for addressing these problems is provided by causal graphical models [25, 23], which have long been used to model causal systems with hidden variables [34, 35, 36, 37, 38, 39]. It is well-known that in general, without additional assumptions, a causal graphical model given by a directed acyclic graph (DAG) is not identifiable in the presence of latent variables [23, 26]. In fact, this is a generic property of nonparametric structural models: Without assumptions, identifiability is impossible, however, given enough structure, identifiability can be rescued. Examples of this phenomenon include linearity [40, 41, 42, 43, 44], independence [45, 46, 43], rank [40, 41], sparsity [44], and graphical constraints [42, 47]. Building upon these results, in this chapter we discuss how combining graphical constraints with neural architectures points to a sweet spot of identifiability and expressivity that is well-suited to modern applications.

We draw from these ideas to develop the model architecture of neuro-causal models, with a hierarchy of conceptual latent variables, that are specified given inputs via hierarchical neural layers. We show how this neuro-symbolic structure naturally leads to identifiability guarantees by extending established ideas from causal graphical models. Together, this leads to a foundational model architecture for designing latent causal models on top of representations learned by deep generative models. We will consider a general setting for this problem that allows for arbitrary (e.g. generally nonlinear) relationships between the conceptual latent features and the raw inputs. The latent causal graph between the latent variables is also allowed to be arbitrary: No assumptions are placed on the structure of this DAG, and the number of hidden variables, their state spaces, and their relationships are entirely unknown. The idea is to provide explicit conditions under which all of this can be recovered uniquely. Our focus in this chapter will be the problem of learning causal relationships *between* latent variables, which is closely related to the problem of learning causal representations [48]. This problem should be contrasted with the equally important problem of causal inference in the presence of latent confounders [49, 50, 42, 51, 52]. To accomplish this, we blend ideas from graphical models, deep representation learning, and nonparametric statistics to address the foundations of causal representation learning as it is commonly practiced.

Outline of chapter The structure of this chapter is broken into two main parts, the first as a prerequisite for the second: After discussing background in Section 1.1, we begin in Section 1.2 by discussing the special case of so-called measurement models with discrete latents. We then generalize this to the case of continuous latents with general dependencies in Section 1.3. The former special case, while interesting in its own right, is an important subroutine for the general case, and hence is presented first.

1.1. Background

In this section, we provide brief background on the key ingredients of our neuro-causal model architecture: latent variable models, graphical models, and identifiability.

1.1.1. Latent variable models

We study latent variable models with latent variables H and observed variables X. The goal will to reconstruct some (or all) of the latent space: the number of latents $|H|$, their state spaces, and the latent distribution $P(H)$; as well as the decoder $P(X \mid H)$. Recall the elementary representation of the observed marginal $P(X)$ in terms of the latent codes and the decoder:

$$P(X) = \int P(X \mid H = h) P(H = h) \, \mathrm{d}h. \tag{1}$$

As it will be useful to distinguish discrete and continuous latents, we further decompose H as $H = (U, Z)$, where $|H| = k + m$ and:

- $U = (U_1, \ldots, U_k) \in \Omega_1 \times \cdots \times \Omega_m := \Omega$, where each Ω_i is a discrete space with $|\Omega_i| \geq 2$.
- $Z = (Z_1, \ldots, Z_m) \in \mathbb{R}^m$.

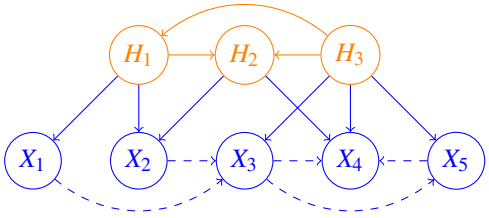


Figure 1. Illustration of a directed graphical model. The DAG G can be decomposed as $\mathsf{G} = \Gamma \cup \Gamma' \cup \Lambda$, where Γ = solid blue arrows, Γ' = dotted blue arrows, and Λ = solid orange arrows.

Throughout we will also assume that $X = (X_1, \ldots, X_n) \in \mathbb{R}^n$, although extensions to more general spaces are possible.

1.1.2. Graphical models

Our approach is based on interpreting (1) as a directed graphical model $H \to X$ with additional structure. Under now standard assumptions, this graph can be interpreted causally–see [25, 26] for details. For this, we adopt the following definitions from the graphical modeling literature.

Let $\mathsf{G} = (V, E)$ be a DAG with $V = (X, H)$. The main assumption is that there are no edges directed from any observed variable X_j to any latent variable H_i: This encodes the goal of learning hidden representations, but not latent confounders that are affected by the observed variables. Under this assumption, the graph G can be decomposed as the union of three subgraphs as follows:

$$\mathsf{G} = \Gamma \cup \Gamma' \cup \Lambda, \tag{2}$$

where

- $\Gamma =$ is the bipartite graph connecting $H \to X$;
- $\Gamma' =$ is the subgraph consisting of edges between the observables X;
- $\Lambda =$ is the subgraph consisting of edges between the latents H.

See Figure 4.

We say that a distribution $\mathbb{P}(V)$ satisfies the *Markov property* with respect to G if

$$\mathbb{P}(V) = \prod_{v \in V} \mathbb{P}(v \mid \mathrm{pa}_{\mathsf{G}}(v)). \tag{3}$$

An important consequence of the Markov property is that it allows one to read off conditional independence relations from the graph G. More specifically, we have the following [23, 26]:

- For each $v \in V$, v is independent of its non-descendants, given its parents.
- For disjoint subsets $V_1, V_2, V_3 \subset V$, if V_1 and V_2 are d-separated given V_3 in G, then $V_1 \perp\!\!\!\perp V_2 \,|\, V_3$ in $\mathbb{P}(V)$.

The concept of d-separation (see §3.3.1 in [23] or §2.3.4 in [26]) gives rise to a set of independence relations, often denoted by $\mathscr{I}(\mathsf{G})$. The Markov property thus implies that $\mathscr{I}(\mathsf{G}) \subset \mathscr{I}(V)$, where $\mathscr{I}(V)$ is the collection of all valid conditional independence relations over V for the distribution $\mathbb{P}(V)$. When the reverse inclusion holds, we say that $\mathbb{P}(V)$ is *faithful* to G (also that G is a *perfect map* of V).

Throughout this chapter, we use standard terminology and notation for ancestral relationships in a DAG, such as $\mathrm{pa}(j)$ for parents, $\mathrm{ch}(j)$ for children, and $\mathrm{ne}(j)$ for neighbors. Specifically, we define

- The parents of a node $v \in V$ are denoted by $\mathrm{pa}(v) = \{u \in V : (u,v) \in E\}$;
- The children of a node $v \in V$ are denoted by $\mathrm{ch}(v) = \{u \in V : (v,u) \in E\}$;
- The neighborhood of a node $v \in V$ is denoted by $\mathrm{ne}(v) = \mathrm{pa}(v) \cup \mathrm{ch}(v)$.

Given a subset $V' \subset V$, $\mathrm{pa}(V') := \cup_{j \in V'} \mathrm{pa}(j)$ and given a subgraph $\mathsf{G}' \subset \mathsf{G}$, $\mathrm{pa}_{\mathsf{G}'}(V') := \mathrm{pa}(V') \cap \mathsf{G}'$, with similar notation for children and neighbors. We let $A \in \{0,1\}^{|X| \times |H|}$ denote the adjacency matrix of Γ and denote its columns by $a_j \in \{0,1\}^{|X|}$. Finally, we adopt the convention that H is identified with the indices $[m] = \{1, \ldots, m\}$, and similar X is identified with $[n] = \{1, \ldots, n\}$. In particular, we use $\mathrm{pa}(i)$ and $\mathrm{pa}(H_i)$ interchangeably when the context is clear.

1.1.3. Identifiability

A statistical model is specified by a (possibly infinite-dimensional, as in our setting) parameter space Θ, a family of distributions $\mathscr{P}$, and a mapping $\pi : \Theta \to \mathscr{P}$; i.e. $\pi(\theta) \in \mathscr{P}$ for each $\theta \in \Theta$. In more conventional notation, we define $\mathscr{P} = \{p_\theta : \theta \in \Theta\}$, in which case $p_\theta = \pi(\theta)$. A statistical model is called *identifiable* if the parameter mapping π is one-to-one (injective). In practical applications, the strict definition of identifiability is too strong, and relaxed notions of identifiability are sufficient. Classical examples include identifiability up to permutation, re-scaling, or orthogonal transformation. More generally, a statistical model is *identifiable up to an equivalence relation* $\sim$ defined on Θ if $\pi(\theta) = \pi(\theta') \implies \theta \sim \theta'$.

More precisely, we use the following definition. Let $f_\sharp P$ denote the pushforward measure of P by f.

Definition 1. *Let $\mathscr{P}$ be a family of probability distributions on $\mathbb{R}^m$ and $\mathscr{F}$ be a family of functions $f : \mathbb{R}^m \to \mathbb{R}^n$.*

1. *For $(P, f) \in \mathscr{P} \times \mathscr{F}$ we say that the prior P is* identifiable (from $f_\sharp P$) up to an affine transformation *if for any $(P', f') \in \mathscr{P} \times \mathscr{F}$ such that $f_\sharp P \equiv f'_\sharp P'$ there exists an invertible affine map $h : \mathbb{R}^m \to \mathbb{R}^m$ such that $P' = h_\sharp P$ (i.e., P' is the pushforward measure of P by h).*
2. *For $(P, f) \in \mathscr{P} \times \mathscr{F}$ we say that the pair (P, f) is* identifiable (from $f_\sharp P$) up to an affine transformation *if for any $(P', f') \in \mathscr{P} \times \mathscr{F}$ such that $f_\sharp P \equiv f'_\sharp P'$ there exists an invertible affine map $h : \mathbb{R}^m \to \mathbb{R}^m$ such that $f' = f \circ h^{-1}$ and $P' = h_\sharp P$.*

This definition is extended to transformations besides affine transformations (e.g. permutations, translations, etc.) in the obvious way.

