Multi-Domain Causal Representation Learning via Weak Distributional Invariances

Kartik Ahuja[†]
FAIR at Meta

Amin Mansouri[†]
Mila-Quebec AI Institute

Yixin Wang
University of Michigan

Abstract

Causal representation learning has emerged as the center of action in causal machine learning research. In particular, multi-domain datasets present a natural opportunity for showcasing the advantages of causal representation learning over standard unsupervised representation learning. While recent works have taken crucial steps towards learning causal representations, they often lack applicability to multi-domain datasets due to over-simplifying assumptions about the data; e.g. each domain comes from a different single-node perfect intervention. In this work, we relax these assumptions and capitalize on the following observation: there often exists a subset of latents whose certain distributional properties (e.g., support, variance) remain stable across domains; this property holds when, for example, each domain comes from a multi-node imperfect intervention. Leveraging this observation, we show that autoencoders that incorporate such invariances can provably identify the stable set of latents from the rest across different settings.

1 Introduction

Despite the incredible success of modern AI systems, they possess limited reasoning and planning skills (Bubeck et al., 2023) and often lack controllability (Leivada et al., 2023). Towards alleviating these concerns, causal representation learning (Schölkopf et al., 2021) aims to build models with a better causal understanding of the world.

The theory of causal representation learning to date has largely focused on developing algorithms that are capable of identifying the underlying causal structure of the data-generating process under minimal supervision. This capability is enabled by endowing these learners with inductive biases that capture natural properties of the data (Locatello et al., 2020; Brehmer et al., 2022). Despite the advances, existing causal representation learners remain far from readily applicable to the increasingly prevalent multi-domain datasets in practice (Gulrajani and Lopez-Paz, 2020; Koh et al., 2021). One wonders why? An important reason is that existing approaches rely on strong assumptions about the data-generating process. For example, many assume that the data in different domains is gathered under perfect interventions. Moreover, many also require that the relationships between the latents can be described by the same fixed directed acyclic graph (DAG) across all data points. This assumption is often violated: e.g. the causal relationships between the latents can have different causal directions in two images, where a cat chases a dog in one image and the dog chases the cat in the other. In this work, we relax these assumptions, making progress towards causal representation learning for complex multi-domain datasets.

Contributions. The invariance principle considered in this paper is reminiscent of the invariance principle in Peters et al. (2016); Arjovsky et al. (2019), though we focus on unlabelled multi-domain data. At a high-level, the principle requires that a fixed subset of latents is not intervened across domains, and their distributions remain invariant. We study different forms of distributional invariance, ranging from weak invariance on the support to strong invariance on the marginal distribution of the latents. We divide our analysis into two parts. We first focus on standard settings where the latents in the entire data are governed by a fixed acyclic structural causal model; we then relax this assumption. We also consider different

[†] These authors contributed equally to this work.

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. Copyright 2024 by the author(s).

assumptions on the mixing function that generates the observations. In our theoretical and empirical analysis, the identification results take the form "*latents with invariant distributional properties can be disentangled from the rest*."

2 Related Works

The field of causal representation learning bears a deep connection to the field of independent component analysis (ICA) (Hyvarinen et al., 2023). The seminal work of Comon (1994) on linear independent component analysis studied linear mixing of independent non-Gaussian latents and proposed a method that identifies the true latents up to permutation and scaling. Since then, much progress has taken place. Existing works in the area of representation identification can be categorized into the following categories based on the assumptions: i) assumptions on the distribution of latent factors, and ii) assumptions on the mixing functions. In the pivotal work of Khemakhem et al. (2020a), the authors studied general diffeomorphisms mixing but made additional assumptions such as the availability of auxiliary information that renders latents conditionally independent. Recently, Kivva et al. (2022) considered a setup similar to Khemakhem et al. (2020a); they relaxed the crucial assumption that auxiliary information is observed but restricted the family of mixing maps to piecewise linear diffeomorphisms, in order to obtain a similar level of identification as Khemakhem et al. (2020a). The recent work of Liang et al. (2023) takes the connection between causal representation learning and ICA one step further. They study the question of identifiability under the supposition that the underlying causal graph is known, much in the same spirit that ICA supposes the graph is known and all latent variables are independent.

More recently, the problem of interventional causal representation learning has come to attention in Ahuja et al. (2022b); Seigal et al. (2022); Varici et al. (2023). Ahuja et al. (2022b) study a) polynomial mixing with interventions that induce independent support; b) general diffeomorphisms with hard do interventions. Seigal et al. (2022) study linear mixing with perfect interventions, and Varici et al. (2023) study linear mixing with perfect and imperfect interventions. The relatively recent work of von Kügelgen et al. (2023) studied general diffeomorphism mixing with perfect interventions, and Buchholz et al. (2023) studied general diffeomorphisms with latents that follow linear Gaussian structural causal model (SCM) under both perfect and imperfect interventions. The different identification guarantees in these works are summarized in Table 1, where we also contrast our results. There are a few aspects that separate us from existing works. Firstly, these works study single-node interventions and we study multi-node imperfect interventions. We also study the setting where a fixed DAG does not explain the relationships between the latents for the entire observational dataset. Another close line of work focuses on the intermediate goal of learning the underlying latent causal graph. Some examples in this line of work include Cai et al. (2019); Xie et al. (2020); Jiang and Aragam (2023) and a concurrent work (Zhang et al., 2023).

Aside from the above works, other causal representation learning settings that have been studied include settings where the learner has access to i) paired observations (e.g., data generated pre- and post-intervention) (Locatello et al., 2020; Lachapelle et al., 2022; Ahuja et al., 2022a; Lippe et al., 2022b,a; Von Kügelgen et al., 2021), ii) temporal data (Hyvarinen et al., 2019; Yao et al., 2022; Lachapelle and Lacoste-Julien, 2022; Ahuja et al., 2021), iii) multi-view data (Gresele et al., 2020) iv) other forms of auxiliary information (Khemakhem et al., 2020a,b; Hyvarinen et al., 2019), v) object-centric inductive biases (Mansouri et al., 2022; Lachapelle et al., 2023; Brady et al., 2023).

Lastly, the distributional invariances used in our work may remind the readers of the seminal works of Ganin et al. (2016); Muandet et al. (2013). There are a few notable differences: i) these works focus on domain generalization in the presence of labeled data, while we focus on the unsupervised setting, ii) these works enforce invariance of the joint distribution of all the latents, while we enforce a weaker invariance on a subset of the latents.

3 Unsupervised Multi-Domain Causal Representation Learning

Problem statement. We are given unlabelled data— a set of x's (e.g., images)—from multiple domains. Consider a domain $j \in [k]$, where k is the number of domains, $[k]$ is shorthand for $\{1, \cdots, k\}$. The latent variables $z \in \mathbb{R}^d$ in domain j are sampled from a distribution $p_Z^{(j)}$ whose support is denoted as $\mathcal{Z}^{(j)}$. These sampled latents z are then rendered by an injective mixing function $g : \mathbb{R}^d \rightarrow \mathbb{R}^n$ to generate $x \in \mathbb{R}^n$. The support of the corresponding x's in domain j is denoted as $\mathcal{X}^{(j)}$. Define the union of the support of the latents across domains as $\mathcal{Z} = \cup_{j\in[k]}\mathcal{Z}^{(j)}$ and correspondingly for the observations x's as $\mathcal{X} = \cup_{j\in[k]}\mathcal{X}^{(j)}$. The data-generating process (DGP) is formally stated below. In each domain $j \in [k]$,

$$z \sim p_Z^{(j)}, \; x \leftarrow g(z) \tag{1}$$

Table 1: Our results compared with related works. Existing works assume that the relationship between latents can be described by a fixed DAG across domains. We relax this assumption to work with general multi-domain settings.

Input data	**Assm. on p_Z**	**Assm. on g**	**Identification**
Observational	$z_i \perp z_j \vert u$, u aux info.	Diffeomorphism	Perm & scale (Khemakhem et al.)
Multi *do* intvn/node	Non-parametric	Diffeomorphism	$\approx$ Comp-wise (Ahuja et al.)
Perfect (1-node)	Linear	Linear	Comp-wise (Seigal et al.)
Perfect (1-node)	Non-parametric	Polynomial	Comp-wise (Ahuja et al.)
Perfect (1-node)	Non-parametric	Diffeomorphism	Comp-wise (Kugelgen et al.)
Imperfect (1-node)	Non-parametric	Linear	Mix consistency (Varici et al.)
Imperfect (1-node)	Non-parametric + ind support	Polynomial	Block affine (Ahuja et al.)
Imperfect (1-node)	Linear Gaussian	Diffeomorphism	Affine (Buchholz et al.)
Imperfect (multi-node)	Non-linear	Polynomial	Block affine (Theorem 3)
General multi-domain	Non-param, sup inv $\mathcal{S}$	Polynomial	Block affine (Theorem 4)
General multi-domain	Non-param, sup inv $\mathcal{S}$	Diffeomorphism	Γ^c identification (Theorem 5)
Counterfactual	Non-parametric	Diffeomorphism	Comp-wise (Brehmer et al.)

The goal of causal representation learning is *provable representation identification*, i.e. to learn an encoder function that can take in the observation x and provably output its underlying true latent z (or its desirable approximation). In practice, such an encoder is often learned via solving a reconstruction identity, $h \circ f(x) = x, \forall x \in \mathcal{X}$, where $f : \mathbb{R}^n \rightarrow \mathbb{R}^d$ and $h : \mathbb{R}^d \rightarrow \mathbb{R}^n$ are a pair of encoder and decoder that jointly satisfy the reconstruction identity. The pair (f, h) together is referred to as the autoencoder. Given the learned encoder f, the resulting representation is $\hat{z} \triangleq f(x)$, which holds the encoder's estimate of the latents. A common goal in causal representation learning is to achieve *component-wise* disentanglement, i.e., each $\hat{z}_i$ is a scalar and invertible function of some z_j, where $\hat{z}_i$ and z_j are i^{th} and j^{th} components of $\hat{z}$ and z.

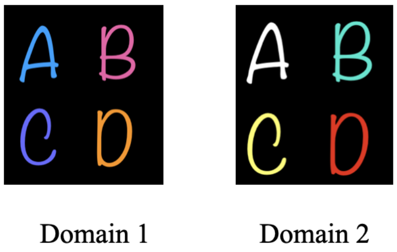

Figure 1: The distribution of the alphabet styles is stable across the domains but the distribution of color is unstable.

Invariance principle for causal representations. The invariance principle we consider here is inspired by the folklore cow-on-the-beach example (Beery et al., 2018). The distributional properties of a certain set of latents (e.g., the alphabets across domains as shown in Figure 1, or the cow characteristics across domains) are stable. In contrast, the distribution properties of the other latents (e.g. color characteristics in Figure 1) are unstable; they vary across domains. More concretely, we divide the different components of latent z into two sets, $\mathcal{S}$ and $\mathcal{U}$, where $\mathcal{S}$ corresponds to the stable set of latents and $\mathcal{U}$ corresponds to the unstable set of latents, and without loss of generality we write $z = [z_{\mathcal{S}}, z_{\mathcal{U}}]$. We require that some aspect of the joint distribution of $\mathcal{S}$—denoted as $p^{(j)}_{z_{\mathcal{S}}}$—does not vary across domains. Formally, there exists a functional F such that $F\big[p^{(j)}_{z_{\mathcal{S}}}\big]$ is invariant across j. If $F[\cdot]$ is the identity functional, then the distribution itself is invariant. Other examples of $F[\cdot]$ include the support of the latents' distributions, the mean of the latents, the variance of the latents, etc. To realize this invariance principle in causal representation learning, we study autoencoders that enforce similar invariance on a certain subset $\hat{\mathcal{S}} \subseteq [d]$ of its estimated latents $\hat{z}$:

$$h \circ f(x) = x, \qquad \forall x \in \mathcal{X}; \tag{2}$$

$$F\big[p^{(p)}_{\hat{z}_{\hat{\mathcal{S}}}}\big] = F\big[p^{(q)}_{\hat{z}_{\hat{\mathcal{S}}}}\big], \qquad \forall p \neq q, p, q \in [k]. \tag{3}$$

In what follows, we will show how autoencoders equipped with this class of invariance constraints can learn to disentangle the stable latents from the unstable latents: they return representations $\hat{z}$ that can provably satisfy $\hat{z}_{\hat{\mathcal{S}}} = u(z_{\mathcal{S}})$, where $u(\cdot)$ is an injective map. For some choice of $\hat{\mathcal{S}}$, a solution to the reconstruction identity under invariance constraint may not exist. The learner can select $\hat{\mathcal{S}}$ as follows. It can start with the largest possible $\hat{\mathcal{S}}$, i.e. a set of size d. It reduces the size of the set by one until a solution to the reconstruction identity under invariance constraint is found, which is guaranteed to occur when $|\hat{\mathcal{S}}| = |\mathcal{S}|$.

3.1 Acyclic Structural Causal Models p_z

We start with the setting where the distribution of the latents p_z comes from an acyclic causal model. To identify the stable latents, we first leverage previous results to achieve affine identification of all latents. We then use distributional invariance to achieve the identification of the stable latents. Let us now revisit a result from Ahuja et al. (2022b) for affine identification under a polynomial mixing g.

Assumption 1. *(Polynomial mixing) The interior of the support of z, denoted as $\mathcal{Z}$, is a non-empty subset of $\mathbb{R}^d$. The mixing map g is a polynomial of finite degree p whose corresponding coefficient matrix G has full column rank. Specifically, g is determined by the coefficient matrix G as follows,*

$$g(z) = G[1, z, z\bar{\otimes}z, \cdots, \underbrace{z\bar{\otimes}\cdots\bar{\otimes}\, z}_{p\ times}]^\top \qquad \forall z \in \mathbb{R}^d,$$

where $\bar{\otimes}$ represents the Kronecker product with all distinct entries; for example, if $z = [z_1, z_2]$, then $z\bar{\otimes}z = [z_1^2, z_1z_2, z_2^2]$.

Constraint 1. *(Polynomial decoder) The learned decoder h is a polynomial of degree p that is determined by its corresponding coefficient matrix H as follows,*

$$h(z) = H[1, z, z\bar{\otimes}z, \cdots, \underbrace{z\bar{\otimes}\cdots\bar{\otimes}\, z}_{p\ times}]^\top \qquad \forall z \in \mathbb{R}^d.$$

Moreover, the interior of the image of the encoder $f(\mathcal{X})$ is a non-empty subset of $\mathbb{R}^d$.

Theorem 1 (Ahuja et al. (2022b))**.** *Suppose the multi-domain data is gathered from the DGP in equation* (1) *under Assumptions 1. Then the autoencoder that solves the reconstruction identity (equation* (2)*) under Constraint 1 achieves affine identification, i.e., $\forall z \in \mathcal{Z}, \hat{z} = Az + c$, where $\hat{z}$ is the encoder f's output, z is the true latent, $A \in \mathbb{R}^{d\times d}$ is invertible and $c \in \mathbb{R}^d$.*

We now strengthen the above affine identification by using the distributional invariance of the stable set of latents. In what follows, we focus on the latents p_z that follow an acyclic structural causal model as follows. In each domain $j \in [k]$,

$$\begin{aligned} &z_i^{(j)} \leftarrow q_i\big(z_{\mathrm{Pa}(i)}^{(j)}\big) + \varrho_i^{(j)}, z_{\mathrm{Pa}(i)}^{(j)} \perp \varrho_i^{(j)}, \forall i \in [d];\\ &x \leftarrow g(z), \end{aligned} \tag{4}$$

where $q_i(\cdot)$ refers to the map that generates $z_i^{(j)}$, namely the i^{th} component of $z^{(j)}$; $\mathrm{Pa}(i)$ is the set of parents of $z_i^{(j)}$; $\varrho_i^{(j)}$ is noise in domain j. Each sampled latent is mixed by g to generate x. We drop the domain index j from $z^{(j)}$ in $x \leftarrow g(z)$ and wherever else it is not needed. We use domain index 1 to denote the observational dataset. The domains from index 2 and onwards correspond to interventional datasets. The interventions considered in this section correspond to imperfect interventions, where the mapping $q_i(\cdot)$ remains unchanged but the distribution of the noise variables changes across domains. We assume that the nodes in $\mathcal{U}$ undergo imperfect interventions, but the nodes in $\mathcal{S}$ are never intervened.

Assumption 2 (Single-node imperfect interventions)**.** *In interventional domain j ($j \geq 2$), exactly one node in $\mathcal{U}$ undergoes an imperfect intervention on the noise term. Moreover, across all domains, each node in $\mathcal{U}$ undergoes intervention at least once. Further, the children of any node in $\mathcal{U}$ must also belong to $\mathcal{U}$.*

Assumption 2 implies that the distribution of $z_{\mathcal{S}}$ remains invariant across domains. To identify $z_{\mathcal{S}}$, we thus impose the following invariance constraint: the marginal distribution of components in subset $\hat{\mathcal{S}} \subseteq [d]$ of the estimated latents must remain invariant across domains.