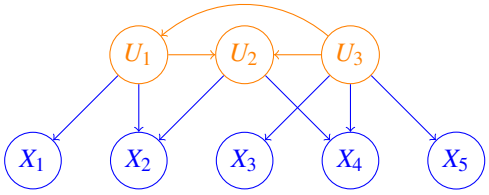


Figure 2. Measurement model with discrete latent variables U. Insert caption here.

Identifiability is a crucial primitive in machine learning tasks that is useful for probing stability, consistency, and robustness. Without identifiability, the output of a model can be unstable and unreliable, in the sense that retraining under small perturbations of the data and/or hyperparameters may result in wildly different models. In the context of deep generative models, the model output of interest is the latent space and the associated representations induced by the model as in Definition 1. The failure of identifiability, also known as *underspecification* and *ill-posedness*, has recently been flagged in the ML literature as a root cause of many failure modes that arise in practice [22, 8, 11]. As a result, there has been a growing emphasis on identification in the deep learning literature, which motivates the current work. Finally, in addition to these reproducibility and interpretability concerns, identifiability is a key component in many applications of latent variable models including causal representation learning [48], independent component analysis [53], and topic modeling [54, 55].

1.2. Neuro-Causal Models: Discrete Latents

We begin by discussing the special case of so-called *measurement models* [42, 40, 52, 43, 56, 57] with discrete latents. In a measurement model, we assume $\Gamma' = \emptyset$, i.e. there are no direct dependencies between observables. Furthermore, since the latent variables are discrete, we have $H = U$. As such, in this section we write U for the latents. See Figure 2. Although interesting in its own right, this model will serve as an important black-box in Section 1.3, where both of these assumptions will be relaxed.

The results in this section are enabled by assuming access to a so-called *mixture oracle* (see Section 1.2.2 for details), which is an oracle that returns the parameters (i.e. components, weights, and order) of a mixture model. This is a richly studied problem [58, 59, 45, 60], and in the next section we will discuss how such an oracle can be constructed for a wide class of deep generative models.

1.2.1. Assumptions

Without additional assumptions, the latent variables U cannot be identified from X. For example, we can always replace a pair of distinct hidden variables U_i and U_j with a single hidden variable U_0 that takes values in $\Omega_i \times \Omega_j$. Similarly, a single latent variable can be split into two or more latent variables. In order to avoid this type of degeneracy, we make the following assumptions:

Assumption 2 (No twins)**.** *For any hidden variables $U_i \neq U_j$ we have* $\text{ne}_\Gamma(U_i) \neq \text{ne}_\Gamma(U_j)$.

Assumption 3 (Maximality)**.** *There is no DAG* $\mathsf{G}' = ((X, U'), E')$ *such that:*

1. $\mathbb{P}(X, U')$ *is Markov with respect to* G'*;*
2. G' *is obtained from* G *by splitting a hidden variable (equivalently,* G *is obtained from* G' *by merging a pair of vertices);*
3. G' *satisfies Assumption 2.*

These assumptions are necessary for the recovery of Λ in the sense that, without these assumptions, latent variables can be created or destroyed without changing the observed distribution $\mathbb{P}(X)$ [61, 35]. Informally, the maximality assumption says that if there are several DAGs that are Markov with respect to the given distribution, we are interested in recovering the most informative among them. Finally, we need to avoid degenerate cases where certain configurations of the latent variables have zero probability:

Assumption 4 (Nondegeneracy)**.** *The distribution over* $V = (X, U)$ *satisfies:*

(a) $\mathbb{P}(U = u) > 0$ *for all* $h \in \Omega_1 \times \ldots \times \Omega_k$.
(b) For all $S \subset X$ *and* $a \neq b$, $\mathbb{P}(S|\,\mathrm{pa}(S) = a) \neq \mathbb{P}(S|\,\mathrm{pa}(S) = b)$*, where* a *and* b *are distinct configurations of* $\mathrm{pa}(S)$.

Without this nondegeneracy condition, H again cannot be identified.

1.2.2. Mixture oracles

Let $S \subset X$ be a subset of the observed variables. We can always write the marginal distribution $\mathbb{P}(S)$ as

$$\mathbb{P}(S) = \sum_{u \in \Omega} \mathbb{P}(U = u)\mathbb{P}(S\,|\,U = u). \tag{4}$$

When $S = X$, this can be interpreted as a mixture model with $K := |\Omega|$ components. When $S \subsetneq X$, however, multiple components can "collapse" onto the same component, resulting in a mixture with fewer than K components. Let $J(S)$ denote this number, so that we may define a discrete random variable G with $J(S)$ states such that for all $j \in [J(S)]$, we have

$$\mathbb{P}(S) = \sum_{j=1}^{J(S)} \underbrace{\mathbb{P}(G = j)}_{:=\pi(S,j)} \underbrace{\mathbb{P}(S\,|\,G = j)}_{:=C(S,j)} = \sum_{j=1}^{J(S)} \pi(S, j)C(S, j). \tag{5}$$

Then $\pi(S, j)$ is the weight of the jth mixture component over S, and $C(S, j)$ is the corresponding jth component. It turns out that these probabilities precisely encode the conditional independence structure of U. To make this formal, we define the following oracle:

Definition 5. *A* mixture oracle *is an oracle that takes* $S \subset X$ *as input and returns the number of components* $J(S)$ *as well as the weights* $\pi(S, j)$ *and components* $C(S, j)$ *for each* $j \in [J(S)]$*. This oracle will be denoted by* $\mathsf{MixOracle}(S)$.

A sufficient condition for the existence of a mixture oracle is that the mixture model over X is identifiable. This is because identifiability implies that the number of components K, the weights $\mathbb{P}(G = j)$, and the mixture components $\mathbb{P}(X \mid G = j)$ are determined by $\mathbb{P}(X)$. The marginal weights $\pi(S, j)$ and components $C(S, j)$ can then be recovered by simply projecting the full mixture over X onto S.

Identifiability results for mixtures are readily available in the literature. For example, if the mixture model (4) comes from any of the following families, a mixture oracle is known to exist:

1. a mixture of gaussian distributions [58, 62], or
2. a mixture of Gamma distributions [58], or
3. an exponential family mixture [62], or
4. a mixture of product distributions [63], or
5. a well-separated (i.e. in TV distance) nonparametric mixture [60].

The list above is by no means exhaustive, and many other results on identifiability of mixture models are known (e.g., see [64, 65]). The results in Section 1.3 are also based on a new identifiability result for nonparametric mixtures.

1.2.3. Recovery of the latent causal graph

Observe that the problem of learning G can be reduced to learning $(\Gamma, \mathbb{P}(U))$: Since we can decompose G into a bipartite subgraph Γ and a latent subgraph Λ (recall $\Gamma' = \emptyset$ in this section), it suffices to learn these two components separately. We then further reduce the problem of learning Λ to learning the latent distribution $\mathbb{P}(U)$. First, we will show how to reconstruct Γ from $\mathsf{MixOracle}(S)$. Then, we will show how to learn the latent distribution $\mathbb{P}(U)$ from $\mathsf{MixOracle}(S)$.

Thus, the problem of learning G is reduced to the mixture oracle:

$$\mathsf{G} \to (\Gamma, \mathbb{P}(U)) \to \mathsf{MixOracle}(S).$$

In the sequel, we focus attention on recovering $(\Gamma, \mathbb{P}(U))$. In order to recover $\mathbb{P}(U)$, we will require the following assumption, which strengthens Assumption 2:

Assumption 6 (Subset condition)**.** *We say that the bipartite graph Γ satisfies the subset condition (SSC) if for any pair of distinct hidden variables U_i, U_j the set $\mathrm{ne}_\Gamma(U_i)$ is not a subset of $\mathrm{ne}_\Gamma(U_j)$.*

This assumption is weaker than the common "anchor words" assumption from the topic modeling literature [54, 66].

Under Assumption 6, we have the following key result:

Theorem 7. *Under Assumptions 2, 3, 4, and 6, $(\Gamma, \mathbb{P}(U))$ can be reconstructed from $\mathbb{P}(X)$ and $\mathsf{MixOracle}(S)$. Furthermore, if additionally the columns of the bipartite adjacency matrix A are linearly independent, there is an efficient algorithm for this reconstruction.*

The proof is constructive and leads to an efficient algorithm as alluded to in the previous theorem. An overview of the main ideas behind the proof of this result are presented in Sections 1.2.4 and 1.2.5.

Once we know $\mathbb{P}(U)$ (e.g. via Theorem 7), identifying Λ from $\mathbb{P}(U)$ is a well-studied problem with many solutions [26, 23]. For example, if we assume that $\mathbb{P}(U)$ is faithful to Λ, then Λ can be learned up to Markov equivalence. Beyond faithfulness, any number of alternative identifiability assumptions on $\mathbb{P}(U)$ can be plugged in; e.g. triangle faithfulness [67], independent noise [68, 69], post-nonlinearity [70], equality of variances [71, 72], etc.