Constraint 2. *(Marginal invariance) For each $i \in \hat{\mathcal{S}}$, $p_{\hat{z}_i^{(p)}} = p_{\hat{z}_i^{(q)}}, \forall p \neq q, p, q \in [k]$.*

Theorem 2 (Single-node imperfect interventions)**.** *Suppose the multi-domain data is gathered from the DGP in equation* (4) *under Assumptions 1 and 2. Then the autoencoder that solves the reconstruction identity (equation* (2)*) under Constraints 1 and 2 achieves block-affine identification, i.e., $\forall z \in \mathcal{Z}, \hat{z}_{\hat{\mathcal{S}}} = Dz_{\mathcal{S}} + e$, where $\hat{z}$ is the encoder's output, z is the true latent, $D \in \mathbb{R}^{|\hat{\mathcal{S}}|\times|\mathcal{S}|}$, and $e \in \mathbb{R}^{|\hat{\mathcal{S}}|}$.*

The proof of Theorem 2 is in the Appendix. Theorem 2 implies that, under single-node imperfect interventions and polynomial mixing, the invariant latents $z_{\mathcal{S}}$ are disentangled from the rest of the latents. While the SCM (equation (4)) of the DGP in Theorem 2 does not involve any confounders, we show how this result readily extends to settings with confounders in the Appendix.

We next study multi-domain data coming from multi-node imperfect interventions. For ease of exposition, we begin with two-node imperfect interventions and assume that the noise distributions are Gaussian. We discuss how to relax these assumptions in the Appendix. Below we describe the key assumptions we make about the mechanisms underlying the interventions.

Assumption 3. *(Multi-node imperfect interventions) (1) The children of any node in $\mathcal{U}$ must also belong to $\mathcal{U}$ and the underlying DAG must have at least two terminal nodes. Further, the noise ϱ's in* (4) *are zero-mean Gaussians with variances for observational data (domain 1) sampled i.i.d. from a non-atomic density p_{σ_ϱ}.*

(2) Interventional data in each domain $j \geq 2$ is generated as follows. For each $i \in \mathcal{U}$, select a random

node j from $\mathcal{U}\setminus\{i\}$ uniformly. The noise variance for those two nodes (i,j) are two independent draws from density p_{σ_ϱ}. Repeat this procedure t times for each node $i\in\mathcal{U}$.

Theorem 3 (Multi-node imperfect interventions)**.** *Suppose the multi-domain data is gathered from the DGP in equation (4) under Assumptions 1 and 3. If the number of multi-node interventions t impacting each node is more than $\frac{\log(d/\delta)}{\log(1/(1-1/2d))}$, then, with probability $1-\delta$, the autoencoder that solves the reconstruction identity (equation (2)) under Constraints 1 and 2 achieves block-affine identification, i.e., $\forall z\in\mathcal{Z}, \hat{z}_{\hat{\mathcal{S}}} = Dz_{\mathcal{S}}+e$, where $\hat{z}$ is the encoder's output, z is the true latent, $D\in\mathbb{R}^{|\hat{\mathcal{S}}|\times|\mathcal{S}|}, e\in\mathbb{R}^{|\hat{\mathcal{S}}|}$.*

The proof of Theorem 3 is in the Appendix. Theorem 3 established that, given sufficiently many random multi-node interventions, we can block identify the stable latents $z_{\mathcal{S}}$. Moreover, the required number of domains scales as $d\Big(\frac{\log(d/\delta)}{\log(1/(1-1/2d))}\Big)$. Before closing this section, we remark that the crucial assumptions that make these results possible involve diversity of interventions and using the structure of the causal model. While we study some relaxations, we believe these results can inspire a lot of exciting future work.

3.2 General Distributions p_z

In the previous section, we made the standard assumption that the relationships between the latents z generating the data x are described by a fixed DAG. In this section, we study a relaxation that is suited to more complex multi-domain datasets, where a fixed DAG is insufficient to capture the complexities of the entire data. For example, in the cow-on-the-beach example, the relationship of the cow to its surroundings changes across samples (Beery et al., 2018). We consider a weaker invariance than one considered in the previous section, i.e., the support of each latent in the target set $\mathcal{S}$ is invariant. Under these relaxations, we prove that one can still identify the stable latents, except that the number of required domains is much larger. We will also discuss how additional assumptions can help reduce this number in the Appendix. Below we begin by stating the invariance condition. The support of z_i in domain p is denoted as $\mathcal{Z}_i^{(p)}$ and the support of estimate $\hat{z}_i$ in domain p is denoted as $\hat{\mathcal{Z}}_i^{(p)}$.

Assumption 4. *(Marginal support invariance.)*

For each $i\in\mathcal{S}$

$$\min_{z\in\mathcal{Z}_i^{(p)}} z = \min_{z\in\mathcal{Z}_i^{(q)}} z, \quad \max_{z\in\mathcal{Z}_i^{(p)}} z = \max_{z\in\mathcal{Z}_i^{(q)}}, \quad \forall p,q\in[k].$$

We now state a key assumption for the next result: there exists a pair of domains whose supports are sufficiently different. We make this notion mathematically precise below.

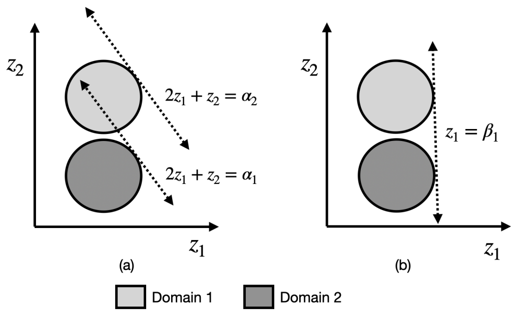

Figure 2: z_1 satisfies support invariance (Assumption 4). $[z_1,z_2]$ satisfies support variability (Assumption 5). In panel a), we show that if $\hat{z}_1$ linearly depends on both z_1 and z_2, then it achieves a different maximum value across the two domains. Thus, support invariance (Constraint 3) is not satisfied by such functions that depend on both z_1 and z_2. In contrast, the function in panel b), which only depends on z_1, achieves the same maximum across domains and satisfies support invariance.

Assumption 5 (Support variability)**.** *There exists two domains $p,q\in[k]$ such that for each $z\in\mathcal{Z}^{(p)}$, there exists a $z^{'}\in\mathcal{Z}^{(q)}$ such that $z^{'}\succcurlyeq z$, namely each component of $z^{'}$ is greater than or equal to z, i.e., $z_i^{'}\geq z_i$. Further, we require that the inequality is strict for unstable components $j\in\mathcal{U}$, $z_j^{'}>z_j$.*

We illustrate the above assumption using an example in Figure 2. The two domains shown in Figure 2 satisfy Assumption 4, 5. The latent z_1 in Domains 1 and 2 satisfies support invariance (Assumption 4). The latents $z=[z_1,z_2]$ in Domains 1 and 2 satisfy Assumption 5. We now state the invariance constraint that enforces that the latents in subset $\hat{\mathcal{S}}$ have the same minimum and maximum across domains.

Constraint 3. *(Marginal support invariance)*

For each $i\in\hat{\mathcal{S}}$,

$$\min_{z\in\hat{\mathcal{Z}}_i^{(p)}} z = \min_{z\in\hat{\mathcal{Z}}_i^{(q)}} z, \quad \max_{z\in\hat{\mathcal{Z}}_i^{(p)}} z = \max_{z\in\hat{\mathcal{Z}}_i^{(q)}}, \ \forall p,q\in[k].$$

Next, we use the above assumptions to provably identify the stable latents up to block affine transformations under polynomial mixing.

Theorem 4. *Suppose the multi-domain data is generated from equation 1 and satisfies Assumptions 1, 4, 5. Then the autoencoder that solves the reconstruction identity in equation 2 under Constraints 1 and 3 achieves the following identification guarantees:*

Each latent component $i \in \mathcal{S}$ satisfies $\hat{z}_i = A_i^\top z + c_i$, where, among all the vectors $A_i \succcurlyeq 0$, the ones that are feasible under the assumptions and constraints in this theorem must satisfy $A_{ir} = 0$ for all $r \in \mathcal{U}$.

The proof of Theorem 4 is in the Appendix.

Extending Theorem 4 beyond the positive orthant. Theorem 4 leveraged the invariance assumption (Assumption 4) to show that $\hat{z}_i$ only depends on the set of invariant latents in $\mathcal{S}$, provided that A_i's are from the positive orthant, i.e., $A_i \succcurlyeq 0$. We next extend this argument to other orthants. Consider A_i's from a different orthant with sign vector s, where each component of s corresponds to the sign of the corresponding component of A_i. We multiply z element-wise with s and denote it as $\bar{z} = z \cdot s$ and define the set of transformed latents of domain q as $\bar{\mathcal{Z}}^{(q)} = \{z \cdot s, z \in \mathcal{Z}^{(q)}\}$. If we modify Assumption 5 with set $\bar{\mathcal{Z}}^{(q)}$ instead of $\mathcal{Z}^{(q)}$, then the condition in Theorem 4 extends to all vectors A_i in orthant with sign vector s. Given this assumption, we require a pair of domains that satisfy a condition analogous to the one in Assumption 5 for each orthant. Since the total number of orthants is 2^d, the total number of domains required grows as 2^{d+1}. In Appendix A.2, we show that the number of domains required can be reduced to d under some additional structural assumptions, e.g. the support is a polytope.

In Theorem 4, we relied on the assumption that g is a polynomial. We next relax this assumption. For ease of exposition, we consider the two-variable case and present the general case in the Appendix.

Two-variable case. Consider two-dimensional z's, i.e., $z = [z_1, z_2]$. We assume that the support of the first component z_1 is invariant across domains and the support of z_2 varies across domains. For the rest of this section, we assume that z_1 and z_2 are bounded between 0 and 1 across all domains. Specifically, the support of z_1 satisfies Assumption 4 and is set to the entire interval $[0, 1]$ across domains. Recall that the support of the first component of the encoder in domain p is $\hat{\mathcal{Z}}_1^{(p)}$. Under the support invariance constraint (Constraint 3), we require that $\hat{\mathcal{Z}}_1^{(p)}$ does not vary with p. Recall $\hat{z} = f(x) = a(z)$, where $a = f \circ g$. The first component of $\hat{z}$ thus satisfies $\hat{z}_1 = a_1(z)$, where a_1 is the first component of the map a. Under this notation, we define a large class of functions Γ and show that, if the supports are sufficiently diverse, then a_1 cannot be an element of Γ, provided that the Constraint 3 is enforced – we call this Γ^c *identification.* The larger the set Γ is, the more likely $a_1(\cdot)$ is equal to a map that only depends on z_1, which is the ideal situation. In contrast, if Constrain 3 is not enforced, then all the invertible maps $a(\cdot)$ will be allowed under reconstruction identity in equation (2). Below we state the result formally.

Definition 1. *Fix some constants $\eta > 0$, $\varepsilon > 0$, and $\iota > 0$. We then define a set of functions Γ as follows. Each function $\gamma_\theta : [0, 1] \times [0, 1] \to \mathbb{R}$ in Γ satisfies i) it is parameterized by $\theta \in \Theta$, where Θ is a bounded subset of $\mathbb{R}^s$,ii) the minima of γ_θ over $[0, 1] \times [0, 1]$ lie in the ε interior of the set, i.e., in $[\varepsilon, 1 - \varepsilon] \times [\varepsilon, 1 - \varepsilon]$, and iii) there exists an interval $[\alpha^\dagger, \beta^\dagger]$ of width at least ι such that*

$$\left| \min_{z \in [0,1] \times [0,1]} \gamma_\theta(z_1, z_2) - \min_{z \in [0,1] \times [\alpha^\dagger, \beta^\dagger]} \gamma_\theta(z_1, z_2) \right| \geq \eta. \tag{5}$$

For each $(z_1, z_2) \in [0, 1] \times [0, 1]$, γ_θ is Lipschitz continuous in $\theta \in \Theta$ with Lipschitz constant L.

In simple words, Γ consists of functions γ_θ whose minima over the entire support $[0, 1] \times [0, 1]$ is η better than any other minima obtained by constraining z_2 to some interval. In particular, the functions that only depend on z_1 do not belong to Γ because the minima of such a map do not depend on z_2. A simple illustrative example of the function class Γ is as follows: $\gamma_\theta : [0, 1] \times [0, 1] \to \mathbb{R}$, $\gamma_\theta(z_1, z_2) = (z_1 - \frac{1}{2})^2 + (z_2 - \theta)^2$, where $\theta \in [\frac{1}{2}\varepsilon, 1 - \frac{3}{2}\varepsilon]$. This function has its minima over $[0, 1] \times [0, 1]$ at $(\frac{1}{2}, \theta)$. The function is Lipschitz continuous in θ for all $(z_1, z_2) \in [0, 1] \times [0, 1]$. Set $\eta = \frac{\varepsilon^2}{4}$ and $\alpha^\dagger = \theta + \frac{\varepsilon}{2}$ and $\beta^\dagger = \theta + \frac{5}{8}\varepsilon$; then the conditions in Definition 1 are satisfied. This example illustrates how these conditions are satisfied when γ_θ has one unique global minima over the region $[0, 1] \times [0, 1]$. We now state an assumption that requires that the domains are drawn at random and their supports satisfy a certain variability condition.

Assumption 6 (Support variability)**.** *The support of z_1 does not vary across domains and is fixed to be $[0, 1]$. The support of z_2 satisfies $\mathbb{P}\big(\mathcal{Z}_2^{(p)} \subseteq [\alpha, \beta]\big) \geq c_1 |(\beta - \alpha)|^l$ and $\mathbb{P}\big(\mathcal{Z}_2^{(p)} \supseteq [\kappa, 1 - \kappa]\big) \geq c_2 \kappa^r$, where l and r are some integers, c_1, c_2 are some constants and $\alpha, \beta, \kappa \in [0, 1]$.*

The first condition on z_2 in Assumption 6 states that the probability of the support of z_2 in a randomly drawn domain being contained in the interval $[\alpha, \beta]$ grows faster than a polynomial in $(|\beta - \alpha|)$. The second condition states that the support of z_2 captures the set $[\kappa, 1 - \kappa]$ with probability at least $c_2\kappa^r$. Below we give an example where these conditions are satisfied: suppose the support of z_2 is sampled as follows. Sample two random variables A and B independently from the uniform distribution over the interval $[0, 1]$. Define the upper and lower limit of the supports as $\max\{A, B\}$ and $\min\{A, B\}$ respectively. In this case, the probabilities in Assumption 6 are given as $(\beta - \alpha)^2$ and $2\kappa^2$.

The next result builds on the following insight. If we sample sufficiently many diverse domains, then it is likely that, for each map $\gamma_\theta \in \Gamma$, we encounter two domains such that the values at the minima are at least η apart as in Definition 1. Thus, $\hat{z}_1$ constructed from any member of Γ violates the support invariance constraint and thus a_1 is not in Γ.

Define $N(\delta, \varepsilon, \eta, \iota) = N_c \log\left(\frac{2N_c}{\delta}\right)\left(\frac{1}{\log\left(\frac{1}{(1-c_1\iota^l)}\right)} + \frac{1}{\log\left(\frac{1}{(1-c_2\varepsilon^r)}\right)}\right)$, with $N_c = \left(\frac{2\max_{\theta\in\Theta}\|\theta\|\sqrt{s}}{\rho}\right)^{s}$, and $\rho = \frac{\eta}{4L}$.

Theorem 5. *If we gather data generated from equation (1), where the support of z_2 for each domain is sampled i.i.d. from Assumption 6 and support of z_1 is fixed to $[0,1]$. Further, suppose the number of domains satisfies $k \geq N(\delta, \varepsilon, \eta, \iota)$. Then the set of maps $a_1(\cdot)$ that relate $\hat{z}_1$ to $[z_1, z_2]$ does not contain any function from Γ and thus achieves Γ^c identification, where $\hat{z}$ is obtained by solving the reconstruction identity (equation 2) under support invariance constraint (Constraint 3) on $\hat{z}_1$.*

The proof of Theorem 5 is in the Appendix. The results studied in this section relied on support variability assumptions. While we study some variations in the Appendix, we believe there is room for new results on multi-domain datasets that are beyond one DAG explaining the entire observational data assumption. In the previous two sections, we saw two types of mixing – a) polynomial mixing (Theorem 3,4), b) general diffeomorphisms (Theorem 5). The results in a) rely on affine identification guarantees afforded by the polynomial mixing. Under different assumptions on g that afford affine identification, the results in Theorem 3,4 can be extended. The seminal result in Donoho and Grimes (2003) established affine identification for locally isometric g. Finally, in most of our results we have obtained a block affine identification. Extending these results to achieve permutation and scaling identification is an exciting future work.

4 Learning Invariance-Constrained Representations

In this section, we describe practical criteria to learn autoencoders described in equation (2) under invariance constraints from equation (3). We will learn in two stages. In the first stage, we learn an autoencoder $(\tilde{f}, \tilde{h})$ that minimizes the reconstruction error – $\mathbb{E}\big[\|h \circ f(x) - x\|^2\big]$, where the expectation is taken over the distribution of the raw input data x. In Stage 2, we use the output of the encoder from Stage 1 denoted as $\tilde{x}$ as inputs. In many cases, this output may have an affine relationship or a more structured relationship with the true latents than the raw inputs. In Stage 2, we learn an autoencoder $(f^\star, h^\star)$ that is constrained to satisfy certain invariances described in the previous section. We enforce these constraints by adding a penalty to the standard reconstruction error in autoencoders, i.e., the learning objective takes the form

$$\mathbb{E}\big[\|h \circ f(\tilde{x}) - \tilde{x}\|^2\big] + \lambda \cdot \text{penalty}, \tag{6}$$

where the expectation is taken over the distribution of the outputs of the encoder from Stage 1, $\tilde{x}$. In Constraint 3, we require that the smallest and the largest values to satisfy invariance. The penalty corresponding to this constraint is stated as

$$\sum_{p\neq q}\sum_{i\in\hat{\mathcal{S}}}\Big(\big(\min_{z\in\tilde{\mathcal{Z}}_i^{(p)}} z - \min_{z\in\tilde{\mathcal{Z}}_i^{(q)}} z\big)^2 + \big(\max_{z\in\tilde{\mathcal{Z}}_i^{(p)}} z - \max_{z\in\tilde{\mathcal{Z}}_i^{(q)}} z\big)^2\Big), \tag{7}$$

where $\tilde{\mathcal{Z}}_i^{(p)}$ corresponds to the support of the i^{th} component of $f^\star(\tilde{x})$ in domain p. We now describe a stronger form of invariance. We can enforce the joint distribution of all components in $\hat{\mathcal{S}}$ to be invariant, which if enforced perfectly would satisfy both Constraint 2 and 3. The penalty described below measures the maximum mean discrepancy (MMD) distance between the joint distributions $\hat{z}_{\hat{\mathcal{S}}}$ across all the domains:

$$\sum_{p\neq q}\text{MMD}(p^{(p)}_{\hat{z}_{\hat{\mathcal{S}}}}, p^{(q)}_{\hat{z}_{\hat{\mathcal{S}}}}). \tag{8}$$

5 Empirical Findings

We carry out experiments to evaluate the invariance-constrained autoencoders in a host of settings that capture varying complexity of g and varying complexity of the distribution p_z. The code to reproduce the experiments can be found at `https://github.com/facebookresearch/MD-CRL`. We study four different types of mixing maps g – i) linear mixing, ii) polynomial mixing, iii) image rendering of balls, iv) unlabeled colored MNIST data. We follow Ahuja et al. (2022b) in the creation of datasets for both polynomial mixing and image rendering of balls. Unlabeled colored MNIST is inspired from labeled colored MNIST used in Arjovsky et al. (2019); note that the challenge posed by this version is significant as we do not use labels of the digits or colors while training to achieve block identification. Our multi-domain datasets respect the following invariance – the distribution of a subset $\mathcal{S}$ (e.g., digit style) of latents does not change across domains. On the other hand, the distributions of latents in $\mathcal{U}$ (e.g., digit color) undergo change across

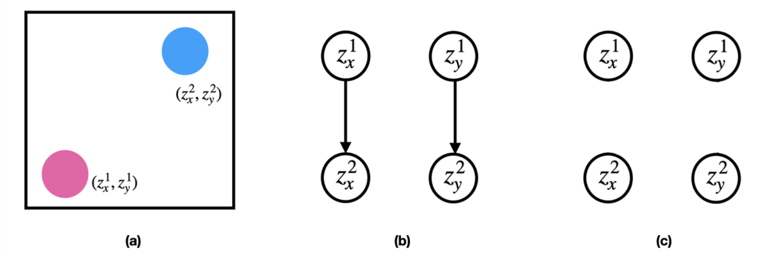

Figure 3: (a) Illustration of an image in balls image dataset. (b,c) The SCM dictating the relationship between the latents varies across data points. (b) For some samples, the coordinates of one ball depend on the other one. (c) For some samples, all causal variables are independent.

domains. We particularly induce change by changing the support of latents in $\mathcal{U}$. For each domain j with distribution $p_z^{(j)}$, we study two types of distributions – i) independent latents, ii) dependent latents. In the dependent latents data, the latents in $\mathcal{U}$ and $\mathcal{S}$ depend on each other. Further, for the dependent latents, the SCM for the latents is not fixed and it varies across data points and thus we call this setup as dynamic SCM (D-SCM). (Further details about data generation are deferred to the Appendix).

Algorithmically, we employ the two-stage procedure described in the previous section. For the linear dataset, we straight carry out Stage 2 directly, because the raw inputs are already linearly related to the true latents. However, for the polynomial and image datasets, we carry out the entire two-stage procedure. For the polynomial dataset, we carry out Stage 1 experiments with an MLP encoder and a polynomial decoder as prescribed in Ahuja et al. (2022b). For the image dataset, we carry out the Stage 1 experiment with a ResNet-based encoder and a simple ConvNet-based decoder. For both the polynomial and the image dataset, we use an MLP encoder-decoder for Stage 2. We train the Stage 2 autoencoder under three different variations of the penalty described in the previous section – i) support invariance penalty from (7) (denoted Min-Max), ii) distribution invariance penalty using MMD distance from (8) (denoted MMD), iii) combination of both support invariance and MMD based invariance (denoted MMD + Min-Max). Other experimental details can be found in the Appendix.

p_Z	Penalty	$(R^2_{\mathcal{S}}, R^2_{\mathcal{U}})$
Indep	Min-Max	$(0.90 \pm 0.01, 0.10 \pm 0.01)$
Indep	MMD	$(0.92 \pm 0.00, 0.16 \pm 0.01)$
Indep	MMD + Min-Max	$(0.94 \pm 0.01, 0.07 \pm 0.01)$
D-SCM	Min-Max	$(0.90 \pm 0.01, 0.10 \pm 0.01)$
D-SCM	MMD	$(0.92 \pm 0.00, 0.16 \pm 0.01)$
D-SCM	MMD + Min-Max	$(0.97 \pm 0.00, 0.04 \pm 0.00)$

Table 2: Comparisons for linear mixing (latent dimension $d = 32$, number of domains $k = 16$)

p_Z	Penalty	$(R^2_{\mathcal{S}}, R^2_{\mathcal{U}})$
Indep	Min-Max	$(0.91 \pm 0.01, 0.02 \pm 0.00)$
Indep	MMD	$(0.93 \pm 0.01, 0.02 \pm 0.00)$
Indep	MMD + Min-Max	$(0.93 \pm 0.01, 0.02 \pm 0.00)$
D-SCM	Min-Max	$(0.93 \pm 0.00, 0.01 \pm 0.00)$
D-SCM	MMD	$(0.95 \pm 0.00, 0.02 \pm 0.00)$
D-SCM	MMD + Min-Max	$(0.95 \pm 0.00, 0.01 \pm 0.00)$

Table 3: Comparisons for polynomial mixing (latent dimension $d = 14$, polynomial degree 3, number of domains $k = 16$).

p_Z	Penalty	$(R^2_{\mathcal{S}}, R^2_{\mathcal{U}})$
Indep	Min-Max	$(0.65 \pm 0.01, 0.19 \pm 0.01)$
Indep	MMD	$(0.63 \pm 0.04, 0.27 \pm 0.05)$
Indep	MMD + Min-Max	$(0.81 \pm 0.04, 0.18 \pm 0.02)$
D-SCM	Min-Max	$(0.61 \pm 0.03, 0.22 \pm 0.01)$
D-SCM	MMD	$(0.55 \pm 0.12, 0.15 \pm 0.04)$
D-SCM	MMD + Min-Max	$(0.82 \pm 0.02, 0.20 \pm 0.04)$

Table 4: Comparisons for ball-images dataset (number of domains $k = 16$).

We evaluate the block affine identification of the models as follows. We predict $z_{\mathcal{S}}$ from $\hat{z}_{\hat{\mathcal{S}}}$ using a linear model and compute the R^2, which we denote as $R^2_{\mathcal{S}}$. We also predict $z_{\mathcal{U}}$ from $\hat{z}_{\hat{\mathcal{S}}}$ using a linear model and compute the coefficient of determination R^2, which is denoted as $R^2_{\mathcal{U}}$. Here $\hat{\mathcal{S}}$ and $\hat{\mathcal{U}}$ are the set of latents on which invariance constraints are enforced and the set of latents on which no such constraints are enforced. High $R^2_{\mathcal{S}}$ and low $R^2_{\mathcal{U}}$ indicates block identification of the latents. For the unlabeled colored MNIST dataset, we do not have access to the z corresponding to the digits. However, we have access to the labels of the digits for evaluation purposes. On this dataset, we predict the digit from $\hat{z}_{\hat{\mathcal{S}}}$ and predict the color from $\hat{z}_{\hat{\mathcal{S}}}$. We denote the accuracy of digit prediction as Acc_{digits} and R^2 for predicting color as R^2_{color}.

p_Z	Penalty	$(Acc_{\text{digits}}, R^2_{\text{color}})$
Indep	Min-Max	$(0.66 \pm 0.01, 0.49 \pm 0.02)$
Indep	MMD	$(0.73 \pm 0.01, 0.63 \pm 0.02)$
Indep	MMD + Min-Max	$(0.74 \pm 0.01, 0.28 \pm 0.01)$
D-SCM	Min-Max	$(0.53 \pm 0.01, 0.43 \pm 0.02)$
D-SCM	MMD	$(0.75 \pm 0.01, 0.65 \pm 0.02)$
D-SCM	MMD + Min-Max	$(0.72 \pm 0.02, 0.31 \pm 0.03)$

Table 5: Comparisons for unlabeled colored MNIST dataset (number of domains $k = 16$).

g	Domains	$(R^2_{\mathcal{S}}, R^2_{\mathcal{U}})$
Linear	2	$(0.33 \pm 0.01, 0.46 \pm 0.03)$
Linear	16	$(0.97 \pm 0.00, 0.04 \pm 0.00)$
Polynomial	2	$(0.58 \pm 0.02, 0.07 \pm 0.01)$
Polynomial	16	$(0.95 \pm 0.00, 0.01 \pm 0.00)$
Ball-images	2	$(0.73 \pm 0.01, 0.35 \pm 0.02)$
Ball-images	16	$(0.82 \pm 0.02, 0.20 \pm 0.04)$

Table 6: Results under varying number of domains.

g	Domains	$(Acc_{\text{digits}}, R^2_{\text{color}})$
Unlabel CMNIST	2	$(0.73 \pm 0.02, 0.73 \pm 0.02)$
Unlabel CMNIST	16	$(0.74 \pm 0.01, 0.28 \pm 0.02)$

Table 7: Results under varying number of domains.

In Tables 2 to 4, we show the results (averaged over five seeds) for independent latents and correlated latents (D-SCM) under linear mixing, polynomial mixing, and ball image rendering. For both linear and polynomial mixing, we find that all three types of penalties work well, i.e., the learned $\hat{z}_{\hat{\mathcal{S}}}$ achieves block affine disentanglement. For the ball-images dataset, we find that the combination of the MMD + Min-Max penalty works the best. In Table 5, we show the results for unlabeled colored MNIST dataset. Here we can see that the combination of the two penalties works much better as well. One important fact to underscore here is that unlabeled colored MNIST is more challenging than balls dataset and separation of color and digit attributes is even more non-trivial. Our approach achieves a noticeable degree of disentanglement in this setting without any supervision, which is quite remarkable given the challenge posed by this setting. In addition, Tables 6 and 7 illustrate the role of the number of domains in identification. We find that increasing the number of domains helps achieve better identification; the number of required domains to achieve useful identification is less than the worst-case requirements in the theorems.

6 Conclusions

In this work, we advance the theory of multi-domain causal representation learning, making it applicable to multi-domain datasets from complex domain shifts (including multi-node imperfect interventions and beyond). We consider a simple invariance principle, namely certain distributional properties of the target latents remain invariant across domains. Following this invariance principle, we propose a class of autoencoders that enforce such weak distributional invariances. We establish identification guarantees of the stable latents for different invariances, ranging from weak invariance of the support to the stronger invariance on the marginal. To conclude, we would like to emphasize that the family of invariance constraints studied here are weaker than those in standard self-supervised learning (SSL) (Von Kügelgen et al., 2021). In SSL, we often require access to pairs of observations, where a portion of the latents (referred to as content) remains invariant between the samples. In contrast, we do not have access to such pairs; instead, we have access to domains where a subset of latents shares some invariant distributional properties. Hence, one can view the principles introduced here as a generalization of ideas in SSL, but applied to pairs of domains instead of pairs of observations.

Acknowledgements

Yixin Wang was supported in part by the Office of Naval Research under grant number N00014-23-1-2590 and the National Science Foundation under Grant No. 2231174 and No. 2310831. Amin Mansouri was supported in part by the support from Canada CIFAR AI Chair Program and from the Canada Excellence Research Chairs Program. This research was enabled in part by compute resources provided by Mila, Quebec AI Institute (mila.quebec).

References

Ahuja, K., Hartford, J., and Bengio, Y. (2021). Properties from mechanisms: an equivariance perspective on identifiable representation learning. *arXiv preprint arXiv:2110.15796.*

Ahuja, K., Hartford, J., and Bengio, Y. (2022a). Weakly supervised representation learning with sparse perturbations. *arXiv preprint arXiv:2206.01101.*

Ahuja, K., Wang, Y., Mahajan, D., and Bengio, Y. (2022b). Interventional causal representation learning. *arXiv preprint arXiv:2209.11924.*

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz,

D. (2019). Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Beery, S., Van Horn, G., and Perona, P. (2018). Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pages 456–473.

Brady, J., Zimmermann, R. S., Sharma, Y., Schölkopf, B., von Kügelgen, J., and Brendel, W. (2023). Provably learning object-centric representations. *arXiv preprint arXiv:2305.14229*.

Brehmer, J., De Haan, P., Lippe, P., and Cohen, T. (2022). Weakly supervised causal representation learning. *arXiv preprint arXiv:2203.16437*.

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. (2023). Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Buchholz, S., Rajendran, G., Rosenfeld, E., Aragam, B., Schölkopf, B., and Ravikumar, P. (2023). Learning linear causal representations from interventions under general nonlinear mixing. *arXiv preprint arXiv:2306.02235*.

Cai, R., Xie, F., Glymour, C., Hao, Z., and Zhang, K. (2019). Triad constraints for learning causal structure of latent variables. *Advances in neural information processing systems*, 32.

Comon, P. (1994). Independent component analysis, a new concept? *Signal processing*, 36(3):287–314.

Donoho, D. L. and Grimes, C. (2003). Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100(10):5591–5596.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016). Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030.

Gresele, L., Rubenstein, P. K., Mehrjou, A., Locatello, F., and Schölkopf, B. (2020). The incomplete rosetta stone problem: Identifiability results for multi-view nonlinear ica. In *Uncertainty in Artificial Intelligence*, pages 217–227. PMLR.

Gulrajani, I. and Lopez-Paz, D. (2020). In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*.

Hyvarinen, A., Khemakhem, I., and Morioka, H. (2023). Nonlinear independent component analysis for principled disentanglement in unsupervised deep learning. *arXiv preprint arXiv:2303.16535*.

Hyvarinen, A., Sasaki, H., and Turner, R. (2019). Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 859–868. PMLR.

Jiang, Y. and Aragam, B. (2023). Learning nonparametric latent causal graphs with unknown interventions. *arXiv preprint arXiv:2306.02899*.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. (2020a). Variational autoencoders and nonlinear ICA: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR.

Khemakhem, I., Monti, R., Kingma, D., and Hyvarinen, A. (2020b). Ice-beem: Identifiable conditional energy-based deep models based on nonlinear ICA. *Advances in Neural Information Processing Systems*, 33:12768–12778.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

Kivva, B., Rajendran, G., Ravikumar, P., and Aragam, B. (2022). Identifiability of deep generative models without auxiliary information. *Advances in Neural Information Processing Systems*, 35:15687–15701.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., et al. (2021). Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR.

Lachapelle, S. and Lacoste-Julien, S. (2022). Partial disentanglement via mechanism sparsity. *arXiv preprint arXiv:2207.07732*.

Lachapelle, S., Mahajan, D., Mitliagkas, I., and Lacoste-Julien, S. (2023). Additive decoders for latent variables identification and cartesian-product extrapolation. *arXiv preprint arXiv:2307.02598*.

Lachapelle, S., Rodriguez, P., Sharma, Y., Everett, K. E., Le Priol, R., Lacoste, A., and Lacoste-Julien, S. (2022). Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ICA.

In *Conference on Causal Learning and Reasoning*, pages 428–484. PMLR.

Leivada, E., Murphy, E., and Marcus, G. (2023). Dall · e 2 fails to reliably capture common syntactic processes. *Social Sciences & Humanities Open*, 8(1):100648.

Liang, W., Kekić, A., von Kügelgen, J., Buchholz, S., Besserve, M., Gresele, L., and Schölkopf, B. (2023). Causal component analysis. *arXiv preprint arXiv:2305.17225*.

Lippe, P., Magliacane, S., Löwe, S., Asano, Y. M., Cohen, T., and Gavves, E. (2022a). icitris: Causal representation learning for instantaneous temporal effects. *arXiv preprint arXiv:2206.06169*.

Lippe, P., Magliacane, S., Löwe, S., Asano, Y. M., Cohen, T., and Gavves, S. (2022b). Citris: Causal identifiability from temporal intervened sequences. In *International Conference on Machine Learning*, pages 13557–13603. PMLR.

Locatello, F., Poole, B., Rätsch, G., Schölkopf, B., Bachem, O., and Tschannen, M. (2020). Weakly-supervised disentanglement without compromises. In *International Conference on Machine Learning*, pages 6348–6359. PMLR.

Mansouri, A., Hartford, J., Ahuja, K., and Bengio, Y. (2022). Object-centric causal representation learning. In *NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations*.

Muandet, K., Balduzzi, D., and Schölkopf, B. (2013). Domain generalization via invariant feature representation. In *International conference on machine learning*, pages 10–18. PMLR.

Peters, J., Bühlmann, P., and Meinshausen, N. (2016). Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(5):947–1012.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. (2021). Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634.