1.2.4. Learning the bipartite graph

In this section we outline the main ideas behind the recovery of Γ in Theorem 7. We begin by establishing conditions that ensure Γ is identifiable, and then proceed to consider efficient algorithms for its recovery.

We study a slightly more general setup in which the identifiability of Γ depends on how much information we request from the MixOracle. Clearly, we want to rely on MixOracle as little as possible. In fact, the only information required for this step are the number of components: Neither the weights nor the components are needed.

Definition 8. *We say that Γ is t-recoverable if Γ can be uniquely recovered from X and the sequence* $(\mathsf{MixOracle}(S) \mid |S| \leq t)$.

Theorem 9. *Let Γ be the bipartite graph between X and U.*

(a) Assume that $\mathrm{ne}_\Gamma(U_i) \neq \mathrm{ne}_\Gamma(U_j)$ for any $i \neq j$. Then Γ and $\dim(U_i)$ are n-recoverable.

(b) Let $t \geq 3$. Assume that for every $S \subseteq U$ with $|S| \geq 2$ we have

$$\dim \operatorname{span}\{a_j \mid j \in S\} \geq \frac{2}{t}|S| + 1,$$

then Γ and $\dim(U_i)$ are t-recoverable.

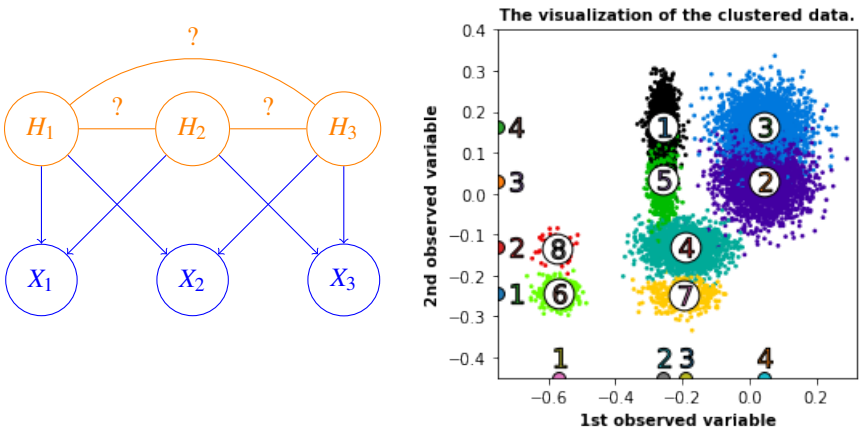
Note that Assumption 6 implies the assumption in Theorem 9(a). Finally, as in Section 1.1, we argue that in the absence of additional assumptions, this assumption is in fact necessary:

Observation 10. *If there is a pair of distinct variables $U_i, U_j \in U$ such that $\mathrm{ne}_\Gamma(U_1) = \mathrm{ne}_\Gamma(U_2)$, then Γ is not n-recoverable.*

Under a simple additional assumption Γ can be recovered efficiently. We are primarily interested in the case $t = 3$. The main idea is to reduce the problem to the recovery of a rank-three tensor involving the columns of A. We can then apply Jennrich's algorithm [73] to decompose the tensor and recover these columns, which yield Γ. To see this, let $I = (i_1, i_2, i_3) \subseteq X$ be a triple of indices, and note that

$$\sum_{j \in U} w(j)(a_j)_{i_1}(a_j)_{i_2}(a_j)_{i_3} = \left(\sum_{j \in U} w(j) a_j \otimes a_j \otimes a_j \right)_{(i_1, i_2, i_3)}. \tag{6}$$

Theorem 11. *Assume that the columns of A are linearly independent. Then Γ and $\dim(U_i)$, for all i, are 3-recoverable in $O(n^3)$ space and $O(n^4)$ time.*



(a) **(b)**

Figure 3. Example of (a) a latent causal graph and (b) its corresponding mixture distribution.

1.2.5. Learning the latent distribution

In this section we outline the main ideas behind the recovery of $\mathbb{P}(U)$ in Theorem 7.

Remark 12. *Since the variables U are not observed,* $\mathsf{MixOracle}(S)$ *only tells us the set*

$$\{(i, \pi(S,i), C(S,i)) \mid i \in [k(S)]\}.$$

But the correspondence $\Omega \ni h \leftrightarrow j \in [K]$ between a possible tuple h of values of hidden variables and the corresponding mixture component is unknown.

Since the values of U are not observed, we may learn this correspondence only up to a relabeling of Ω_i. By definition, the input distribution has $K = |\Omega|$ mixture components over X and $k_i = J(X_i)$ mixture components over X_i. Fix any enumeration of these components by $[K]$ and $[k_i]$, respectively. To recover the correspondence $\Omega \ni h \leftrightarrow j \in [K]$, we will need access to the map

$$L : [K] \to [k_1] \times \cdots \times [k_n], \tag{7}$$

defined so that $[L(j)]_i$ equals to the index of the mixture component $C(X, j)$ (marginalized over X_i) in the marginal distribution over X_i. Crucially, this discussion establishes that L can be computed from a combination of $\mathsf{MixOracle}(X)$ and $\mathsf{MixOracle}(X_i)$ for each i.

The map L encodes partial information about the causal structure in G. Indeed, if $U_1, U_2 \in \Omega$ are a pair of states of hidden variables U that coincide on $\mathrm{pa}(X_i)$ for some $X_i \in X$, then by the Markov property the components that correspond to U_1 and U_2 should have the same marginal distribution over X_i.

Example 13. *Consider the DAG on Figure 3. We do not make any assumptions about the causal structure between hidden variables. This DAG has* 3 *hidden variables, and*

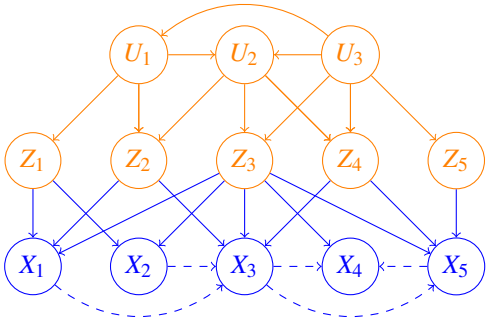


Figure 4. Deep generative model with continuous latents and a mixture prior. Insert caption here.

we assume that each of them takes values in the set $\{0,1\}$. *Then by Assumption 4, every observed variable is a mixture of* 4 *components, while the distribution on* X *is a mixture of* 8 *components. Note that the anchor word assumption is violated here, while (SSC) assumption is satisfied. The map* $L : [8] \to [4] \times [4] \times [4]$ *for an example as in Fig. 3 has form*

$$\begin{array}{cccccccccc} i: & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ L(i): & (2,4,3), & (4,3,4), & (4,4,2), & (3,2,4), & (2,3,1), & (1,1,3), & (3,1,2), & (1,2,1) \end{array}$$

The goal is to find the correspondence between $h \in \Omega = \{0,1\}^3$ *and* $i \in [8]$. *(The projection on the third variable is not shown on Figure 3, so the third coordinate of* L *cannot be deduced from the plot.)*

We now show that there is an algorithm that exactly recovers $\mathbb{P}(U)$ from the bipartite graph Γ, the map $L : [K] \to [k_1] \times \cdots \times [k_n]$, and the mixture weights (probabilities) $\{\pi(X,i) \mid i \in [K]\} = \{\mathbb{P}(Z = i) \mid i \in [K]\}$. Each of these inputs can be computed from MixOracle.

Definition 14. *Let* J *be an order-m tensor whose i-th mode is indexed by values of* U_i, *such that* $J(U_1, U_2, \ldots, U_m) = \mathbb{P}(U = u)$. *That is,* J *is the joint probability table of* U.

Theorem 15. *Suppose Assumptions 4 and 6 hold. Then the correspondence* $\Omega \ni h \leftrightarrow C(X,i)$ *and the tensor* $J(U_1, U_2, \ldots, U_m) = \mathbb{P}(U = (U_1, U_2, \ldots, U_m))$ *can be efficiently reconstructed from* L, Γ *and* $\{\pi(X,i)\}_{i \in [K]}$.

Remark 16. *If Assumption 6 is violated, then in general* J *cannot be reconstructed uniquely and moreover,* G *cannot be uniquely identified.*

1.3. Neuro-Causal Models: Continuous Latents

We now consider a generalization of the measurement model from Section 1.2 to allow for continuous latents and dependencies between observables, i.e. $\Gamma' \neq \emptyset$, as in [74]. As we shall see, this model is easily interpreted as a deep generative model that has been commonly adopted in practice.

Consider the following generative model for observations x:

$$x = f(z) + \varepsilon, \quad x = (x_1, \ldots, x_n) \in \mathbb{R}^n, \quad z = (z_1, \ldots, z_m) \in \mathbb{R}^m, \tag{8}$$

where the latent vector z follows a Gaussian mixture model (GMM),[1] $f : \mathbb{R}^m \to \mathbb{R}^n$ is a piecewise affine nonlinearity such as a ReLU network, and $\varepsilon \in \mathbb{R}^n$ is independent, random noise.[2] We do not assume that the number of mixture components, nor the architecture of the ReLU network, are known in advance, nor do we assume that z has independent components. Both the mixture model and neural network may be arbitrarily complex, and we allow for the discrete hidden state that generates the latent mixture prior to be high-dimensional and dependent. This includes both vanilla VAEs (i.e. with a standard isotropic Gaussian prior) and classical ICA models (i.e. for which the latent variables are mutually independent) as special cases. Since both z and f are allowed to be arbitrarily complex, the model (8) has universal approximation capabilities, which is crucial for modern applications in AI.