Seigal, A., Squires, C., and Uhler, C. (2022). Linear causal disentanglement via interventions. *arXiv preprint arXiv:2211.16467*.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge University Press.

Varici, B., Acarturk, E., Shanmugam, K., Kumar, A., and Tajer, A. (2023). Score-based causal representation learning with interventions. *arXiv preprint arXiv:2301.08230*.

von Kügelgen, J., Besserve, M., Liang, W., Gresele, L., Kekić, A., Bareinboim, E., Blei, D. M., and Schölkopf, B. (2023). Nonparametric identifiability of causal representations from unknown interventions. *arXiv preprint arXiv:2306.00542*.

Von Kügelgen, J., Sharma, Y., Gresele, L., Brendel, W., Schölkopf, B., Besserve, M., and Locatello, F. (2021). Self-supervised learning with data augmentations provably isolates content from style. *Advances in Neural Information Processing Systems*, 34:16451–16467.

Xie, F., Cai, R., Huang, B., Glymour, C., Hao, Z., and Zhang, K. (2020). Generalized independent noise condition for estimating latent variable causal graphs. *Advances in neural information processing systems*, 33:14891–14902.

Yao, W., Sun, Y., Ho, A., Sun, C., and Zhang, K. (2022). Learning temporally causal latent processes from general temporal data. In *International Conference on Learning Representations*.

Zhang, J., Squires, C., Greenewald, K., Srivastava, A., Shanmugam, K., and Uhler, C. (2023). Identifiability guarantees for causal disentanglement from soft interventions. *arXiv preprint arXiv:2307.06250*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]
3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]

   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Not Applicable]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Appendix

A Theorems and Proofs

Theorem 2 (Single-node imperfect interventions)**.** *Suppose the multi-domain data is gathered from the DGP in equation (4) under Assumptions 1 and 2. Then the autoencoder that solves the reconstruction identity (equation (2)) under Constraints 1 and 2 achieves block-affine identification, i.e., $\forall z \in \mathcal{Z}, \hat{z}_{\hat{\mathcal{S}}} = Dz_{\mathcal{S}} + e$, where $\hat{z}$ is the encoder's output, z is the true latent, $D \in \mathbb{R}^{|\hat{\mathcal{S}}| \times |\mathcal{S}|}$, and $e \in \mathbb{R}^{|\hat{\mathcal{S}}|}$.*

Proof. We begin by first checking that the solution to reconstruction identity under the above-said constraints exists. Set $f = g^{-1}$ and $h = g$ and $\hat{\mathcal{S}} = \mathcal{S}$. Firstly, the reconstruction identity is easily satisfied. Also, the Constraint 2 is satisfied as Assumption 2 holds.

We construct a proof based on the principle of induction. We sort the vertices in $\mathcal{U}$ in the reverse topological order based on the DAG to obtain a list $\mathcal{U}^\star$. We use the principle of induction on this sorted list. Due to Assumption 2, it follows that the first node in the sorted list has to be a terminal node, say this node is j. Consider a component $\hat{z}_i$ of $\hat{z}_{\hat{\mathcal{S}}}$. From affine identification (follows from Theorem 1), we already know that $\hat{z}_i = A_i^\top z + c_i$. Suppose j undergoes an imperfect intervention in domain p. We write the invariance constraint condition equating the distribution of $\hat{z}_i$ between domain 1 and domain p as

$$\begin{aligned}
&\hat{z}_i^{(1)} \overset{d}{=} \hat{z}_i^{(p)}, \\
&A_i^\top z^{(1)} \overset{d}{=} A_i^\top z^{(p)}, \\
&A_i^\top [z_j^{(1)}, z_{-j}^{(1)}] \overset{d}{=} A_i^\top [z_j^{(p)}, z_{-j}^{(p)}].
\end{aligned} \tag{9}$$

Recall $z_j^{(q)} = q_j\big(z_{\text{Pa}(j)}^{(q)}\big) + \varrho_j^{(q)}, \forall q \in [k]$. For all $q \in [k]$, define $w^{(q)} = A_{i,-j}^\top z_{-j}^{(q)} + A_{ij} q_j\big(z_{\text{Pa}(j)}^{(q)}\big)$, where $A_{i,-j}$ is the vector of components in A_i other than $A_{i,j}$ and $z_{-j}^{(q)}$ is the vector of all components of $z^{(q)}$ except $z_j^{(q)}$. Define $v^{(q)} = A_{ij}\varrho_j^{(q)}, \forall q \in [k]$. Substitute these in the above to obtain

$$w^{(1)} + v^{(1)} \overset{d}{=} w^{(p)} + v^{(p)}. \tag{10}$$

We make some important observations now. Observe that $v^{(1)} \perp w^{(1)}$ and $v^{(p)} \perp w^{(p)}$. Also, since the intervention only changes the noise distribution of j and leaves all rest nodes in the graph unaltered $w^{(1)} \overset{d}{=} w^{(p)}$. We now write the moment generating function (MGF) of $w^{(1)} + v^{(1)}$ and equate it to MGF of $w^{(p)} + v^{(p)}$ as follows.

$$M_{w^{(1)}}(t) M_{v^{(1)}}(t) = M_{w^{(p)}}(t) M_{v^{(p)}}(t) \tag{11}$$

Since $w^{(1)} \overset{d}{=} w^{(p)}$, the MGFs are equal. As a result, the MGFs of $v^{(1)}$ and $v^{(p)}$ are equal as well. If the MGFs are equal, then $v^{(1)} \overset{d}{=} v^{(p)}$. If $A_{ij} \neq 0$, then this implies $\varrho^{(1)} \overset{d}{=} \varrho^{(p)}$, which is a contradiction. Therefore, $A_{ij} = 0$. This establishes the base case for the induction.

$$\begin{aligned}
&\hat{z}_i^{(1)} \overset{d}{=} \hat{z}_i^{(s)}, \\
&A_i^\top z^{(1)} \overset{d}{=} A_i^\top z^{(s)}, \\
&A_i^\top [z_j^{(1)}, z_{-j}^{(1)}] \overset{d}{=} A_i^\top [z_j^{(s)}, z_{-j}^{(s)}].
\end{aligned} \tag{12}$$

In domain s, where node j above is intervened, the only nodes that are impacted are j and its descendants. In $w^{(q)} = A_{i,-j}^\top z_{-j}^{(q)} + A_{ij} q_j\big(z_{\text{Pa}(j)}^{(q)}\big)$, the distribution of second term $A_{ij} q_j\big(z_{\text{Pa}(j)}^{(q)}\big)$ is determinded by distribution of parents of j, which are not impacted. The first term $A_{i,-j}^\top z_{-j}^{(q)}$ comprises of both the descendants of j and other non-descendants. Observe that all the descendants of j precede it in the list $\mathcal{U}^\star$. As a result, all the coefficients in

$A_{i,-j}$ corresponding to the descendants of j are zero. Therefore, the distribution of the first term $A_{i,-j}^{\top} z_{-j}^{(s)}$ is same as distribution of $A_{i,-j}^{\top} z_{-j}^{(1)}$. On the whole, the distribution of $w^{(s)}$ is same as distribution of $w^{(1)}$. Also, since the contribution of descendants of j in $w^{(q)}$ is zero, we can conclude that $v^{(q)} \perp w^{(q)}$. We now repeat the same argument as before. We now write the moment generating function (MGF) of $w^{(1)} + v^{(1)}$ and equate it to MGF of $w^{(s)} + v^{(s)}$ as follows.

$$M_{w^{(1)}}(t) M_{v^{(1)}}(t) = M_{w^{(s)}}(t) M_{v^{(s)}}(t) \tag{13}$$

Since $w^{(1)} \overset{d}{=} w^{(s)}$, the MGFs are equal. As a result, the MGFs of $v^{(1)}$ and $v^{(s)}$ are equal as well. If the MGFs are equal, then $v^{(1)} \overset{d}{=} v^{(s)}$. If $A_{ij} \neq 0$, then this implies $\varrho^{(1)} \overset{d}{=} \varrho^{(s)}$, which is a contradiction. Therefore, $A_{ij} = 0$. This completes the proof.

□

Extension of Theorem 2 The DGP considered above has the form $z_j^{(q)} = q_j\big(z_{\mathrm{Pa}(j)}^{(q)}\big) + \varrho_j^{(q)}$. Alternatively, if we consider a new DGP that involves confounder $z_j^{(q)} = q_j\big(z_{\mathrm{Pa}(j)}^{(q)}, u_{\mathrm{Pa}(j)}^{(q)}\big) + \varrho_j^{(q)}$, where $u_{\mathrm{Pa}(j)}^{(q)}$ are confounders that impact at least two latents but are not input to the mixing map g, i.e., $x \leftarrow g(z)$. The exact proof steps can be repeated for this more general data generation process provided the additive noise variable is independent of the parent variables, i.e., $\varrho_j^{(q)} \perp \big(z_{\mathrm{Pa}(j)}^{(q)}, u_{\mathrm{Pa}(j)}^{(q)}\big)$. Observe that we have the following the crucial steps: i) affine identification, ii) $v^{(1)} \perp w^{(1)}$, $v^{(p)} \perp w^{(p)}$ and $w^{(1)} \overset{d}{=} w^{(p)}$, iii) product of MGFs based separation in equation (11), are not impacted by this change and as a result the proof of this extension goes through.

Define $u(\delta) = \frac{\log\big(d/\delta\big)}{\log\big(1/(1-1/2d)\big)}$. We characterize *good interventions* next. If a node s is paired with terminal node w and if the variance of both the intervened nodes increases or decreases in comparison to observational data, then s undergoes a good intervention.

Lemma 1. *Consider the random intervention mechanism described in Assumption 3. If $t \geq u(\delta)$, then with probability $1-\delta$ each node in $\mathcal{U}$ is involved in a good intervention with one of the terminal nodes.*

Proof. Select one of the terminal nodes w. Consider all other nodes in $\mathcal{U} \setminus \{w\}$. The mechanism of interventions described in Assumption 3 goes over the nodes in $\mathcal{U}$ iteratively. In iteration corresponding to interventions for node s, each node in $\mathcal{U} \setminus \{s\}$ is equally likely to be selected for concurrent intervention. Define an event O, which is true if under the intervention the variance of both intervened nodes is increased in comparison to observational data (Domain 1) or if under the intervention variance of intervened is decreased in comparison to observational data. Due to symmetry and non-atomic density p_{σ_ϱ}, the probability of this event is $\frac{1}{2}$. Therefore, the probability p that in iteration for node s it undergoes a good intervention is $p = \frac{1}{2(|\mathcal{U}|-1)}$.

Define an event S such that S occurs if in all of $(|\mathcal{U}|-1)t$ interventions each node in $\mathcal{U} \setminus \{w\}$ undergoes a good intervention, i.e., it is paired with the terminal node w at least once and for each of these interventions event O occurs for the paired nodes. Consider a node $s \in \mathcal{U} \setminus \{w\}$. Define event E_s, where E_s occurs if none of the t interventions conducted in the iteration concerning s are good interventions. This probability evaluates to $P(E_s) = (1-p)^t$. The probability that at least one of E_s is true is bounded above using union bound as follows: $P(\cup_{s \in \mathcal{U}\setminus\{w\}} E_s) \leq (|\mathcal{U}|-1)(1-p)^t$. The probability $P(S) = 1 - P(\cup_{s \in \mathcal{U}\setminus\{w\}} E_s) \geq 1 - (|\mathcal{U}|-1)(1-p)^t$. Observe that if $t \geq u(\delta)$, then $P(S) \geq 1-\delta$.

□

Theorem 3 (Multi-node imperfect interventions)**.** *Suppose the multi-domain data is gathered from the DGP in equation (4) under Assumptions 1 and 3. If the number of multi-node interventions t impacting each node is more than $\frac{\log(d/\delta)}{\log(1/(1-1/2d))}$, then, with probability $1-\delta$, the autoencoder that solves the reconstruction identity (equation (2)) under Constraints 1 and 2 achieves block-affine identification, i.e., $\forall z \in \mathcal{Z}, \hat{z}_{\hat{\mathcal{S}}} = Dz_{\mathcal{S}} + e$, where $\hat{z}$ is the encoder's output, z is the true latent, $D \in \mathbb{R}^{|\hat{\mathcal{S}}| \times |\mathcal{S}|}, e \in \mathbb{R}^{|\hat{\mathcal{S}}|}$.*

Proof. We begin by first checking that the solution to reconstruction identity under the above-said constraints exists. Set $f = g^{-1}$ and $h = g$ and $\hat{\mathcal{S}} = \mathcal{S}$. Firstly, the reconstruction identity is easily satisfied. Also, the Constraint 2 is satisfied as Assumption 3 holds.

We construct a proof based on the principle of induction.

Consider a component $\hat{z}_i$ of $\hat{z}_{\hat{\mathcal{S}}}$. From affine identification (follows from Theorem 1), we already know that $\hat{z}_i = A_i^\top z + c_i$. We sort the vertices in $\mathcal{U}$ in the reverse topological order to obtain a list $\mathcal{U}^\star$. We use the principle of induction on this sorted list. Due to Assumption 3, it follows that the first two nodes in the sorted list have to be a terminal node, which we denote as $\{j, l\}$. Suppose these nodes are intervened in domain p. Observe that since $t \geq u(\delta)$ both of these nodes are intervened with probability $1-\delta$. From the invariance constraint on distribution $\hat{z}_i$ in domain 1 and p it follows

$$\begin{aligned} &\hat{z}_i^{(1)} \overset{d}{=} \hat{z}_i^{(p)}, \\ &A_i^\top z^{(1)} \overset{d}{=} A_i^\top z^{(p)}, \\ &A_i^\top [z_j^{(1)}, z_l^{(1)}, z_{-jl}^{(1)}] \overset{d}{=} A_i^\top [z_j^{(p)}, z_l^{(p)}, z_{-jl}^{(p)}]. \end{aligned} \tag{14}$$

Recall $z_j^{(q)} = q_j\big(z_{\mathrm{Pa}(j)}^{(q)}\big) + \varrho_j^{(q)}, \forall q \in [k]$. For all $q \in [k]$, define $w^{(q)} = A_{i,-jl}^\top z_{-jl}^{(q)} + A_{ij} q_j\big(z_{\mathrm{Pa}(j)}^{(q)}\big) + A_{il} q_l\big(z_{\mathrm{Pa}(l)}^{(q)}\big), \forall q \in [k]$, where $A_{i,-jl}$ is the vector of components in A_i other than $A_{i,j}$ and $A_{i,l}$, and $z_{-jl}^{(q)}$ is the vector of all components of $z^{(q)}$ except $z_j^{(q)}$ and $z_l^{(q)}$. Define $v^{(q)} = A_{ij}\varrho_j^{(q)} + A_{il}\varrho_l^{(q)}$, $\forall q \in [k]$. Substitute these in the above to obtain

$$w^{(1)} + v^{(1)} \overset{d}{=} w^{(p)} + v^{(p)}. \tag{15}$$

We make some important observations now. Observe that $v^{(1)} \perp w^{(1)}$ and $v^{(p)} \perp w^{(p)}$. This is true since $v^{(q)}$ is determined by the noise variables at the terminal nodes. Also, since the intervention only changes the noise distribution of j and l, which are terminal nodes, leaving the rest of the nodes unaltered. Therefore, $w^{(1)} \overset{d}{=} w^{(p)}$. We now write the moment generating function (MGF) of $w^{(1)} + v^{(1)}$ and equate it to MGF of $w^{(p)} + v^{(p)}$ as follows

$$M_{w^{(1)}}(t) M_{v^{(1)}}(t) = M_{w^{(p)}}(t) M_{v^{(p)}}(t). \tag{16}$$

Since $w^{(1)} \overset{d}{=} w^{(p)}$, the MGFs are equal. As a result, the MGFs of $v^{(1)}$ and $v^{(p)}$ are equal as well. If the MGFs are equal, then $v^{(1)} \overset{d}{=} v^{(p)}$. If $A_{ij} \neq 0$ and $A_{il} = 0$, then this implies $\varrho_j^{(1)} \overset{d}{=} \varrho_j^{(s)}$, which is a contradiction. Similarly, $A_{il} \neq 0$ and $A_{ij} = 0$ is not possible either. The last case is $A_{ij} \neq 0$ and $A_{il} \neq 0$. From $v^{(1)} \overset{d}{=} v^{(p)} \implies A_{ij}\varrho_j^{(1)} + A_{il}\varrho_l^{(1)} \overset{d}{=} A_{ij}\varrho_j^{(p)} + A_{il}\varrho_l^{(p)}$. This can only be true if $A_{ij}^2\sigma^2_{\varrho_j^{(1)}} + A_{il}^2\sigma^2_{\varrho_l^{(1)}} = A_{ij}^2\sigma^2_{\varrho_j^{(p)}} + A_{il}^2\sigma^2_{\varrho_l^{(p)}}$. Due to Lemma 1, the selected domain p is such that the two terminal nodes undergo a good intervention and as a result, the variance in LHS is strictly less or strictly greater than the RHS, which makes the equality impossible. Therefore, $A_{ij} = 0$ and $A_{il} = 0$.

This establishes the base case for the induction.