This model has been widely studied in the literature from a variety of different perspectives:

- *Nonlinear ICA.* When the z_i are mutually independent, (8) recovers the standard nonlinear ICA model that has been extensively studied in the literature [12, 75, 76, 77, 78, 79]. Although our most general results do not make independence assumptions, they cover nonlinear ICA as a special case (see Section 1.3.3 for more discussion).
- *VAE with mixture priors.* When the prior over z is a mixture model (e.g. such as a GMM), the model (8) is closely related to popular autoencoder architectures such as VaDE [80], SVAE [81], GMVAE [82], DLGMM [83], VampPrior [84], MFC-VAE [85], etc.
- *Warped mixtures.* Another closely related model is the warped mixture model of [86], which is a Bayesian version of (8).
- *iVAE.* Finally, (8) is also the basis of the iVAE model introduced by [21], where identifiability (up to certain equivalences) is proved when there is an additional auxiliary variable u that is observed such that $z_i \perp\!\!\!\perp z_j \,|\, u$.

Thus, the proposed neuro-causal architecture provides a general framework for analyzing these models, and blends practical modeling assumptions that have been adopted in practice with theoretical guarantees.

1.3.1. Assumptions

As before, the observations $x \in \mathbb{R}^n$ are realizations of a random vector X, and are generated according to the generative model (8), where $z \in \mathbb{R}^m$ represents realizations of an unobserved random vector Z. We make the following assumptions on Z and f:[3]

[1] See Remark 17 for extensions to more general mixture priors.

[2] Our results include the noiseless case $\varepsilon = 0$ as a special case.

[3] In the sequel, we will use (P#) to index assumptions on the prior $P(Z)$, and (F#) to index assumptions on the decoder f.

Assumptions on f	Assumptions on Z	Theoretical guarantees	Result
(P1)	(F1), (F2)	$\mathbb{P}(Z)$ identifiable up to an affine transformation	Theorems 22(a), 23(a)
(P1)	(F1), (F4)	$\mathbb{P}(Z)$ and f up to identifiable an affine transformation	Theorems 22(c), 23(d)
(P1), (P2)	(F1), (F4)	$\mathbb{P}(Z)$ and f identifiable up to permutation, scaling and translation	Theorems 22(b), 23(b)
(P1), (P2), (P3)	(F1), (F4)	$\mathbb{P}(U,Z)$ and f are identifiable up to permutation, scaling and translation	Theorems 23(c), 23(d)

Table 1. Summary of identifiability results. The strength of the assumptions increases in each successive row, as do the strength of the guarantees. See Section 1.3.2 for formal statements.

(P1) $P(Z)$ is a (possibly degenerate) Gaussian mixture model with an unknown number of components $J \geq 1$, i.e.

$$p(z) = \sum_{j=1}^{J} \lambda_j \varphi(z; \mu_j, \Sigma_j), \quad \sum_{j=1}^{J} \lambda_j = 1, \quad \lambda_j > 0, \tag{9}$$

where $p(z)$ is the density of $P(Z)$ with respect to some base measure, and $\varphi(z; \mu_j, \Sigma_j)$ is the gaussian density with mean μ_j and covariance Σ_j.

(F1) f is a piecewise affine function, such as a multilayer perceptron with ReLU (or leaky ReLU) activations.

Recall that an affine function is a function $x \mapsto Ax + b$ for some matrix A. As already discussed, special cases of this model have been extensively studied in both applications and theory, and both (P1)–(F1) are quite standard in the literature on deep generative models and represent a useful model that is widely used in practice [82, 85, 80, 81, 87, 88, 89, 87]. In particular, when $J = 1$ this is simply a classical VAE with an isotropic Gaussian prior (see Section 1.3.3 for more discussion).

Remark 17. *The assumption that $P(Z)$ is a GMM can be replaced with more general exponential family mixtures [90] as long as (a) the resulting mixture prior $p(z)$ is an analytic function and (b) the exponential family is closed under affine transformations.*

Remark 18. *Under assumptions (P1)–(F1), the model (8) has universal approximation capabilities. In fact, any distribution can be approximated by a mixture model (9) with sufficiently many components J [91]. Alternatively, when J is bounded, by taking f to be a sufficiently deep and/or wide ReLU network, any distribution can be approximated by $f(Z)$ [92, 93], even if f is invertible [94]. Thus, there is no loss in representational capacity in (P1)–(F1).*

For any positive integer d, let $[d] = \{1, \ldots, d\}$. By (P1), we can write the model (8) as follows. Let $U = (U_1, \ldots, U_k) \in [d_1] \times \cdots [d_k]$ where $d_i := \dim(U_i)$ and $k := \dim(U)$; we allow U to be multivariate ($k > 1$) and dependent—i.e., we do not assume that the U_i

are marginally independent. It follows trivially from (P1) that $P(U_1 = u_1, \ldots, U_k = u_k) \in \{\lambda_1, \ldots, \lambda_J\}$ and $J = \prod_i d_i$, where we recall that J is the *unknown* number of mixture components in $P(Z)$. Denote the marginal distribution of U, which depends on λ_j, by P_λ. The variables (U, Z) are unobserved and encode the underlying latent structure:

$$\left.\begin{aligned} U = u &\sim P_\lambda(U = u) \\ [Z \,|\, U = u] &\sim N(\mu_u, \Sigma_u) \\ [X \,|\, Z = z] &\sim f(z) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2) \end{aligned}\right\} \implies U \to Z \to X. \tag{10}$$

Here, P_λ is the distribution on U described above. The goal is to identify the latent distribution $P(U, Z)$ and/or the nonlinear decoder f from the marginal distribution $P(X)$ induced by (10).

Our main results (Theorems 22-23) provide a hierarchy of progressively stronger conditions under which $P(U, Z)$, f, or both, can be identified in progressively stronger ways. The idea is to illustrate explicitly what conditions are sufficient to identify the latent structure up to affine equivalence (the weakest notion of identifiability we consider), equivalence up to permutation, scaling, and translation, and permutation equivalence (the strongest notion of identifiability we consider, and the strongest possible for any latent variable model).

Possible assumptions on f: To distinguish cases where f is and is not identifiable, we require the following technical definition. Recall that for sets A, B, $f^{-1}(A) = \{x : f(x) \in A\}$ and $f(B) = \{f(x) : x \in B\}$.

Definition 19. *Let $m \leq n$ and $f : \mathbb{R}^m \to \mathbb{R}^n$.*

- *(F2) We say that f is* weakly injective *if (i) there exists $x_0 \in \mathbb{R}^n$ and $\delta > 0$ s.t. $|f^{-1}(\{x\})| = 1$ for every $x \in B(x_0, \delta) \cap f(\mathbb{R}^m)$, and (ii) $\{x \in \mathbb{R}^n : |f^{-1}(\{x\})| = \infty\} \subseteq f(\mathbb{R}^m)$ has measure zero with respect to the Lebesgue measure on $f(\mathbb{R}^m)$.*
- *(F3) We say that f is* observably injective *if $\{x \in \mathbb{R}^n : |f^{-1}(\{x\})| > 1\} \subseteq f(\mathbb{R}^m)$ has measure zero with respect to the Lebesgue measure on $f(\mathbb{R}^m)$. In other words, f is injective for almost every x in its image $f(\mathbb{R}^m)$ (i.e. almost every "observable" x).*
- *(F4) We say that f is* injective *if $|f^{-1}(\{x\})| = 1$ for every $x \in f(\mathbb{R}^m)$.*

Example 20. *In general, a deep ReLU network may be either injective or observably injective, or neither (e.g. $\mathrm{ReLU}(-\mathrm{ReLU}(x)) = 0$). For example, although $x \mapsto \mathrm{ReLU}(x)$ is not injective, it is observably injective, where $\mathrm{ReLU}(x) = \max\{0, x\}$ is the usual rectified linear unit. To see this, note that image of ReLU is the set $\mathbb{R}_{\geq} = \{y \mid y \geq 0\}$, and ReLU has the unique preimage for every $y \in \mathbb{R}_{>} = \{y \mid y > 0\}$. Clearly, $(\mathbb{R}_{\geq} \setminus \mathbb{R}_{>}) = \{0\}$ has measure zero inside $\mathbb{R}_{\geq}$. At the same time, $x \mapsto 0$ and $x \mapsto |x|$ are not even weakly injective.*

Remark 21. *Under simple assumptions on their architecture, ReLU networks or Leaky ReLU networks are generically observably injective (and hence also weakly injective) [74].*

Possible assumptions on Z: The weakest result in the identifiability hierarchy requires no additional assumptions on Z beyond (P1). Under stronger assumptions, more can be concluded. As with the previous section, the assumptions presented here are not necessary, but may be imposed in order to extract stronger results.

The first condition is a mild condition that allows us to strengthen affine identifiability:

(P2) $Z_i \perp\!\!\!\perp Z_j \mid U$ for all $i \neq j$ and there exist a pair of states $U = u_1$ and $U = u_2$ such that all $\left(\left(\Sigma_{u_1}\right)_{tt} / \left(\Sigma_{u_2}\right)_{tt} \mid t \in [m]\right)$ are distinct. (Note that this implies $J \geq 2$).

The second condition is taken from Section 1.2, and is only necessary if $k > 1$ and we wish to identify $P(U)$ in addition to $P(Z)$. Note that $P(U)$ is not needed to sample from (8), as long as we have $P(Z)$.

(P3) Assumptions 2, 3, 4, and 6 hold.

1.3.2. Main identifiability results

When $\dim(U) = 1$, there is no additional structure in U to learn, and so the setting simplifies considerably. We begin with this special case before considering the case of general multivariate U.

Theorem 22. *Assume* $\dim(U) = 1$. *Under (P1)+(F1), we have the following:*

(a) *(F2)* $\Longrightarrow$ *$P(U,Z)$ is identifiable from $P(X)$ up to an affine transformation of Z.*
(b) *(F2)+(P2)* $\Longrightarrow$ *$P(U,Z)$ is identifiable from $P(X)$ up to permutation, scaling, and/or translation of Z.*
(c) In either (a) or (b), if additionally (F4) holds and f is continuous, then f is also identifiable from $P(X)$ up to an affine transformation.

The next result generalizes Theorem 22 to arbitrary (possibly multivariate) discrete U.

Theorem 23. *Under (P1)+(F1), we have the following:*

(a) *(F2)* $\Longrightarrow$ *$P(Z)$ is identifiable from $P(X)$ up to an affine transformation.*
(b) *(F2)+(P2)* $\Longrightarrow$ *$P(Z)$ is identifiable from $P(X)$ up to permutation, scaling, and/or translation.*
(c) *(F2)+(P2)+(P3)* $\Longrightarrow$ *$(k, d_1, \ldots, d_k, P(U))$ are identifiable from $P(X)$ up to a permutation of U, and $P(Z)$ is identifiable up to permutation, scaling, and/or translation.*
(d) In any of (a), (b), or (c), if additionally (F4) holds and f is continuous, then f is also identifiable from $P(X)$ up to an affine transformation.

Without (P3), [57] have shown that it is not possible to recover the high-dimensional latent state U, however, we can still identify the continuous latent state Z, which is enough to generate random samples from the model (8). In order to have fine-grained control over the individual variables in U, however, it is necessary to assume (P3).

Remark 24. *If the assumption (F2) that f is weakly injective is removed, then the claim of Theorem 22 is not true anymore. Consider $g(x) = f(x) = |x|$ and*

$$\begin{aligned} P &= \frac{1}{3}N(-2,\sigma^2) + \frac{1}{3}N(-1,\sigma^2) + \frac{1}{3}N(3,\sigma^2) \quad \text{and} \\ P' &= \frac{1}{3}N(-2,\sigma^2) + \frac{1}{3}N(1,\sigma^2) + \frac{1}{3}N(3,\sigma^2). \end{aligned} \tag{11}$$

It is easy to verify that P cannot be transformed into P' by an affine transformation, but $f_\sharp P$ and $g_\sharp P'$ are identically distributed.

Remark 25. *In Theorems 22(a) and 23(a), the identifiability up to an affine transformation is the best possible if no additional assumptions on Z are made (i.e. beyond (P1)). Indeed, for an arbitrary invertible affine map $h : \mathbb{R}^m \to \mathbb{R}^m$, $h(Z)$ has a GMM distribution, $f \circ h^{-1}$ is an invertible piecewise affine map, and (U, Z, f) and $(U, h(Z), f \circ h^{-1})$ in model (10) generate the same distribution.*

1.3.3. Special cases and counterexamples

These results contain some notable special cases that warrant additional discussion.

Classical VAE The classical vanilla VAE [2, 3] with an isotropic Gaussian prior is equivalent to (10) with $J = 1$. In this case, U is trivial and the Gaussian distribution $P(Z)$ can be transformed by an affine map to a standard isotropic Gaussian $\mathcal{N}(0, I)$. In this case, Theorem 22(c) shows that f is identifiable from $P(X)$ up to an orthogonal transformation. In fact, this case can readily be deduced from known results on the identifiability of ReLU networks, e.g. [95].

Although the $J = 1$ case is already identifiable, there are clear reasons to prefer a clustered latent space: It is natural to model data that has several clusters by a latent space that has similar clusters (e.g. Figure 5). Although in principle any distribution can be approximated by $f(Z)$ where $Z \sim \mathcal{N}(0, I)$ and f is piecewise affine, such f is likely to be extremely complex. At the same time, the same distribution may have a representation with Z being a simple GMM and f being a simple piecewise affine function. Clearly, the latter representation is preferable to the former and can likely be more robustly learned in practice. This is consistent with previous empirical work [82, 85, 80, 81, 87, 88, 89].

Linear ICA In classical linear ICA [53], we observe $X = AZ$, where Z is assumed to have independent components. Compared to the general model (8), this corresponds to the special case where f is linear and $\varepsilon = 0$. In the most general setting under (F2) only, Theorems 22 and 23 imply that $P(Z)$ can be recovered up to an affine transformation *without* assuming independent components, which might seem surprising at first. This is, however, easily explained: In this case, X is also a GMM, and hence $P(Z)$ can already be trivially recovered up to the affine transformation $z \mapsto Az$. This follows from well-known identifiability results for GMMs [58]. This provides some intuition to how the mixture prior assumption (P1) helps to achieve identifiability.

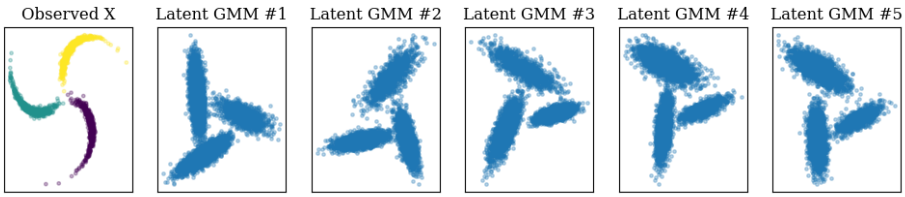


Figure 5. Recovered latent spaces for 5 runs of VaDE on pinwheel dataset with 3 clusters

Nonlinear ICA In classical nonlinear ICA, one assumes the model (8) with (a) no assumptions on f and (b) independence assumptions in the latent space. It is well-known that this model is nonidentifiable [12]. Our problem setting is distinguished from the classical nonlinear ICA model via assumptions (P1)+(F1). While we do not require the Z_i to be mutually independent, we impose assumptions on the form of f. It is precisely this inductive bias that allows us to recover identifiability. As a result, the identifiability theory developed here does not contradict known results such as the Darmois construction [96] discussed in [12].

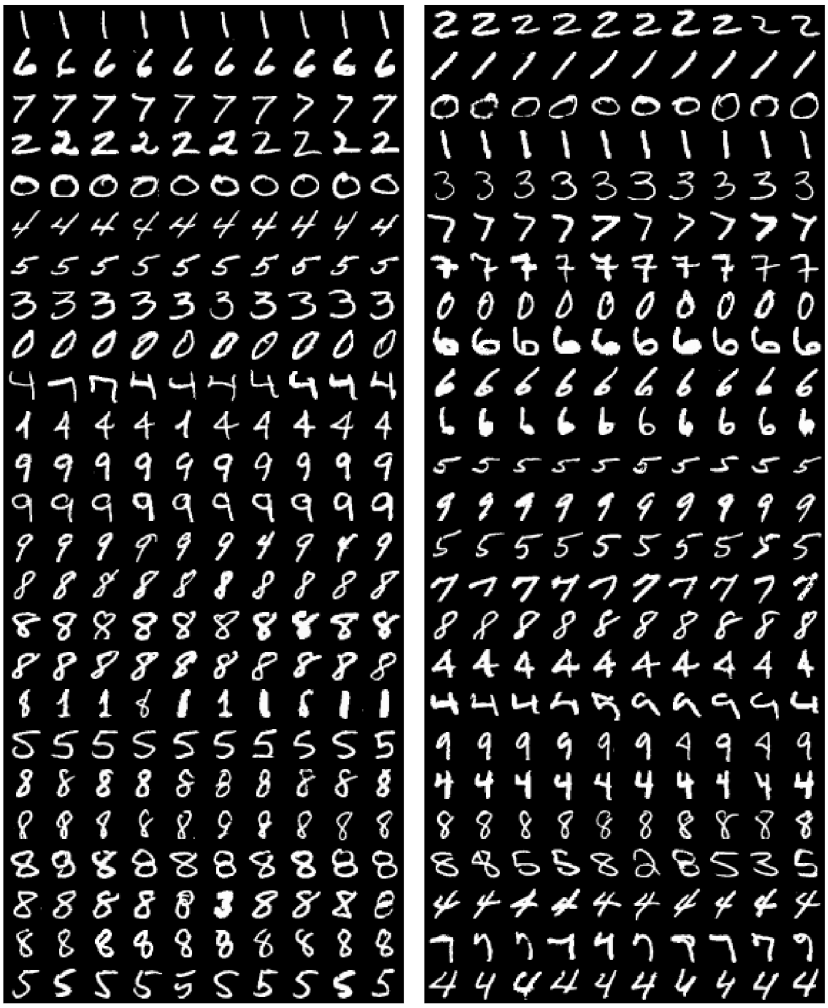
Finally, a natural question is whether or not the mixture prior (P1) or the piecewise affine nonlinearity (F1) can be relaxed while still maintaining identifiability. In fact, it is not hard to show this is not possible: If either (P1) or (F1) is broken, then the model (8) becomes nonidentifiable. Of course, this is entirely expected given known negative results on nonlinear ICA [12].