Consider an arbitrary vertex say $s \in \mathcal{U}^\star$. Suppose $A_{ir} = 0$ for all that preceded s in $\mathcal{U}^\star$. Further, suppose that this node s undergoes an imperfect intervention along with terminal node l in domain u. Note here again since $t \geq u(\delta)$, such a domain exists with probability $1-\delta$. From the invariance condition between domain 1 and domain u, it follows

$$\begin{aligned} &\hat{z}_i^{(1)} \overset{d}{=} \hat{z}_i^{(u)}, \\ &A_i^\top z^{(1)} \overset{d}{=} A_i^\top z^{(u)}, \\ &A_i^\top [z_s^{(1)}, z_l^{(1)}, z_{-sl}^{(1)}] \overset{d}{=} A_i^\top [z_s^{(u)}, z_l^{(u)}, z_{-sl}^{(u)}]. \end{aligned} \tag{17}$$

Consider domain u, where node s and l above are intervened simultaneously. Recall $w^{(q)} = A_{i,-sl}^\top z_{-sl}^{(q)} + A_{is} q_s\big(z_{\mathrm{Pa}(s)}^{(q)}\big) + A_{il} q_l\big(z_{\mathrm{Pa}(l)}^{(q)}\big), \forall q \in [k]$. We already showed that $A_{il} = 0$ so the third term is zero. Further, in $A_{i,-sl}$ the terms corresponding to the descendants of s are zero due to supposition in induction principle that

$A_{ir} = 0$ for all that preceded s in $\mathcal{U}^\star$. Hence, no descendant of s contributes to the expression $w^{(q)}$. The term $v^{(q)} = A_{is}\varrho_s^{(q)} + A_{il}\varrho_l^{(q)}$, which again simplifies to $v^{(q)} = A_{is}\varrho_s^{(q)}$. Since $w^{(q)}$ does not involve s or its descendants, we can conclude that $w^{(q)} \perp v^{(q)}, \forall q \in [k]$ and $w^{(u)} \overset{d}{=} w^{(1)}$.

The above expressions in equation (17) can be stated as

$$w^{(u)} + v^{(u)} \overset{d}{=} w^{(1)} + v^{(1)} \tag{18}$$

Since $w^{(q)} \perp v^{(q)}$ and $w^{(u)} \overset{d}{=} w^{(1)}$, it follows that $v^{(u)} \overset{d}{=} v^{(1)}$. If $A_{is} \neq 0$, then this implies $\varrho_s^{(u)} \overset{d}{=} \varrho_s^{(1)}$, which leads to a contradiction. Hence, $A_{is} = 0$. This completes the proof.

□

Extension of Theorem 3 In Theorem 3, we considered two-node interventions. Let us ask what happens if m-interventions occur at the same time. If we extend the Assumption 2 to require m terminal nodes, the rest of the argument extends to this case too. Firstly, in Lemma 1 we showed that if the minimum number of interventions t that each node is involved is sufficiently large, then all the nodes end up being paired with one of the terminal nodes. The extension of Lemma 1 reads: if the minimum number of interventions t that each node is involved is sufficiently large, then all the nodes end up being paired with $m-1$ terminal nodes under a good intervention. In the proof of Theorem 3, in the base case, we showed that the A_{ij} and A_{il} are zero where $\{j, l\}$ are two terminal nodes involved in the intervention. In the extension, we consider the domain in which m terminal nodes are involved in the intervention and the coefficient A_{ir} is zero for all r corresponding to indices of the terminal nodes intervened in that domain. The rest of the argument from the principle of induction is identical.

Theorem 4. *Suppose the multi-domain data is generated from equation 1 and satisfies Assumptions 1, 4, 5. Then the autoencoder that solves the reconstruction identity in equation 2 under Constraints 1 and 3 achieves the following identification guarantees: Each latent component $i \in \mathcal{S}$ satisfies $\hat{z}_i = A_i^\top z + c_i$, where, among all the vectors $A_i \succcurlyeq 0$, the ones that are feasible under the assumptions and constraints in this theorem must satisfy $A_{ir} = 0$ for all $r \in \mathcal{U}$.*

Proof. We begin by first checking that the solution to reconstruction identity under the above-said constraints exists. Set $f = g^{-1}$ and $h = g$ and $\hat{\mathcal{S}} = \mathcal{S}$. Firstly, the reconstruction identity is easily satisfied. Also, the Constraint 3 is satisfied as Assumption 4 holds.

From the Assumptions 1 and Constraint 1 we know that $\hat{z} = Az + c$ (follows from Theorem 1). Let us consider $i \in \hat{\mathcal{S}}$. We know that $\hat{z}_i = A_i^\top z + c_i$. Suppose $A_i \succcurlyeq 0$, where each component of A_i is non-negative.

Let us consider the domains p, q, from Assumption 5. We compute the maximum value of $\hat{z}_i$ in domain p and q below.

$$z^{\max,p} = \arg\max_{z \in \mathcal{Z}^{(p)}} A_i^\top z + c_i \tag{19}$$

$$z^{\max,q} = \arg\max_{z \in \mathcal{Z}^{(q)}} A_i^\top z + c_i \tag{20}$$

From Constraint 3, $A_i^\top z^{\max,p} = A_i^\top z^{\max,q}$. Suppose $A_{ik} > 0$ for some $k \in \mathcal{U}$. From Assumption 5, it follows that there exists a $z \in \mathcal{Z}^{(q)}$ such that $z \succcurlyeq z^{\max,p}$ and $z_j > z_j^{\max,q}$ for all $j \in \mathcal{U}$. Therefore, $A_i^\top z > A_i^\top z^{\max,p}$. This contradicts $A_i^\top z^{\max,p} = A_i^\top z^{\max,q}$. Therefore, $A_{ik} = 0$. □

Remark on Definition 1 We illustrate the type of functions captured by Definition 1 in Figure 4. In Figure 4, we show that a function γ_θ has three minima (shown as stars) over $[0,1] \times [0,1]$. We illustrate two domains in panels a) and b). For Domain 1 in panel a), the minima over Domain 1 coincides with minima over $[0,1] \times [0,1]$ but for Domain 2 that is not the case. The figure lays down the examples idea behind the proof we are about to present next. Under sufficiently many diverse interventions, it can be guaranteed that we obtain one domain that is similar to Domain 1 (capturing the minima over $[0,1] \times [0,1]$) in Figure 4 and another domain that is similar to Domain 2 (not capturing the minima over $[0,1] \times [0,1]$) in Figure 4.

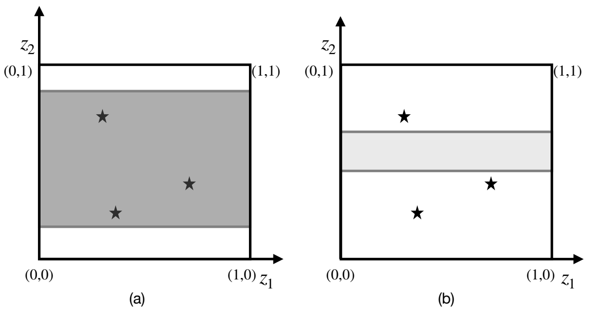

Figure 4: The minima of a candidate function γ_θ over $[0,1]\times[0,1]$ is attained at points shown in stars. Support of Domain 1 and Domain 2 are shown in light and dark grey. The minimum value of γ_θ over Domain 1 is not the same as the minimum over Domain 2. Therefore, $a_1(\cdot)$ relating the first component of the autoencoder, which satisfies support invariance constraint, to the true latent cannot be equal to the candidate function γ_θ.

Theorem 5. *If we gather data generated from equation (1), where the support of z_2 for each domain is sampled i.i.d. from Assumption 6 and support of z_1 is fixed to $[0,1]$. Further, suppose the number of domains satisfies $k \geq N(\delta,\varepsilon,\eta,\iota)$. Then the set of maps $a_1(\cdot)$ that relate $\hat{z}_1$ to $[z_1,z_2]$ does not contain any function from Γ and thus achieves Γ^c identification, where $\hat{z}$ is obtained by solving the reconstruction identity (equation 2) under support invariance constraint (Constraint 3) on $\hat{z}_1$.*

Proof. Consider the set Θ of parameters, which characterize all the functions in Γ. Let us construct a ρ-cover for the set Θ with $\rho = \frac{\eta}{4L}$, where η and L are constants from Definition 1. We define the set of functions in the cover as $\Gamma_c = \{\gamma_1,\cdots,\gamma_{N_c}\}$, where N_c is the size of the cover and $N_c = \left(\frac{2\max_{\theta\in\Theta}\|\theta\|\sqrt{s}}{\rho}\right)^s$ (follows from (Shalev-Shwartz and Ben-David, 2014)).

Consider a $\gamma_j \in \Gamma_c$ with parameters θ_j. From Definition 1, there exists an interval $[\alpha^\dagger,\beta^\dagger]$ with width at least ι such that the minimum value in $[0,1]\times[\alpha^\dagger,\beta^\dagger]$ is at least η more than the minimum value over the entire set $[0,1]\times[0,1]$. Since the support to z_2 is sampled randomly, we compute the probability that one of the sampled domain's support is contained in $[\alpha^\dagger,\beta^\dagger]$. The probability of first success (where success is the event that support of z_2 is a subset of $[\alpha^\dagger,\beta^\dagger]$) in one of the t trials is $1-(1-p_s)^t$. We want

$$1-(1-p_s)^t \geq 1-\frac{\delta}{2} \implies \frac{\delta}{2} \geq (1-p_s)^t \implies \log\left(\frac{2}{\delta}\right)/\log(1/(1-p_s)) \leq t$$

We plug $p_s = c_1\iota^l$ following Assumption 6. If we set $t \geq t^1_{\text{min}} = \log(\frac{2}{\delta})/\log(1/(1-c_1\iota^l))$, then with probability $1-\delta/2$ at least for one of the domains indexed from 1 to t^1_{min} the minimum value of γ_j in $[0,1]\times[\alpha^\dagger,\beta^\dagger]$ is η larger than the minimum value in $[0,1]\times[0,1]$.

Next, we show that if the number of domains is sufficiently large, then the probability that one of the domains support contains $[\varepsilon,1-\varepsilon]$ is sufficiently high. The probability of first success (where success is the event that the intervention support contains $[\varepsilon,1-\varepsilon]$). In this case, we follow the same calculations as above. It follows that if $t \geq t^2_{\text{min}} = \log(\frac{2}{\delta})/\log(1/(1-c_2\varepsilon^r))$, then with probability $1-\delta/2$ the support of z_2 in at least one of the domains indexed from $t^1_{\text{min}}+1$ to $t^1_{\text{min}}+t^2_{\text{min}}$ contains $[\varepsilon,1-\varepsilon]$ the global minimum of γ_j with probability at least $1-\delta/2$. Hence, we can conclude that with probability $1-\delta$ both the success events described above happen. In the case of this event, the function γ_j cannot satisfy the support invariance constraint.

Let us consider all the elements in Γ_c together now. We now derive a bound on the number of domains such that none of the elements in Γ_c satisfy the support invariance constraint. We divide the total k domains into blocks of equal length. The first block is chosen to be sufficiently large to ensure that with probability $1-\frac{\delta}{N_c}$, the first element of Γ_c, i.e., γ_1 does not satisfy support invariance constraints. Similarly, the second block is chosen to be sufficiently large such that γ_2 cannot satisfy support invariance constraints and so on. The minimum size of each block is computed by substituting δ with δ/N_c in the expression for $t^1_{\min} + t^2_{\min}$ derived above. The final expression for $N(\delta, \varepsilon, \eta, \iota)$ is given as

$$N_c\Bigg(\log\bigg(\frac{2N_c}{\delta}\bigg)/\log\bigg(1/(1-c_1\iota^l)\bigg) + \log\bigg(\frac{2N_c}{\delta}\bigg)/\log\bigg(1/(1-c_2\varepsilon^r)\bigg)\Bigg)$$

where $N_c = \Bigg(\frac{2\max_{\theta\in\Theta}\|\theta\|\sqrt{s}}{\rho}\Bigg)^s$ and $\rho = \frac{\eta}{4L}$.

Observe that since the probability of success is bounded below by $1 - \frac{\delta}{N_c}$, the overall probability is bounded by at least $1-\delta$. So far, we have shown that none of the elements in the cover of Θ, i.e., Γ_c satisfy support invariance constraints.

Let us now consider a $\gamma_\theta \in \Gamma$. The nearest neighbor of this γ_θ in the cover is say γ_j. Suppose the parameter associated with γ_j is θ_j. Therefore, $\gamma_j = \gamma_{\theta_j}$. Since θ_j is an element of $\rho-$cover, the separation between their corresponding parameters is $\|\theta_j - \theta\| \leq \rho$. Since the number of domains is larger than $N(\delta, \varepsilon, \eta, \iota)$ we can state the following. With probability $1 - \delta/N_c$, there exists a pair of domains whose supports say $\mathcal{Z}$ and $\tilde{\mathcal{Z}}$, where γ_{θ_j}'s minimum value on the former is at least η higher than the minimum value on $\tilde{\mathcal{Z}}$. Let us now compute a lower bound on the minimum value of γ on $\mathcal{Z}$. For all $z \in \mathcal{Z}$

$$|\gamma_\theta(z) - \gamma_{\theta_j}(z)| \leq L\|\theta - \theta_j\| \leq L\rho \implies \gamma_\theta(z) \geq \gamma_{\theta_j}(z) - L\rho$$

In the first inequality, we rely on Lipschitz continuity of γ_θ in θ (from Assumption 6). From the above, it follows that

$$\min_{z\in\mathcal{Z}} \gamma_\theta(z) \geq \min_{z\in\mathcal{Z}} \gamma_{\theta_j}(z) - L\rho \tag{21}$$

Next, we compute an upper bound on the minimum value of γ_θ on $\tilde{\mathcal{Z}}$

$$|\gamma_\theta(z) - \gamma_{\theta_j}(z)| \leq L\|\theta - \theta_j\| \leq L\rho \implies \gamma_\theta(z) \leq \gamma_{\theta_j}(z) + L\rho$$

From the above, it follows that

$$\min_{z\in\tilde{\mathcal{Z}}} \gamma_\theta(z) \leq \min_{z\in\tilde{\mathcal{Z}}} \gamma_{\theta_j}(z) + L\rho \tag{22}$$

We now take the difference of the bounds in equation (21) and (22) above to arrive at the following.

$$\min_{z\in\tilde{\mathcal{Z}}} \gamma_\theta(z) - \min_{z\in\mathcal{Z}} \gamma_\theta(z) \geq \min_{z\in\mathcal{Z}} \gamma_{\theta_j}(z) - \min_{z\in\tilde{\mathcal{Z}}} \gamma_{\theta_j}(z) - 2L\rho \geq \eta - 2L\rho = \frac{\eta}{2}$$

where we set $\rho = \eta/4L$ in the last inequality. Therefore, γ_θ does not satisfy support invariance. We require the above argument to hold for all $\gamma_\theta \in \Gamma$. Here we exploit the fact that with probability $1-\delta$ all elements in the cover Γ_c do not satisfy the support invariance constraint. Therefore, we can pick any $\gamma_\theta \in \Gamma$, select the corresponding nearest neighbor in the cover, and apply the argument stated above. This completes the proof.

□

A.1 Beyond the Two Variable Case

In this section, we aim to generalize the results presented in the previous section to more than two variables. We first adapt the Definition 1.

Definition 2. *Fix some constants $\eta > 0$, $\varepsilon > 0$ and $\iota > 0$. Given these constants, we define a set of functions Γ as follows. Each function $\gamma_\theta : [0,1]^d \rightarrow \mathbb{R}$ in Γ i) is parameterized by $\theta \in \Theta$, where Θ is a bounded subset of*

$\mathbb{R}^s$,ii) the minima of γ_θ over the entire set $[0,1]^d$ lie in the ε interior of the set, i.e., in $[\varepsilon, 1-\varepsilon]^d$, and iii) there exists a hypercube $\mathcal{L}$ of volume at least ι such that

$$\left|\min_{z\in[0,1]^d}\gamma(z) - \min_{z\in[0,1]\times\mathcal{L}}\gamma(z)\right| \geq \eta.$$

For each $z \in [0,1]^d$, γ_θ is Lipschitz continuous in the parameter $\theta \in \Theta$ with Lipschitz constant L.

Next, we adapt Assumption 6.