1.4. Implementation and evaluation

Although the development so far has been primarily theoretical, these ideas lead to practical algorithms and estimators that can be implemented. For full details of the algorithm implementing the discrete measurement model from Section 1.2, see [57]. We focus here on the more general setting of Section 1.3, which can be implemented as a variational autoencoder (VAE) with a Gaussian mixture prior.

There has been extensive work to verify empirically that the model (8) under (P1)+(F1) is identifiable. For example, [97] observe that deep generative models with clustered latent spaces are empirically identifiable, and compared this directly to models that rely on side information, and [85] show that meaningful latent variables can be learned consistently in a fully unsupervised manner even when they have high-dimensional structure. Moreover, [85] indicate that high-dimensional structure is important for improved performance. Beyond these, it is well-known that VAEs with mixture priors such as VaDE [80] achieve competitive performance on many benchmark tasks; see [82, 85, 81, 87, 88, 89, 87] for additional experiments and verification.

Here we briefly illustrate these methods on (misspecified) simulated models and the MNIST dataset. Figure 5 illustrates the performance of VaDE on the simulated "pinwheel" dataset. The learned latent spaces show strong evidence of recovery of the latent space up to affine transformations. This can in fact be quantified explicitly; see [74] for more details. Figure 6 further illustrates the stability of the learnt latent space by training MFCVAE [85] on MNIST 10 times with different initializations and then comparing the latent representations learnt.



(a) (b)

Figure 6. Output of MFCVAE on MNIST data: Synthetically generated samples. Each row corresponds to a different learnt component. The columns are samples generated from the component. The rows are sorted by average confidence.

1.5. Summary

Ensuring that the representations learned by deep generative models are replicable and causally interpretable is a major unresolved challenge in modern artificial intelligence. Classical approaches lean heavily on purely neural approaches, however, by combining the best of both neural and symbolic approaches leads to a rich theory of identifiable neuro-causal models with strong guarantees. This theory establishes general sufficient (and essentially necessary) conditions under which a latent causal model G is identifi-

able (Theorem 7), and leads to efficient algorithms by a reduction to a mixture oracle, which exists whenever the mixture model over X, naturally induced by symbolic latent variables, is identifiable. This further leads to a general series of results describing a hierarchy of identifiability for deep generative models that are currently used in practice, and have state-of-the-art performance on real data.

Our approach in this chapter has been purely observational: We attempt to recover latent causal structure from i.i.d. draws from the observed marginal $P(X)$. This is a notoriously challenging problem, and the limitations of learning causal structure from purely observational data are well-known [26, 25]. These challenges are exacerbated in the nonparametric setting [12, 13]. Neuro-causal models offer a suitable nonparametric framework for combining graphical constraints with neural architectures that addresses these problems directly by providing a "sweet spot" of identifiability and flexibility. Fortunately, in many applications, we have access to richer data modalities, such as weak supervision, interventions, and multiple environments. Exploiting these richer data modalities is an important direction for future work on neuro-causal models. Indeed, there has recently been an explosion of progress on these fronts which we briefly outline here for the interested reader.

One of the earliest approaches to identifiability in deep generative models was to assume we have weak supervision in the form of *auxiliary information*, which could be a time index, segment label, or environment index. This approach was pioneered in an influential paper introducing the *identifiable VAE* [21], which inspired many follow-up works [98, 99, 100, 101, 102, 103, 104, 105]. Another approach is to combine data arising from multiple environments [106, 107, 108]. A special case of different environments arises when each "environment" corresponds to a different experiment, i.e. data from different interventions [109, 110, 111, 112, 113, 114, 115, 116, 117]. Unlike the neuro-causal models introduced in this chapter, many of these results rely on parametric assumptions, although recent developments have extended these ideas to nonparametric neuro-causal models [118, 119]. Interventions can also be interpreted as a type of *mechanism shift*, and several hypotheses regarding the behaviour of mechanism shifts are known to encourage identifiability [120, 121, 122, 111, 123, 124]. These are all active and ongoing areas of research.

References

[1] Larochelle H, Murray I. The neural autoregressive distribution estimator. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics; JMLR Workshop and Conference Proceedings; 2011. p. 29–37.

[2] Kingma DP, Welling M. Auto-encoding variational bayes. arXiv preprint arXiv:13126114. 2013;.

[3] Rezende DJ, Mohamed S, Wierstra D. Stochastic backpropagation and approximate inference in deep generative models. In: Xing EP, Jebara T, editors. Proceedings of the 31st International Conference on Machine Learning; (Proceedings of Machine Learning Research; Vol. 32); 22–24 Jun; Bejing, China. PMLR; 2014. p. 1278–1286.

[4] Dinh L, Sohl-Dickstein J, Bengio S. Density estimation using real nvp. In: International Conference on Learning Representations; 2014.

[5] Goodfellow IJ, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. In: Advances in Neural Information Processing Systems; 2014. p. 2672–2680.

[6] Rezende DJ, Mohamed S. Variational inference with normalizing flows. In: International Conference on Machine Learning; 2015. p. 1530–1538.

[7] Sohl-Dickstein J, Weiss EA. Deep unsupervised learning using nonequilibrium thermodynamics. In: International Conference on Learning Representations; 2015.

[8] Yacoby Y, Pan W, Doshi-Velez F. Failure modes of variational autoencoders and their effects on downstream tasks. In: ICML Workshop on Uncertainty and Robustness in Deep Learning (UDL); 2020.

[9] Dai B, Wang Z, Wipf D. The usual suspects? reassessing blame for vae posterior collapse. In: International Conference on Machine Learning; PMLR; 2020. p. 2313–2322.

[10] He J, Spokoyny D, Neubig G, et al. Lagging inference networks and posterior collapse in variational autoencoders. In: International Conference on Learning Representations; 2018.

[11] Wang Y, Blei D, Cunningham JP. Posterior collapse and latent variable non-identifiability. Advances in Neural Information Processing Systems. 2021;34.

[12] Hyvärinen A, Pajunen P. Nonlinear independent component analysis: Existence and uniqueness results. Neural networks. 1999;12(3):429–439.

[13] Locatello F, Bauer S, Lucic M, et al. Challenging common assumptions in the unsupervised learning of disentangled representations. In: international conference on machine learning; PMLR; 2019. p. 4114–4124.

[14] Klys J, Snell J, Zemel R. Learning latent subspaces in variational autoencoders. Advances in Neural Information Processing Systems. 2018;31.

[15] Van Den Oord A, Vinyals O, et al. Neural discrete representation learning. Advances in neural information processing systems. 2017;30.

[16] Schott L, von Kügelgen J, Träuble F, et al. Visual representation learning does not generalize strongly within the same domain. arXiv preprint arXiv:210708221. 2021;.

[17] Luise G, Pontil M, Ciliberto C. Generalization properties of optimal transport gans with latent distribution learning. arXiv preprint arXiv:200714641. 2020;.

[18] Bansal Y, Nakkiran P, Barak B. Revisiting model stitching to compare neural representations. Advances in Neural Information Processing Systems. 2021;34.

[19] Csiszárik A, Kőrösi-Szabó P, Matszangosz Á, et al. Similarity and matching of neural network representations. Advances in Neural Information Processing Systems. 2021;34.

[20] Lenc K, Vedaldi A. Understanding image representations by measuring their equivariance and equivalence. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 991–999.

[21] Khemakhem I, Kingma D, Monti R, et al. Variational autoencoders and nonlinear ica: A unifying framework. In: International Conference on Artificial Intelligence and Statistics; PMLR; 2020. p. 2207–2217.

[22] D'Amour A, Heller K, Moldovan D, et al. Underspecification presents challenges for credibility in modern machine learning. arXiv preprint arXiv:201103395. 2020;.

[23] Pearl J. Probabilistic reasoning in intelligent systems: Networks of plausible inference. Morgan Kaufmann; 1988.

[24] Larrañaga P, Moral S. Probabilistic graphical models in artificial intelligence. Applied soft computing. 2011;11(2):1511–1528.

[25] Pearl J. Causality. Cambridge university press; 2009.

[26] Spirtes P, Glymour C, Scheines R. Causation, prediction, and search. Vol. 81. The MIT Press; 2000.

[27] Chernozhukov V, Chetverikov D, Demirer M, et al. Double/debiased machine learning for treatment and structural parameters. 2018;.

[28] Farrell MH, Liang T, Misra S. Deep neural networks for estimation and inference. Econometrica. 2021; 89(1):181–213.

[29] Shi C, Blei D, Veitch V. Adapting neural networks for the estimation of treatment effects. Advances in neural information processing systems. 2019;32.

[30] Zheng X, Aragam B, Ravikumar PK, et al. DAGs with NO TEARS: Continuous Optimization for Structure Learning. In: Advances in neural information processing systems 31; 2018. p. 9472–9483.

[31] Zheng X, Dan C, Aragam B, et al. Learning Sparse Nonparametric DAGs. In: Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics; (Proceedings of Machine Learning Research; Vol. 108); 2020. p. 3414–3425.

[32] Yu Y, Chen J, Gao T, et al. Dag-gnn: Dag structure learning with graph neural networks. In: International Conference on Machine Learning; PMLR; 2019. p. 7154–7163.

[33] Lachapelle S, Brouillard P, Deleu T, et al. Gradient-based neural dag learning. arXiv preprint arXiv:190602226. 2019;.

[34] Richardson T, Spirtes P, et al. Ancestral graph markov models. The Annals of Statistics. 2002; 30(4):962–1030.
[35] Evans RJ. Graphs for margins of bayesian networks. Scandinavian Journal of Statistics. 2016; 43(3):625–648.
[36] Evans RJ, Richardson TS. Markovian acyclic directed mixed graphs for discrete data. The Annals of Statistics. 2014;:1452–1482.
[37] Evans RJ, et al. Margins of discrete bayesian networks. The Annals of Statistics. 2018;46(6A):2623–2656.
[38] Evans RJ, Richardson TS, et al. Smooth, identifiable supermodels of discrete dag models with latent variables. Bernoulli. 2019;25(2):848–876.
[39] Richardson TS, Evans RJ, Robins JM, et al. Nested markov properties for acyclic directed mixed graphs. arXiv preprint arXiv:170106686. 2017;.
[40] Frot B, Nandy P, Maathuis MH. Robust causal structure learning with some hidden variables. arXiv preprint arXiv:170801151. 2017;.
[41] Chandrasekaran V, Parrilo PA, Willsky AS, et al. Latent variable graphical model selection via convex optimization. The Annals of Statistics. 2012;40(4):1935–1967.
[42] Anandkumar A, Hsu D, Javanmard A, et al. Learning linear Bayesian networks with latent variables. In: Proceedings of The 30th International Conference on Machine Learning; 2013. p. 249–257.
[43] Xie F, Cai R, Huang B, et al. Generalized independent noise condition for estimating latent variable causal graphs. Advances in Neural Information Processing Systems. 2020;33.
[44] Anderson TW. Estimating linear statistical relationships. The Annals of Statistics. 1984;:1–45.
[45] Allman ES, Matias C, Rhodes JA. Identifiability of parameters in latent structure models with many observed variables. Annals of Statistics. 2009;:3099–3132.
[46] Bonhomme S, Jochmans K, Robin JM. Estimating multivariate latent-structure models. Annals of Statistics. 2016;44(2):540–563.
[47] Anandkumar A, Valluvan R, et al. Learning loopy graphical models with latent variables: Efficient methods and guarantees. Annals of Statistics. 2013;41(2):401–435.
[48] Schölkopf B, Locatello F, Bauer S, et al. Toward causal representation learning. Proceedings of the IEEE. 2021;109(5):612–634.
[49] Colombo D, Maathuis MH, Kalisch M, et al. Learning high-dimensional directed acyclic graphs with latent and selection variables. Annals of Statistics. 2012;40(1):294–321.
[50] Spirtes PL, Meek C, Richardson TS. Causal inference in the presence of latent variables and selection bias. arXiv preprint arXiv:13024983. 2013;.
[51] Hoyer PO, Shimizu S, Kerminen AJ. Estimation of linear, non-gaussian causal models in the presence of confounding latent variables. arXiv preprint cs/0603038. 2006;.
[52] Silva R, Scheine R, Glymour C, et al. Learning the structure of linear latent variable models. Journal of Machine Learning Research. 2006;7(Feb):191–246.
[53] Comon P. Independent component analysis, a new concept? Signal processing. 1994;36(3):287–314.
[54] Arora S, Ge R, Moitra A. Learning topic models–going beyond svd. In: 2012 IEEE 53rd annual symposium on foundations of computer science; IEEE; 2012. p. 1–10.
[55] Anandkumar A, Hsu DJ, Janzamin M, et al. When are overcomplete topic models identifiable? uniqueness of tensor tucker decompositions with structured sparsity. Advances in neural information processing systems. 2013;26.
[56] Markham A, Grosse-Wentrup M. Measurement dependence inducing latent causal models. In: Conference on Uncertainty in Artificial Intelligence; PMLR; 2020. p. 590–599.
[57] Kivva B, Rajendran G, Ravikumar P, et al. Learning latent causal graphs via mixture oracles. Advances in Neural Information Processing Systems. 2021;34.
[58] Teicher H. Identifiability of finite mixtures. The annals of Mathematical statistics. 1963;:1265–1269.
[59] Frühwirth-Schnatter S. Finite mixture and markov switching models. Springer Science & Business Media; 2006.
[60] Aragam B, Dan C, Xing EP, et al. Identifiability of nonparametric mixture models and bayes optimal clustering. Ann Statist. 2020;48(4):2277–2302. ArXiv:1802.04397.
[61] Pearl J, Verma TS. A statistical semantics for causation. Statistics and Computing. 1992;2(2):91–95.
[62] Yakowitz SJ, Spragins JD. On the identifiability of finite mixtures. The Annals of Mathematical Statistics. 1968;39(1):209–214.
[63] Teicher H. Identifiability of mixtures of product measures. The Annals of Mathematical Statistics.

1967;38(4):1300–1302.
[64] McLachlan GJ, Lee SX, Rathnayake SI. Finite mixture models. Annual review of statistics and its application. 2019;6:355–378.
[65] Ritter G. Robust cluster analysis and variable selection. CRC Press; 2014.
[66] Arora S, Ge R, Halpern Y, et al. A practical algorithm for topic modeling with provable guarantees. In: International Conference on Machine Learning; PMLR; 2013. p. 280–288.
[67] Spirtes P, Zhang J. A uniformly consistent estimator of causal effects under the k-triangle-faithfulness assumption. Statistical Science. 2014;:662–678.
[68] Shimizu S, Hoyer PO, Hyvärinen A, et al. A linear non-Gaussian acyclic model for causal discovery. Journal of Machine Learning Research. 2006;7:2003–2030.
[69] Peters J, Mooij JM, Janzing D, et al. Causal discovery with continuous additive noise models. Journal of Machine Learning Research. 2014;15(1):2009–2053.
[70] Zhang K, Hyvärinen A. On the identifiability of the post-nonlinear causal model. In: Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence; AUAI Press; 2009. p. 647–655.
[71] Peters J, Bühlmann P. Identifiability of Gaussian structural equation models with equal error variances. Biometrika. 2013;101(1):219–228.
[72] Gao M, Ding Y, Aragam B. A polynomial-time algorithm for learning nonparametric causal graphs. Advances in Neural Information Processing Systems. 2020;33.
[73] Harshman RA. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis. 1970;.
[74] Kivva B, Rajendran G, Ravikumar P, et al. Identifiability of deep generative models without auxiliary information. Advances in Neural Information Processing Systems. 2022;35.
[75] Achard S, Jutten C. Identifiability of post-nonlinear mixtures. IEEE Signal Processing Letters. 2005; 12(5):423–426.
[76] Zhang K, Chan L. Minimal nonlinear distortion principle for nonlinear independent component analysis. Journal of Machine Learning Research. 2008;9(Nov):2455–2487.
[77] Hyvarinen A, Morioka H. Nonlinear ica of temporally dependent stationary sources. In: Artificial Intelligence and Statistics; PMLR; 2017. p. 460–469.
[78] Hyvarinen A, Sasaki H, Turner R. Nonlinear ica using auxiliary variables and generalized contrastive learning. In: The 22nd International Conference on Artificial Intelligence and Statistics; PMLR; 2019. p. 859–868.
[79] Hyvarinen A, Morioka H. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. Advances in Neural Information Processing Systems. 2016;29.
[80] Jiang Z, Zheng Y, Tan H, et al. Variational deep embedding: An unsupervised and generative approach to clustering. arXiv preprint arXiv:161105148. 2016;.
[81] Johnson MJ, Duvenaud DK, Wiltschko A, et al. Composing graphical models with neural networks for structured representations and fast inference. Advances in neural information processing systems. 2016;29.
[82] Dilokthanakul N, Mediano PA, Garnelo M, et al. Deep unsupervised clustering with gaussian mixture variational autoencoders. arXiv preprint arXiv:161102648. 2016;.
[83] Nalisnick E, Hertel L, Smyth P. Approximate inference for deep latent gaussian mixtures. In: NIPS Workshop on Bayesian Deep Learning; Vol. 2; 2016. p. 131.
[84] Tomczak J, Welling M. Vae with a vampprior. In: International Conference on Artificial Intelligence and Statistics; PMLR; 2018. p. 1214–1223.
[85] Falck F, Zhang H, Willetts M, et al. Multi-facet clustering variational autoencoders. Advances in Neural Information Processing Systems. 2021;34.
[86] Iwata T, Duvenaud D, Ghahramani Z. Warped mixtures for nonparametric cluster shapes. In: Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence; 2013. p. 311–320.
[87] Lee DB, Min D, Lee S, et al. Meta-gmvae: Mixture of gaussian vae for unsupervised meta-learning. In: International Conference on Learning Representations; 2020.
[88] Li X, Chen Z, Poon LK, et al. Learning latent superstructures in variational autoencoders for deep multidimensional clustering. arXiv preprint arXiv:180305206. 2018;.
[89] Willetts M, Roberts S, Holmes C. Disentangling to cluster: Gaussian mixture variational ladder autoencoders. arXiv preprint arXiv:190911501. 2019;.
[90] Barndorff-Nielsen O. Identifiability of mixtures of exponential families. Journal of Mathematical Analysis and Applications. 1965;12(1):115–121.