Assumption 7. *We assume that the domains are drawn at random and the support of latents in $\mathcal{U}$ satisfy $\mathbb{P}\left(\mathcal{Z}_{\mathcal{U}}^{(p)} \subseteq [\alpha_1, \beta_1] \times \cdots [\alpha_{|\mathcal{U}|}, \beta_{|\mathcal{U}|}]\right) \geq c_1 \mathrm{vol}^l[[\alpha_1,\beta_1]\times\cdots[\alpha_{|\mathcal{U}|},\beta_{|\mathcal{U}|}]]$ and $\mathbb{P}\left(\mathcal{Z}_{\mathcal{U}}^{(p)} \supseteq [\kappa, 1-\kappa]^q\right) \geq c_2\kappa^{qr}$, where l and r are some integers and c_1, c_2 some constants.*

Define

$$\tilde{N}(\delta,\varepsilon,\eta,\iota) = N_c\left(\log\left(\frac{2N_c}{\delta}\right)/\log\left(1/(1-c_1\iota^l)\right) + \log\left(\frac{2N_c}{\delta}\right)/\log\left(1/(1-c_2\varepsilon^{dr})\right)\right)$$

where $N_c = \left(\frac{2\max_{\theta\in\Theta}\|\theta\|\sqrt{s}}{\rho}\right)^s$ and $\rho = \frac{\eta}{4L}$

Theorem 6. *If we gather data generated from equation (1), where the support of z_2 for each domain is sampled i.i.d. from Assumption 7. Further, if the number of domains $k \geq \tilde{N}(\delta,\varepsilon,\eta,\iota)$, then the maps $a_1(\cdot)$, which are obtained from autoencoders that solve the reconstruction identity in equation (2) under support invariance constraint (Constraint 3) on $\hat{z}_1$, do not contain function from Γ.*

Proof. We will follow the same line of reasoning as in the proof of the two-variable case. Consider the set Θ characterizing the functions γ. Let us construct a ρ-cover for the set Θ. The cover consists of functions in the set $\Gamma_c = \{\gamma_1, \cdots, \gamma_{N_c}\}$, where N_c is the size of the cover. Consider $\gamma_j \in \Gamma_c$ with parameters θ_j. From Assumption 7, there exists a hypercube $\mathcal{L}$ with volume at least ι such that the minimum value in that hypercube is η more than the global minimum on the set $[0,1]^d$. The probability that one of the domain's support is contained in the hypercube $\mathcal{L}$ is calculated as follows. The probability of first success (where success is the event that intervention support is a subset of $\mathcal{L}$) in one of the t trials is $1-(1-p_s)^t$. We want

$$1-(1-p_s)^t \geq 1-\frac{\delta}{2} \implies \frac{\delta}{2} \geq (1-p_s)^t \implies \log\left(\frac{2}{\delta}\right)/\log(1/(1-p_s)) \leq t$$

Finally we have $t \geq t^1_{\text{min}} = \log\left(\frac{2}{\delta}\right)/\log(1/(1-c_1\iota^l))$. Therefore, with probability $1-\delta/2$ at least one of the domains s indexed from 1 to t^1_{min} achieves a minima η larger than the global minimum of γ_j.

Next, we derive the probability that one of the domain's support contains $[\varepsilon, 1-\varepsilon]^d$. The probability of first success (where success is the event that the domain contains $[\varepsilon, 1-\varepsilon]^d$). In this case, we have $t \geq t^2_{\text{min}} = \log(\frac{2}{\delta})/\log(1/(1-c_2\varepsilon^{rd}))$. Therefore, with probability $1-\delta/2$ at least one of the domains indexed from $t^1_{\text{min}}+1$ to $t^1_{\text{min}}+t^2_{\text{min}}$ achieves the global minimum of γ_j with probability at least $1-\delta/2$. Hence, we can conclude that with probability $1-\delta$ both the success events described above happen. In the case of this event, the function γ_j cannot satisfy the invariance constraint.

Let us consider all the elements in Γ_c together now. We would require the total interventions to be divided into blocks of equal length. The first block is chosen to be sufficiently large to ensure that with probability $1-\frac{\delta}{N_c}$, γ_1 cannot satisfy support invariance constraints. Similarly, the second block is chosen to be sufficiently large such that γ_2 cannot satisfy support invariance constraints and so on. Due to symmetry, the minimum size of each block is computed by substituting δ with δ/N_c in the expression for $t^1_{\text{min}}+t^2_{\text{min}}$ derived above. The final expression for $\tilde{N}(\delta,\varepsilon,\eta,\iota)$ is given as

$$N_c\left(\log\left(\frac{2N_c}{\delta}\right)/\log\left(1/(1-c_1\iota^l)\right) + \log\left(\frac{2N_c}{\delta}\right)/\log\left(1/(1-c_2\varepsilon^{dr})\right)\right)$$

where $N_c = \left(\frac{2\max_{\theta\in\Theta}\|\theta\|\sqrt{s}}{\rho}\right)^s$ and $\rho = \frac{\eta}{4L}$. Observe that since the probability of success is bounded below by $1-\frac{\delta}{N_c}$, the overall probability is bounded by at least $1-\delta$. So far we have shown that none of the elements in the cover of Θ, i.e., Γ_c satisfy support invariance constraints.

Let us now consider a $\gamma \in \Theta$. The nearest neighbor of this γ in the cover is say γ_j. Suppose the parameter of γ_j is θ_j. Therefore, $\gamma_j = \gamma_{\theta_j}$. The separation between their corresponding parameters is $\|\theta_j - \theta\| \leq \rho$. We know that with probability $1-\delta$, γ_i does not satisfy the support invariance constraint. There exist interventional distributions whose supports say $\mathcal{Z}$ and $\tilde{\mathcal{Z}}$, where γ_j's minimum value on the former is at least η higher than the minimum value on $\tilde{\mathcal{Z}}$. Let us now compute a lower bound on the minimum value of γ on $\mathcal{Z}$.

$$|\gamma_\theta(z) - \gamma_{\theta_j}(z)| \leq L\|\theta - \theta_j\| \leq L\rho \implies \gamma_\theta(z) \geq \gamma_{\theta_j}(z) - L\rho$$

From the above, it follows that

$$\min_{z\in\mathcal{Z}} \gamma_\theta(z) \geq \min_{z\in\mathcal{Z}} \gamma_{\theta_j}(z) - L\rho$$

Next, we compute an upper bound on the minimum value of γ on $\tilde{\mathcal{Z}}$

$$|\gamma_\theta(z) - \gamma_{\theta_j}(z)| \leq L\|\theta - \theta_j\| \leq L\rho \implies \gamma_\theta(z) \leq \gamma_{\theta_j}(z) + L\rho$$

From the above, it follows that

$$\min_{z\in\tilde{\mathcal{Z}}} \gamma_\theta(z) \leq \min_{z\in\tilde{\mathcal{Z}}} \gamma_{\theta_j}(z) + L\rho$$

We now take the difference of the bounds above to arrive at the following.

$$\min_{z\in\tilde{\mathcal{Z}}} \gamma_\theta(z) - \min_{z\in\mathcal{Z}} \gamma_\theta(z) \geq \min_{z\in\mathcal{Z}} \gamma_{\theta_j}(z) - \min_{z\in\tilde{\mathcal{Z}}} \gamma_{\theta_j}(z) - 2L\rho \geq \eta - 2L\rho = \frac{\eta}{2}$$

where we set $\rho = \eta/4L$ in the last inequality. Therefore, γ does not satisfy support invariance. Note that the above argument is general and applies to every $\gamma \in \Theta$ since we can pick the corresponding nearest neighbor in the cover.

This completes the proof. □

A.2 Polytope Support

In this section, we assume that the support of latents in each domain is characterized by bounded polytopes – the convex hull of a finite number of vertices, where each vertex has a bounded norm. Under the assumptions and the constraint (Assumption 1 and Constraint 1) we know that $\hat{z}$ is an affine function of z. If the support of z is a bounded polytope, then evaluating the maximum and minimum value that each component of $\hat{z}$ depends only on the vertices of the polytope following the fundamental theorem of linear programming. This allows us to provide identification guarantees by placing assumptions on the diversity of these polytopes, i.e., on these vertices, observed across domains. .

Following Constraint 3, we equate the maximum value of components in $\hat{\mathcal{S}}$ across domains. Suppose we equate the maximum of $\hat{z}_i$ across domain p and q. We obtain $A_i^\top(z^{\max,p} - z^{\max,q}) = 0$, where $z^{\max,p}$, $z^{\max,q}$ correspond to the vertex of the support polytope in domain p, q respectively. Observe how the expression depends on the difference of vertices from different polytopes. We define a set of matrices $\mathcal{M}$ formed by taking the difference of vertices from the different polytopes as follows. Firstly, we fix the first domain as the reference domain and we define difference vectors with respect to the vertices in this domain. We also fix some arbitrary ordering of vertices in the polytope; say they are in the increasing order of the first coordinate. We start with the first vertex in the first domain. Next, pick the second domain and pick its first vertex. Take the difference of the two selected vectors, this difference vector forms the first row of one of the matrices. Pick the third domain, take its first vertex, and again take the difference to get the second row of the matrix. Repeat this process for all the domains. As a result, we get a matrix with $k-1$ rows and d columns. To summarize, the set of matrices $\mathcal{M}$ consists of $k-1 \times d$ matrices that satisfy the following condition. For each matrix $M \in \mathcal{M}$, the r^{th} row of the matrix is defined as the difference between some vertex from $(r+1)^{th}$ domain and another vertex from the first domain.

Assumption 8.

- *The support of latents in each domain $p \in [k]$, i.e., $\mathcal{Z}^{(p)}$ is a bounded polytope; the number of domains $k \geq d+1$. Each matrix $M \in \mathcal{M}$ has a rank equal to the number of non-zero columns.*
- *For each component $j \in \mathcal{U}$, there exists a domain $p \in [k]$ such that the following condition holds. We denote the value assumed by z_j in $\mathcal{Z}^{(1)}$ on vertex r as v^r. We assume that there exists another domain p with support $\mathcal{Z}^{(p)}$ such that z_j does not take the same value as v^r at any vertex of $\mathcal{Z}^{(p)}$.*

The first part of the above assumption states a simple regularity condition on matrix M. The second part of the above assumption is also a simple regularity condition on components in $\mathcal{U}$. The condition only requires that the value attained at some vertex is not attained at any other vertex for some other domain.

Further remarks on Assumption 8. Next, we illustrate that the Assumption 8 holds rather easily in many settings. Consider a setting where $z = [z_1, z_2]$ and both z_1, z_2 take values between 0 and 1. We consider the setting where the support of z forms a polytope. For each domain p, the polytope is sampled as follows. Each polytope consists of M vertices and we sample M values for z_2 uniformly at random $[0,1]$. For z_1, we sample $M-2$ vertices uniformly at random from the interval $[0,1]$. For the remaining 2 vertices, we fix z_1 to take value 0 on one of them and 1 on the other. We generate k polytopes following the above process and check if the rank constraint in the first part of Assumption 8 is satisfied. We repeat this process over ten thousand trials and find that the assumption always holds for different values of M and k. The second part of the assumption holds trivially in the above case as two uniform random variables sampled independently from $[0,1]$ are not equal to probability one.

In what follows, we use the notation $a_{\mathcal{B}}$ to denote a vector formed by components of a whose indices in a are from the set $\mathcal{B}$.

Theorem 7. *Suppose the data is generated from different domains following equation (1) such that Assumptions 1, 4, 8 are satisfied. The autoencoder that solves the reconstruction identity in equation (2) under Constraint 1, 3 satisfies*

$$\hat{z}_{\hat{\mathcal{S}}} = D z_{\mathcal{S}} + e$$

where $D \in \mathbb{R}^{|\hat{\mathcal{S}}| \times |\mathcal{S}|}$, $e \in \mathbb{R}^{|\hat{\mathcal{S}}|}$.

Proof. We begin by first checking that the solution to reconstruction identity under the above-said constraints exists. Set $f = g^{-1}$ and $h = g$ and $\hat{\mathcal{S}} = \mathcal{S}$. The reconstruction identity and Constraint 3 is satisfied as Assumption 4 holds.

Consider a component $m \in \hat{\mathcal{S}}$. From Constraint 3, we know that the support of $\hat{z}_m$ does not change. From Theorem 1, we also know that there is an affine relationship between $\hat{z}$ and z. Therefore, we can write

$$\hat{z}_m = A_m^\top z + c_m \tag{23}$$

The support of $\hat{z}_m$ is determined by the maximum and minimum of $A_m^\top z + c_m$ computed on the respective domains. Let us compute the maximum and minimum of $\hat{z}_m$ in domain p as follows.

$$z^{\max}(A_m, p) = \arg\max_{z \in \mathcal{Z}^{(p)}} A_m^\top z + c_m \tag{24}$$

$$z^{\min}(A_m, p) = \arg\min_{z \in \mathcal{Z}^{(p)}} A_m^\top z + c_m \tag{25}$$

We define a vector $A_m^{\mathcal{U}}$ that contains components of A_m whose indices in A_m form the set $\mathcal{U}$. We now show that support invariance constraints in Constraint 3 implies that $A_m^{\mathcal{U}} = 0$. Suppose $A_m^{\mathcal{U}} \neq 0$ (at least one element of this vector is non-zero). In this case, we write the maximum value of the objective as

$$\sum_l A_{ml} z^{\max}(A_m, p)$$

Due to the support invariance constraint we get

$$\sum_l A_{ml} z_l^{\max}(A_m, p) = \sum_l A_{ml} z_l^{\max}(A_m, 1)$$
$$\sum_l A_{ml}\big(z_l^{\max}(A_m, p) - z_l^{\max}(A_m, 1)\big) = 0$$
$$z_{\text{diff}}^{\top}(A_m, p) A_m = 0$$

$z_{\text{diff}}^{\top}(A_m, p)$ is the difference vector formed by taking the difference $z_l^{\max}(A_m, p) - z_l^{\max}(A_m, 1)$. Construct a matrix $Z_{\text{diff}}(A_m) \in \mathbb{R}^{k-1\times l}$ by stacking the difference vectors $z_{\text{diff}}^{\top}(A_m, p)$ for all p in $\{2, \cdots, k\}$.

Let us consider the largest submatrix of $Z_{\text{diff}}(A_m)$ with no zero columns and denote it as $Z_{\text{diff}}^s(A_m)$. Following Assumption 8, $Z_{\text{diff}}^s(A_m)$ has a full column rank. Therefore, a non-trivial solution to $Z_{\text{diff}}^s(A_m)v = 0$ does not exist and thus $v = 0$. Consider an element $j \in \mathcal{U}$. Due to Assumption 8, the column in $Z_{\text{diff}}(A_m)$ corresponding to j is non-zero. Therefore, for each element $j \in \mathcal{U}$ the corresponding columns in $Z_{\text{diff}}(A_m)$ are non-zero. The columns of $Z_{\text{diff}}^s(A_m)$ contain all coefficients in $\mathcal{U}$, which implies $A_m^{\mathcal{U}} = 0$. This completes the proof.

□

B Experiments

In this section, we provide additional experimental results and other additional details for the experiments. The experiments were carried out using the internal cluster of Mila, Quebec AI Institute. For the first stage training of image datasets (Balls dataset, and unlabeled coloured MNIST) we use NVIDIA A100 GPUs. For the second stage of all datasets, as well as the first stage of polynomial mixing dataset, and both stages of the linear mixing dataset, we only train our model on the CPU as these are rather inexpensive runs.

B.1 Linear Mixing

B.1.1 Data Generation and Model Architecture

For both linear and polynomial $g(\cdot)$ we sample $z = (z_{\mathcal{S}}, z_{\mathcal{U}})^{\top}$ as follows ($z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$). We sample $z_{\mathcal{S}} \sim$ Uniform$[0, 1]$ across all domains. For domain $i \in [k]$ we sample $z_{\mathcal{U}} \sim$ Uniform$[l^i, h^i]$, where $l^i, h^i \sim$ Uniform$[-5, 5]$. Then for the Independent SCM setting, we obtain the observational data via $x = Az$, where $A \in R^{n\times d}$ is a full-rank random matrix whose entries are drawn from Uniform$[0, 1]$. We obtain a Dynamic SCM by altering the above z as follows. For each sample, $z_{\mathcal{U}}^j$ will be offset by the $z_{\mathcal{S}}^j$ with probability p and remain unchanged otherwise. In our experiments we set p to 0.5. We generate 10000 samples for the training split and 2000 for the validation split. The test and validation sets are the same, since we do not search over hyperparameters (see Section B.5).
For Linear mixing $g(\cdot)$, stages 1, 2 are carried out simultaneously by a linear autoencoder that is jointly optimized with reconstruction objective and invariance penalty.

B.1.2 Results

In Table 8, 9 we provide additional results when z's follow an independent and dynamic SCM as described in the main body respectively. We conducted experiments for different values of the dimension of the underlying latents d and for a varying number of domains k. We divide the table into three sections with top block corresponding to Min-Max penalty, the middle block corresponding to the MMD penalty and the bottom block corresponding to the combination of the two denoted Min-Max + MMD. Across the different settings, we observe that as the number of domains increase we achieve high $R^2_{\mathcal{S}}$ and low $R^2_{\mathcal{U}}$.

B.2 Polynomial Mixing

B.2.1 Data Generation and Model Architecture

The latents z are sampled identical to the procedure for linear mixing dataset and the details of the polynomial mixing function $g(\cdot)$ are found in Assumption 1. For stage 1, we use a polynomial autoencoder as follows. The

Table 8: Linear Mixing Dataset R^2 scores, **Independent SCM** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{2d}$. The three sections from top to bottom correspond to Min-Max, MMD, and the combination.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
8	0.45±0.02	0.99±0.00	0.93±0.04	0.98±0.01	0.30±0.03	0.01±0.00	0.08±0.04	0.03±0.01
16	0.40±0.01	0.81±0.02	0.95±0.02	0.91±0.03	0.30±0.03	0.10±0.02	0.03±0.01	0.11±0.03
32	0.35±0.00	0.51±0.03	0.80±0.01	0.88±0.01	0.45±0.05	0.24±0.02	0.13±0.02	0.12±0.02
64	0.32±0.00	0.35±0.00	0.63±0.01	0.80±0.00	0.52±0.04	0.35±0.01	0.19±0.00	0.15±0.00
8	0.64±0.05	0.94±0.03	1.00±0.00	0.99±0.00	0.19±0.01	0.03±0.01	0.01±0.00	0.01±0.00
16	0.51±0.04	0.77±0.01	0.92±0.01	0.97±0.01	0.09±0.04	0.10±0.02	0.09±0.04	0.06±0.01
32	0.38±0.02	0.60±0.01	0.75±0.03	0.90±0.01	0.34±0.04	0.17±0.00	0.12±0.02	0.07±0.01
64	0.30±0.01	0.34±0.01	0.57±0.02	0.75±0.00	0.44±0.02	0.27±0.01	0.19±0.01	0.12±0.00
8	0.63±0.06	0.99±0.00	0.99±0.00	0.99±0.00	0.19±0.01	0.01±0.00	0.02±0.00	0.02±0.00
16	0.53±0.02	0.82±0.01	0.93±0.02	0.97±0.00	0.24±0.04	0.10±0.02	0.06±0.02	0.04±0.00
32	0.39±0.02	0.64±0.04	0.81±0.01	0.91±0.01	0.38±0.06	0.17±0.03	0.11±0.00	0.08±0.01
64	0.33±0.00	0.39±0.00	0.64±0.01	0.80±0.00	0.44±0.03	0.30±0.02	0.16±0.01	0.12±0.01

encoder architecture is given in Table 12 where n, d denote the dimensions of x, z, respectively. For decoding the outputs of the above encoder $\hat{z}$, we use a polynomial decoder which takes $\hat{z}$ and follows the procedure explained in Assumption 1, where the coefficient matrix G is to be learned and is parameterized by a linear layer.

B.2.2 Results

In Tables 13, 14, 15 and 16, 17, 18 we provide additional results when z's follow an independent and dynamic SCM via a polynomial mixing $g(\cdot)$. We conducted experiments for different values of the dimension of the underlying latents d, different polynomial degrees, and for a varying number of domains k. We divide each table into two sections with top block corresponding to degree two polynomials with varying d, and the bottom block corresponding to the degree three polynomials. For each dimension we present two rows, the top row corresponding to the R^2 scores after training an autoencoder with reconstruction objective only, and right before enforcing any distributional invariances. Since we only need an autoencoder that can fully reconstruct the input, there is no need for training multiple perfect autoencoders, hence there is no standard error reported for such entries. We then take the perfectly trained autoencoder and enforce the distributional invariance penalty with 5 seeds, and present the results in the bottom row per each dimension. Across the different settings, we observe that as the number of domains increase we achieve high $R^2_{\mathcal{S}}$ and low $R^2_{\mathcal{U}}$.

B.3 Balls Dataset

B.3.1 Data Generation and Model Architecture

In Tables 23, 24 we provide additional results when z's (i.e., balls' coordinates) follow an independent and dynamic SCM. As described in the main body, we observe that as the number of domains increase we achieve high $R^2_{\mathcal{S}}$ and low $R^2_{\mathcal{U}}$ and especially under the combination of Min-Max and MMD penalty. When z's follow an independent SCM, $z_{\mathcal{S}}$, the invariant block of z corresponds to the coordinates of the ball that is always sampled in an $m \times n$ rectangle that is at a fix location across all domains. The other ball that accounts for $z_{\mathcal{U}}$ is sampled from an $m' \times n'$ rectangle whose location varies across the k domains. When z's follow a dynamic SCM, we alter each component of $z_{\mathcal{U}}$ with probability 0.5 by adding or subtracting its counterpart in $z_{\mathcal{S}}$, subject to the constraints that $z_{\mathcal{U}}$ remains inside the frame, and that the two balls do not overlap to violate the injectivity assumption. The training and validation splits comprise 50000 and 10000 samples, respectively. We conducted experiments for a varying number of domains k. We divide the table into three sections with top block corresponding to Min-Max penalty, the middle block corresponding to the MMD penalty and the bottom block corresponding to the combination of the two denoted Min-Max + MMD. For each penalty we present two rows, the top row corresponding to the R^2 scores after training an autoencoder with reconstruction objective only, and right before enforcing any distributional invariances. Again, since we only need an autoencoder that can fully reconstruct the

Table 9: Linear Mixing Dataset R^2 scores, **Dynamic SCM** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{2d}$. The three sections from top to bottom correspond to Min-Max, MMD, and the combination.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
8	0.48±0.01	0.98±0.00	0.97±0.01	0.95±0.01	0.30±0.02	0.02±0.00	0.03±0.01	0.03±0.01
16	0.43±0.04	0.73±0.02	0.93±0.02	0.98±0.00	0.33±0.05	0.14±0.01	0.06±0.00	0.03±0.00
32	0.35±0.00	0.51±0.02	0.81±0.00	0.89±0.00	0.38±0.05	0.24±0.01	0.14±0.01	0.11±0.00
64	0.27±0.01	0.34±0.00	0.63±0.01	0.80±0.00	0.55±0.02	0.33±0.01	0.19±0.00	0.14±0.01
8	0.60±0.06	0.92±0.03	0.99±0.00	0.99±0.00	0.25±0.04	0.05±0.01	0.02±0.00	0.02±0.00
16	0.46±0.03	0.74±0.01	0.92±0.02	0.98±0.01	0.29±0.04	0.13±0.02	0.04±0.00	0.04±0.01
32	0.35±0.02	0.56±0.01	0.74±0.03	0.91±0.01	0.37±0.04	0.20±0.01	0.11±0.01	0.08±0.02
64	0.28±0.00	0.31±0.01	0.53±0.01	0.72±0.00	0.47±0.02	0.30±0.01	0.20±0.00	0.16±0.00
8	0.61±0.05	0.98±0.00	0.99±0.00	0.99±0.00	0.21±0.01	0.02±0.00	0.02±0.00	0.02±0.00
16	0.51±0.01	0.80±0.02	0.96±0.01	0.98±0.01	0.28±0.05	0.10±0.02	0.04±0.00	0.04±0.00
32	0.36±0.02	0.65±0.03	0.82±0.01	0.91±0.00	0.35±0.05	0.18±0.02	0.09±0.01	0.09±0.00
64	0.31±0.00	0.38±0.00	0.64±0.01	0.81±0.00	0.48±0.03	0.31±0.01	0.18±0.00	0.11±0.00

input, there is no need for training multiple perfect autoencoders, hence there is no standard error reported for such entries. We then take the perfectly trained autoencoder and enforce the distributional invariance penalty with 5 seeds, and present the results in the bottom row per each penalty. Our autoencoder architecture comprises a ResNet18 (He et al., 2016) encoder with standard deconvolutional layers in the decoder. We closely follow the architecture from Ahuja et al. (2022b). In all experiments, the encoder's output is 128-dimensional, and the invariance penalty is enforced on the first 64-dimensional block of encoder's output. For sample reconstructions see Figure 5.

B.4 Unlabeled colored MNIST

B.4.1 Data Generation and Model Architecture

Data Generation All of the digit pixels will be coloured according to $z_{\mathcal{U}}$. The background remains untouched (coloured digits on black background). Now we describe the colouring scheme across domains and different SCMs.
Independent SCM. For each domain $i \in [k]$ we sample $l^i_c, h^i_c \sim \text{Uniform}[0, 1]$, such that $c \sim \text{Uniform}[l^i_c, h^i_c]$, where $c \in \{r, g, b\}$ denotes the colour channel. In other terms, each of the RGB channels comes from a uniform distribution that is unique to each domain i. The digits then are coloured by sampling $z_{\mathcal{U}} = (r, g, b)$ for each domain.
Dynamic SCM. To obtain a Dynamic SCM, we introduce a probabilistic relation among digits and $z_{\mathcal{U}}$ as follows. For any domain i, we sample each channel $c \sim \text{Uniform}[l^i_c, h^i_c]$ with a probability of 0.2, and with a probability of 0.8 we introduce the following relation among digits and the colours. If the image contains digits from 0-4, the channels are sampled according to $c \sim \text{Uniform}[l^i_c, (l^i_c + h^i_c)/2]$, and if the image contains digits from 5-9, the channels are sampled according to $c \sim \text{Uniform}[(l^i_c + h^i_c)/2, h^i_c]$. In simple words, most of the time we introduce a correlation between the digits and the colours, and for a small portion of the dataset, digits and colours are sampled independently, thus overall, we achieve a Dynamic SCM.

Model Architecture All experiments are carried out in two stages similar to polynomial mixing, and balls image datasets. The architectures of the autoencoders at stages 1,2 are given in Tables 21,22.

The results for the Independent and Dynamic SCM are given in Tables 27,28, respectively.

B.5 $\beta-$VAE Baseline

For all experiments in sections B.1, B.2, B.3, we implement a baseline based on $\beta-$VAE (Higgins et al., 2017) and report the metrics in tables 10, 11 for Linear Mixing, in tables 19, 20 for Polynomial Mixing, and in tables 25, 26 for the Balls image dataset. To obtain the scores for this baseline, we employ a similar 2 stage procedure,

Table 10: β-VAE - Linear Mixing Dataset R^2 scores, **Independent SCM** DGP. The results are averaged over 5 seeds. Each set of 4 rows correspond to a specific d and from top to bottom denote the R^2 scores before training β-VAE, and after training with $\beta \in [0.1, 1.0, 10.0]$. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{2d}$.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
8	0.11±0.08	−0.21±0.25	−0.59±0.31	−0.57±0.30	0.69±0.16	0.81±0.05	0.78±0.04	0.81±0.03
	0.15±0.09	−0.50±0.40	−0.53±0.41	−0.31±0.32	0.83±0.08	0.85±0.04	0.77±0.05	0.80±0.06
	0.17±0.06	−0.86±0.40	−0.84±0.51	−0.92±0.47	0.87±0.04	0.86±0.03	0.85±0.03	0.84±0.03
	0.05±0.08	−0.56±0.30	−0.95±0.35	−1.24±0.39	0.81±0.10	0.92±0.01	0.85±0.04	0.90±0.02
16	0.42±0.29	−0.61±0.23	−1.46±0.26	−1.57±0.43	0.31±0.27	0.63±0.04	0.49±0.03	0.51±0.02
	0.19±0.03	0.04±0.04	−0.24±0.26	−0.49±0.42	0.83±0.07	0.92±0.02	0.90±0.01	0.90±0.00
	0.19±0.03	−0.03±0.22	−0.42±0.29	−0.60±0.25	0.86±0.05	0.94±0.02	0.92±0.01	0.90±0.00
	0.15±0.06	−0.19±0.30	−0.09±0.07	−0.52±0.29	0.94±0.01	0.96±0.01	0.94±0.00	0.94±0.01
32	0.35±0.17	−0.49±0.23	−1.10±0.48	−1.24±0.32	0.25±0.17	0.41±0.04	0.35±0.02	0.28±0.01
	0.23±0.01	0.10±0.03	0.06±0.03	0.01±0.04	0.91±0.02	0.92±0.00	0.92±0.00	0.92±0.00
	0.20±0.01	0.12±0.02	0.06±0.01	0.02±0.02	0.94±0.01	0.95±0.00	0.95±0.00	0.93±0.00
	0.22±0.01	0.13±0.02	0.06±0.03	−0.11±0.17	0.93±0.01	0.94±0.00	0.95±0.00	0.94±0.00
64	0.30±0.14	−0.52±0.41	−0.72±0.31	−0.85±0.33	0.26±0.09	0.33±0.05	0.27±0.02	0.19±0.01
	0.25±0.01	0.18±0.00	0.14±0.01	0.10±0.01	0.91±0.01	0.95±0.00	0.95±0.00	0.95±0.00
	0.24±0.02	0.18±0.01	0.13±0.02	0.07±0.03	0.91±0.01	0.94±0.00	0.95±0.00	0.94±0.00
	0.27±0.02	0.23±0.01	0.16±0.01	0.11±0.03	0.89±0.01	0.92±0.00	0.93±0.00	0.92±0.00

Table 11: β-VAE - Linear Mixing Dataset R^2 scores, **Dynamic SCM** DGP. The results are averaged over 5 seeds. Each set of 4 rows correspond to a specific d and from top to bottom denote the R^2 scores before training β-VAE, and after training with $\beta \in [0.1, 1.0, 10.0]$. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{2d}$.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
8	0.08±0.10	−0.23±0.32	−0.54±0.30	−0.51±0.27	0.73±0.13	0.81±0.05	0.79±0.04	0.81±0.04
	0.26±0.02	−0.44±0.39	−0.35±0.27	−0.22±0.57	0.85±0.05	0.85±0.04	0.77±0.05	0.82±0.06
	0.17±0.09	−0.61±0.34	−0.44±0.32	−0.74±0.39	0.88±0.03	0.87±0.03	0.84±0.02	0.83±0.04
	−0.17±0.33	−0.52±0.51	−0.71±1.07	−1.04±0.33	0.85±0.04	0.73±0.19	0.89±0.04	0.90±0.03
16	−0.53±0.43	−0.56±0.54	−1.18±0.44	−1.23±0.32	0.33±0.28	0.62±0.04	0.50±0.03	0.51±0.02
	0.19±0.03	0.04±0.03	−0.15±0.16	−0.32±0.31	0.85±0.05	0.91±0.03	0.90±0.02	0.91±0.01
	0.17±0.09	0.00±0.08	−0.37±0.31	−0.51±0.49	0.88±0.03	0.94±0.01	0.93±0.01	0.92±0.01
	0.13±0.08	−0.13±0.32	−0.15±0.16	−0.56±0.28	0.94±0.01	0.95±0.01	0.93±0.00	0.93±0.00
32	−0.33±0.40	−0.35±0.11	−0.79±0.36	−0.86±0.21	0.29±0.15	0.41±0.03	0.35±0.02	0.29±0.01
	0.22±0.01	0.12±0.02	0.05±0.02	−0.01±0.12	0.91±0.02	0.93±0.00	0.93±0.00	0.92±0.00
	0.20±0.01	0.12±0.02	0.07±0.00	−0.04±0.10	0.93±0.01	0.94±0.00	0.95±0.00	0.93±0.01
	0.23±0.01	0.15±0.02	0.10±0.02	0.03±0.04	0.93±0.01	0.93±0.01	0.94±0.00	0.90±0.01
64	−0.16±0.11	−0.31±0.22	−0.47±0.20	−0.53±0.16	0.34±0.05	0.35±0.04	0.27±0.02	0.19±0.01
	0.24±0.01	0.19±0.00	0.15±0.01	0.11±0.01	0.90±0.01	0.95±0.00	0.95±0.00	0.94±0.00
	0.20±0.01	0.20±0.01	0.15±0.01	0.10±0.03	0.93±0.01	0.94±0.00	0.94±0.00	0.93±0.00
	0.29±0.01	0.23±0.01	0.18±0.01	0.16±0.03	0.88±0.01	0.92±0.00	0.92±0.00	0.89±0.01

Table 12: Polynomial Encoder.

Layer	Input Size	Output Size	Bias	Activation
Linear (1)	n	$n/2$	False	LeakyReLU(0.5)
Linear (2)	$n/2$	$n/2$	False	LeakyReLU(0.5)
Linear (3)	$n/2$	d	False	-

Table 13: Polynomial Mixing Dataset R^2 scores, **Independent** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{200}$. Penalty used here is Min-Max. Top section and bottom section correspond to polynomial degrees of 2 and 3. For each dimension d, the top row corresponds to the scores after training the autoencoder with reconstruction objective only, and the bottom row denotes the scores after enforcing distributional invariances in 5 different runs.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 2$	$k = 4$	$k = 8$	$k = 16$
6	0.03	0.08	0.09	0.02	0.96	0.83	0.89	0.98
	0.42±0.04	0.62±0.01	0.99±0.00	0.99±0.00	0.01±0.00	0.01±0.00	0.00±0.00	0.00±0.00
8	0.26	0.15	0.08	0.12	0.80	0.92	0.91	0.95
	0.34±0.02	0.99±0.00	0.98±0.01	0.97±0.01	0.01±0.00	0.01±0.00	0.00±0.00	0.01±0.00
10	0.22	0.04	0.03	0.05	0.79	0.97	0.98	0.96
	0.32±0.03	0.90±0.04	0.94±0.04	0.92±0.04	0.04±0.00	0.01±0.00	0.01±0.00	0.01±0.00
12	0.07	0.19	0.04	0.17	0.95	0.90	0.98	0.88
	0.38±0.03	0.83±0.01	0.89±0.00	0.95±0.02	0.06±0.02	0.01±0.00	0.01±0.00	0.01±0.00
14	0.12	0.17	0.17	0.11	0.94	0.93	0.86	0.93
	0.34±0.03	0.67±0.01	0.95±0.02	0.96±0.02	0.05±0.01	0.04±0.01	0.01±0.00	0.01±0.00
6	0.16	0.04	0.05	0.09	0.83	0.96	0.95	0.92
	0.33±0.01	0.62±0.00	0.80±0.00	0.97±0.01	0.05±0.02	0.00±0.00	0.00±0.00	0.00±0.00
8	0.06	0.25	0.23	0.20	0.95	0.87	0.83	0.81
	0.45±0.06	0.84±0.03	0.93±0.01	0.92±0.01	0.06±0.04	0.01±0.00	0.01±0.00	0.00±0.00
10	0.13	0.15	0.11	0.04	0.89	0.80	0.93	0.96
	0.48±0.01	0.83±0.00	0.87±0.00	0.93±0.01	0.02±0.00	0.01±0.00	0.01±0.00	0.01±0.00
12	0.20	0.21	0.18	0.19	0.85	0.84	0.85	0.82
	0.52±0.02	0.80±0.03	0.88±0.01	0.95±0.01	0.07±0.02	0.02±0.00	0.01±0.00	0.02±0.00
14	0.18	0.06	0.29	0.27	0.74	0.80	0.76	0.71
	0.26±0.02	0.32±0.01	0.91±0.01	0.93±0.00	0.10±0.01	0.03±0.00	0.01±0.00	0.01±0.00

where in stage 1, we train an autoencoder with reconstruction objective only. Then at stage 2, we employ the KL divergence constraint from Higgins et al. (2017) on the representations obtained from the autoencoder at stage 1, and randomly divide the resulting $\hat{z}$ into two halves to represent $\hat{z}_{\hat{\mathcal{S}}}$,$\hat{z}_{\hat{\mathcal{U}}}$, and compute the R^2 scores against $z_{\mathcal{S}}$,$z_{\mathcal{U}}$. Note that unlike our method that directly affects a known subset of $\hat{z}$ to obtain $\hat{z}_{\hat{\mathcal{S}}}$, we have no way of knowing beforehand such subsets with the KL divergence penalty of Higgins et al. (2017), hence the need for randomly selecting such features.