[91] Nguyen HD, McLachlan G. On approximations via convolution-defined mixture models. Communications in Statistics-Theory and Methods. 2019;48(16):3945–3955.

[92] Lu Y, Lu J. A universal approximation theorem of deep neural networks for expressing probability distributions. Advances in neural information processing systems. 2020;33:3094–3105.

[93] Teshima T, Ishikawa I, Tojo K, et al. Coupling-based invertible neural networks are universal diffeomorphism approximators. Advances in Neural Information Processing Systems. 2020;33:3362–3373.

[94] Ishikawa I, Teshima T, Tojo K, et al. Universal approximation property of invertible neural networks. arXiv preprint arXiv:220407415. 2022;.

[95] Stock P, Gribonval R. An embedding of relu networks and an analysis of their identifiability. arXiv preprint arXiv:210709370. 2021;.

[96] Darmois G. Analyse des liaisons de probabilité. In: Proc. Int. Stat. Conferences 1947; 1951. p. 231.

[97] Willetts M, Paige B. I don't need **u**: Identifiable non-linear ica without side information. arXiv preprint arXiv:210605238. 2021;.

[98] Hälvä H, Hyvarinen A. Hidden markov nonlinear ica: Unsupervised learning from nonstationary time series. In: Conference on Uncertainty in Artificial Intelligence; PMLR; 2020. p. 939–948.

[99] Hälvä H, Corff SL, Lehéricy L, et al. Disentangling identifiable features from noisy data with structured nonlinear ica. arXiv preprint arXiv:210609620. 2021;.

[100] Khemakhem I, Kingma DP, Monti RP, et al. Ice-beem: Identifiable conditional energy-based deep models. NeurIPS2020. 2020;.

[101] Li S, Hooi B, Lee GH. Identifying through flows for recovering latent representations. arXiv preprint arXiv:190912555. 2019;.

[102] Mita G, Filippone M, Michiardi P. An identifiable double vae for disentangled representations. In: Meila M, Zhang T, editors. Proceedings of the 38th International Conference on Machine Learning; (Proceedings of Machine Learning Research; Vol. 139); 18–24 Jul. PMLR; 2021. p. 7769–7779.

[103] Sorrenson P, Rother C, Köthe U. Disentanglement by nonlinear ica with general incompressible-flow networks (GIN). In: International Conference on Learning Representations; 2019.

[104] Yang X, Wang Y, Sun J, et al. Nonlinear ica using volume-preserving transformations. In: International Conference on Learning Representations; 2021.

[105] Klindt DA, Schott L, Sharma Y, et al. Towards nonlinear disentanglement in natural data with temporal sparse coding. In: International Conference on Learning Representations; 2020.

[106] Mooij JM, Magliacane S, Claassen T. Joint causal inference from multiple contexts. Journal of Machine Learning Research. 2020;21:1–108.

[107] Huang B, Zhang K, Zhang J, et al. Causal discovery from heterogeneous/nonstationary data. The Journal of Machine Learning Research. 2020;21(1):3482–3534.

[108] Ghassami A, Kiyavash N, Huang B, et al. Multi-domain causal structure learning in linear systems. Advances in neural information processing systems. 2018;31.

[109] Lippe P, Magliacane S, Löwe S, et al. Citris: Causal identifiability from temporal intervened sequences. In: International Conference on Machine Learning; PMLR; 2022. p. 13557–13603.

[110] Lippe P, Magliacane S, Löwe S, et al. Intervention design for causal representation learning. In: UAI 2022 Workshop on Causal Representation Learning; ????

[111] Lachapelle S, Rodriguez P, Sharma Y, et al. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ica. In: Conference on Causal Learning and Reasoning; PMLR; 2022. p. 428–484.

[112] Brehmer J, De Haan P, Lippe P, et al. Weakly supervised causal representation learning. arXiv preprint arXiv:220316437. 2022;.

[113] Zimmermann RS, Sharma Y, Schneider S, et al. Contrastive learning inverts the data generating process. In: International Conference on Machine Learning; PMLR; 2021. p. 12979–12990.

[114] Ahuja K, Hartford JS, Bengio Y. Weakly supervised representation learning with sparse perturbations. Advances in Neural Information Processing Systems. 2022;35:15516–15528.

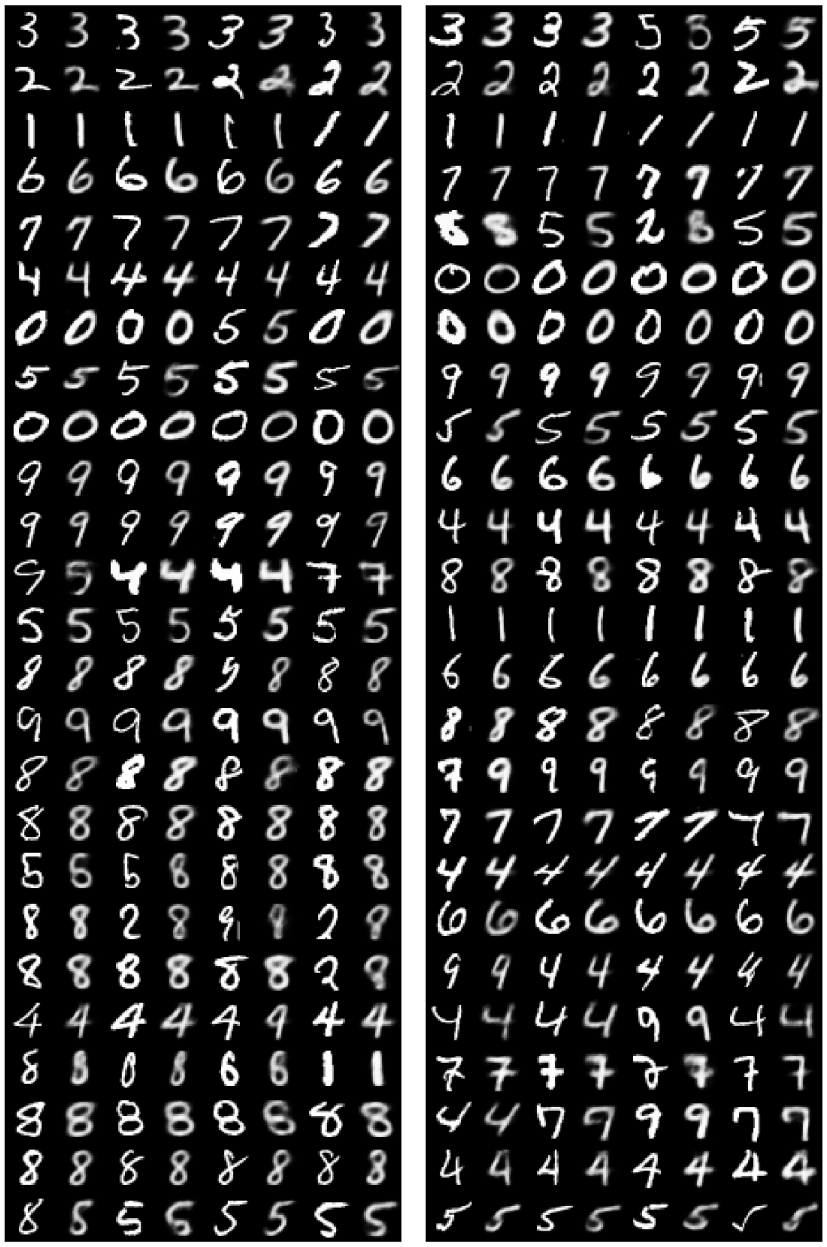
[115] Squires C, Seigal A, Bhate S, et al. Linear causal disentanglement via interventions ; 2023.

[116] Varici B, Acarturk E, Shanmugam K, et al. Score-based causal representation learning with interventions. arXiv preprint arXiv:230108230. 2023;.

[117] Ahuja K, Wang Y, Mahajan D, et al. Interventional causal representation learning. arXiv preprint arXiv:220911924. 2022;.

[118] Buchholz S, Rajendran G, Rosenfeld E, et al. Learning linear causal representations from interventions under general nonlinear mixing. arXiv preprint. 2023;.

[119] Jiang Y, Aragam B. Learning latent causal graphs with unknown interventions. arXiv preprint. 2023;.
[120] Lachapelle S, Lacoste-Julien S. Partial disentanglement via mechanism sparsity. arXiv preprint arXiv:220707732. 2022;.
[121] Sliwa J, Ghosh S, Stimper V, et al. Probing the robustness of independent mechanism analysis for representation learning. arXiv preprint arXiv:220706137. 2022;.
[122] Gresele L, Von Kügelgen J, Stimper V, et al. Independent mechanism analysis, a new concept? Advances in Neural Information Processing Systems. 2021;34.
[123] Reizinger P, Gresele L, Brady J, et al. Embrace the gap: Vaes perform independent mechanism analysis. arXiv preprint arXiv:220602416. 2022;.
[124] Perry R, Von Kügelgen J, Schölkopf B. Causal discovery in heterogeneous environments under the sparse mechanism shift hypothesis. arXiv preprint arXiv:220602013. 2022;.



(a) **(b)**

Figure 7. Output of MFCVAE on MNIST data: Reconstruction accuracy. Each row corresponds to a different learnt component, the columns correspond to 4 different pairs of x and $\hat{x}$ in that order.