Training Details and Hyperparameter Selection It should be noted that hyperparameter selection in unsupervised scenarios such as this work differs crucially from the conventional setups as in practice one does not have access to the ground-truth latents z. Therefore we focus on using default hyperparameters and demonstrate the robustness and versatility of our approach across the different datasets. We train all models with Adam (Kingma and Ba, 2014) optimizer with a learning rate of 10^{-3} without weight decay, $\epsilon = 10^{-8}, \beta_1 = 0.9, \beta_2 = 0.999$. We reduce the learning rate by a factor of 0.5 if the training objective is not improved for 10 epochs. This drop is followed by a cool-down period of 10 epochs, and the learning rate cannot decrease to lower than 10^{-4}. For all datasets we use a batch size of 1024 and early stop the training at 2000 steps. The weight of invariance penalty is always set to 1.0, regardless of the combination of penalties used. To ensure the robustness of the Min-Max penalty, we enforce the support invariance not just on the minimum and maximum across a batch, rather, we sort the batch and for each component of $z_{\mathcal{S}}$ take the top 10 for computing the penalty. For MMD penalty we always use the standard RBF kernel with a default bandwidth of 1.0, with the only exception of using an adaptive bandwidth for linear mixing experiments.

Table 14: Polynomial Mixing Dataset R^2 scores, **Independent** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{200}$. Penalty used here is MMD. Top section and bottom section correspond to polynomial degrees of 2 and 3. For each dimension d, the top and bottom rows correspond to the scores after Stages 1, 2.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 2$	$k = 4$	$k = 8$	$k = 16$
6	0.03 0.54±0.04	0.08 0.55±0.04	0.09 0.99±0.00	0.02 0.99±0.00	0.96 0.01±0.00	0.83 0.01±0.00	0.86 0.01±0.00	0.98 0.00±0.00
8	0.26 0.42±0.05	0.15 0.76±0.00	0.08 0.98±0.01	0.12 0.97±0.02	0.80 0.08±0.05	0.92 0.01±0.00	0.91 0.00±0.00	0.92 0.01±0.00
10	0.22 0.52±0.04	0.04 0.81±0.00	0.03 0.86±0.00	0.05 0.91±0.04	0.79 0.05±0.01	0.97 0.01±0.00	0.98 0.01±0.00	0.96 0.01±0.00
12	0.07 0.48±0.01	0.19 0.65±0.02	0.04 0.90±0.00	0.17 0.98±0.00	0.95 0.12±0.03	0.90 0.02±0.00	0.98 0.01±0.00	0.88 0.01±0.00
14	0.12 0.52±0.05	0.17 0.55±0.01	0.17 0.99±0.00	0.11 0.98±0.01	0.94 0.05±0.01	0.93 0.06±0.02	0.86 0.01±0.00	0.93 0.01±0.00
6	0.16 0.46±0.05	0.04 0.63±0.00	0.05 0.80±0.00	0.09 0.98±0.01	0.83 0.05±0.03	0.96 0.01±0.00	0.95 0.00±0.00	0.92 0.00±0.00
8	0.06 0.54±0.02	0.25 0.73±0.01	0.23 0.92±0.04	0.20 0.98±0.00	0.95 0.07±0.04	0.87 0.02±0.00	0.83 0.01±0.00	0.81 0.00±0.00
10	0.13 0.49±0.04	0.15 0.72±0.01	0.11 0.87±0.00	0.04 0.98±0.00	0.89 0.05±0.01	0.80 0.01±0.00	0.93 0.01±0.00	0.96 0.01±0.00
12	0.20 0.51±0.02	0.21 0.74±0.02	0.18 0.89±0.00	0.19 0.97±0.00	0.85 0.12±0.02	0.84 0.04±0.00	0.85 0.01±0.00	0.82 0.01±0.00
14	0.18 0.38±0.02	0.06 0.40±0.00	0.29 0.94±0.00	0.27 0.95±0.00	0.74 0.08±0.00	0.80 0.03±0.00	0.76 0.02±0.00	0.71 0.01±0.00

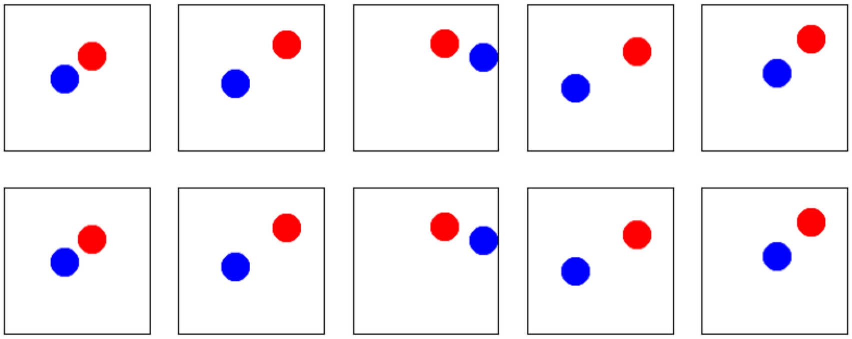

Figure 5: The top row shows the inputs to the image autoencoder, and the bottom row shows model's reconstructions.

Table 15: Polynomial Mixing Dataset R^2 scores, **Independent** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{200}$. Penalty used here is MMD+Min-Max. Top section and bottom section correspond to polynomial degrees of 2 and 3. For each dimension d, the top and bottom rows correspond to the scores after Stages 1, 2.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 2$	$k = 4$	$k = 8$	$k = 16$
6	0.03	0.08	0.09	0.02	0.96	0.83	0.89	0.98
	0.60±0.06	0.60±0.00	0.99±0.00	0.99±0.01	0.02±0.01	0.00±0.00	0.01±0.00	0.00±0.00
8	0.26	0.15	0.08	0.12	0.80	0.92	0.91	0.92
	0.52±0.04	0.98±0.00	0.97±0.01	0.99±0.00	0.03±0.02	0.00±0.00	0.00±0.00	0.01±0.00
10	0.22	0.04	0.03	0.05	0.79	0.97	0.98	0.96
	0.68±0.03	0.96±0.01	0.95±0.03	0.94±0.01	0.02±0.00	0.01±0.00	0.01±0.00	0.01±0.00
12	0.07	0.19	0.04	0.17	0.95	0.90	0.98	0.88
	0.63±0.04	0.92±0.00	0.90±0.00	0.97±0.01	0.02±0.00	0.01±0.00	0.01±0.00	0.01±0.00
14	0.12	0.17	0.17	0.11	0.94	0.93	0.86	0.93
	0.63±0.02	0.65±0.00	0.96±0.02	0.98±0.01	0.04±0.02	0.04±0.01	0.01±0.00	0.01±0.00
6	0.16	0.04	0.05	0.10	0.83	0.96	0.95	0.92
	0.44±0.03	0.63±0.00	0.80±0.00	0.96±0.02	0.03±0.01	0.00±0.00	0.00±0.00	0.01±0.00
8	0.06	0.25	0.23	0.20	0.95	0.87	0.83	0.81
	0.63±0.04	0.91±0.02	0.93±0.02	0.97±0.01	0.03±0.02	0.01±0.00	0.01±0.00	0.00±0.00
10	0.13	0.15	0.11	0.04	0.89	0.80	0.93	0.96
	0.62±0.04	0.79±0.04	0.85±0.01	0.97±0.00	0.02±0.00	0.01±0.00	0.01±0.00	0.01±0.00
12	0.20	0.21	0.18	0.20	0.85	0.84	0.85	0.82
	0.62±0.02	0.81±0.01	0.89±0.00	0.97±0.00	0.08±0.02	0.02±0.00	0.01±0.00	0.01±0.00
14	0.18	0.06	0.29	0.27	0.74	0.80	0.76	0.71
	0.36±0.01	0.39±0.00	0.93±0.02	0.95±0.00	0.10±0.01	0.03±0.00	0.02±0.00	0.01±0.00

Table 16: Polynomial Mixing Dataset R^2 scores, **Dynamic SCM** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{200}$. Penalty used here is Min-Max. Top section and bottom section correspond to polynomial degrees of 2 and 3. For each dimension d, the top and bottom rows correspond to the scores after Stages 1, 2.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
6	0.05	0.02	0.01	0.05	0.95	0.97	0.99	0.95
	0.28±0.04	0.96±0.01	0.71±0.00	0.99±0.00	0.03±0.02	0.01±0.00	0.01±0.00	0.01±0.00
8	0.20	0.18	0.14	0.01	0.88	0.87	0.85	0.99
	0.39±0.03	0.71±0.01	0.78±0.01	0.93±0.04	0.05±0.01	0.01±0.00	0.01±0.00	0.01±0.00
10	0.19	0.04	0.02	0.05	0.86	0.98	0.99	0.97
	0.35±0.04	0.76±0.02	0.98±0.00	0.99±0.00	0.07±0.03	0.01±0.00	0.01±0.00	0.01±0.00
12	0.05	0.18	0.13	0.11	0.97	0.90	0.92	0.93
	0.41±0.03	0.80±0.01	0.97±0.01	0.97±0.01	0.09±0.03	0.01±0.00	0.01±0.00	0.01±0.00
14	0.11	0.16	0.17	0.04	0.95	0.94	0.83	0.98
	0.39±0.01	0.69±0.01	0.95±0.02	0.97±0.01	0.06±0.01	0.04±0.01	0.02±0.00	0.01±0.00
6	0.30	0.16	0.30	0.07	0.71	0.88	0.74	0.94
	0.34±0.01	0.92±0.01	0.98±0.00	0.99±0.00	0.03±0.02	0.01±0.00	0.01±0.00	0.01±0.00
8	0.18	0.06	0.13	0.21	0.86	0.98	0.89	0.80
	0.45±0.05	0.90±0.04	0.95±0.01	0.95±0.02	0.07±0.04	0.01±0.00	0.01±0.00	0.01±0.00
10	0.21	0.12	0.22	0.07	0.87	0.84	0.87	0.94
	0.42±0.02	0.70±0.00	0.95±0.01	0.95±0.01	0.08±0.03	0.01±0.00	0.01±0.00	0.01±0.00
12	0.15	0.23	0.14	0.17	0.88	0.81	0.87	0.85
	0.44±0.02	0.81±0.02	0.87±0.03	0.94±0.01	0.11±0.01	0.02±0.00	0.01±0.00	0.01±0.00
14	0.22	0.16	0.25	0.22	0.69	0.79	0.80	0.73
	0.34±0.02	0.63±0.01	0.88±0.02	0.92±0.00	0.08±0.01	0.03±0.00	0.02±0.00	0.02±0.00

Table 17: Polynomial Mixing Dataset R^2 scores, **Dynamic SCM** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{200}$. Penalty used here is MMD. Top section and bottom section correspond to polynomial degrees of 2 and 3. For each dimension d, the top and bottom rows correspond to the scores after Stages 1, 2.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
6	0.05	0.02	0.01	0.05	0.95	0.97	0.99	0.95
	0.25±0.06	0.85±0.03	0.64±0.02	0.99±0.00	0.08±0.05	0.05±0.01	0.01±0.00	0.01±0.00
8	0.20	0.18	0.14	0.01	0.88	0.87	0.85	0.99
	0.41±0.03	0.58±0.03	0.73±0.02	0.95±0.03	0.08±0.04	0.08±0.03	0.02±0.00	0.01±0.00
10	0.19	0.04	0.02	0.04	0.86	0.98	0.99	0.97
	0.41±0.02	0.53±0.02	0.91±0.01	0.96±0.02	0.10±0.03	0.04±0.01	0.02±0.00	0.01±0.00
12	0.05	0.18	0.13	0.11	0.97	0.90	0.92	0.93
	0.39±0.03	0.54±0.02	0.89±0.01	0.99±0.00	0.10±0.03	0.04±0.01	0.03±0.01	0.01±0.00
14	0.11	0.16	0.17	0.04	0.95	0.94	0.83	0.98
	0.38±0.02	0.48±0.02	0.89±0.01	0.99±0.00	0.09±0.02	0.10±0.02	0.03±0.00	0.01±0.00
6	0.30	0.16	0.30	0.07	0.71	0.88	0.74	0.94
	0.41±0.03	0.77±0.03	0.86±0.03	0.98±0.00	0.04±0.01	0.05±0.02	0.02±0.01	0.01±0.00
8	0.18	0.06	0.13	0.21	0.86	0.98	0.89	0.80
	0.46±0.02	0.66±0.02	0.84±0.03	0.98±0.00	0.05±0.02	0.02±0.01	0.02±0.00	0.01±0.00
10	0.21	0.12	0.22	0.07	0.87	0.84	0.87	0.94
	0.42±0.03	0.55±0.02	0.90±0.01	0.96±0.00	0.08±0.01	0.03±0.00	0.03±0.00	0.01±0.00
12	0.15	0.23	0.14	0.17	0.88	0.81	0.87	0.85
	0.45±0.01	0.64±0.02	0.85±0.01	0.96±0.00	0.11±0.01	0.07±0.01	0.04±0.00	0.01±0.00
14	0.22	0.16	0.25	0.22	0.69	0.79	0.80	0.73
	0.34±0.01	0.52±0.01	0.85±0.01	0.93±0.01	0.09±0.01	0.09±0.02	0.05±0.00	0.02±0.00

Table 18: Polynomial Mixing Dataset R^2 scores, **Dynamic SCM** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{200}$. Penalty used here is MMD+Min-Max. Top section and bottom section correspond to polynomial degrees of 2 and 3. For each dimension d, the top and bottom rows correspond to the scores after Stages 1, 2.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
6	0.05	0.16	0.01	0.05	0.95	0.88	0.99	0.95
	0.32±0.04	0.81±0.03	0.69±0.01	0.99±0.00	0.02±0.00	0.05±0.02	0.01±0.00	0.01±0.00
8	0.20	0.06	0.14	0.01	0.88	0.98	0.85	0.99
	0.43±0.02	0.82±0.02	0.80±0.00	0.95±0.03	0.05±0.01	0.02±0.01	0.01±0.00	0.01±0.00
10	0.19	0.04	0.02	0.04	0.86	0.98	0.99	0.97
	0.53±0.02	0.65±0.02	0.90±0.01	0.99±0.00	0.07±0.01	0.02±0.00	0.02±0.00	0.01±0.00
12	0.05	0.18	0.13	0.11	0.97	0.90	0.92	0.93
	0.48±0.03	0.72±0.02	0.88±0.03	0.98±0.00	0.07±0.02	0.03±0.00	0.02±0.00	0.01±0.00
14	0.11	0.16	0.17	0.04	0.95	0.94	0.83	0.98
	0.49±0.02	0.56±0.02	0.89±0.01	0.97±0.02	0.06±0.01	0.07±0.01	0.03±0.00	0.01±0.00
6	0.30	0.02	0.30	0.07	0.71	0.97	0.74	0.94
	0.36±0.03	0.88±0.03	0.87±0.04	0.99±0.00	0.04±0.02	0.05±0.01	0.02±0.00	0.01±0.00
8	0.18	0.18	0.13	0.21	0.86	0.87	0.89	0.80
	0.53±0.04	0.62±0.03	0.83±0.04	0.98±0.00	0.06±0.03	0.04±0.01	0.02±0.00	0.01±0.00
10	0.21	0.12	0.22	0.07	0.87	0.84	0.87	0.94
	0.53±0.02	0.62±0.01	0.90±0.01	0.96±0.00	0.08±0.03	0.03±0.00	0.03±0.00	0.01±0.00
12	0.15	0.23	0.14	0.17	0.88	0.81	0.87	0.85
	0.52±0.02	0.73±0.02	0.83±0.01	0.96±0.00	0.10±0.02	0.05±0.00	0.04±0.00	0.01±0.00
14	0.22	0.16	0.25	0.22	0.69	0.79	0.80	0.73
	0.41±0.01	0.55±0.01	0.85±0.01	0.93±0.00	0.08±0.01	0.07±0.01	0.04±0.00	0.02±0.00

Table 19: β-VAE - Polynomial Mixing Dataset R^2 scores, **Independent SCM** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{200}$. Top section and bottom section correspond to polynomial degrees of 2 and 3. Each set of 4 rows correspond to a specific d and from top to bottom denote the R^2 scores before training β-VAE, and after training with $\beta \in [0.1, 1.0, 10.0]$.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
6	0.37±0.02	0.19±0.05	0.35±0.00	0.19±0.04	0.96±0.02	0.99±0.00	0.98±0.00	0.99±0.00
	0.12±0.07	0.03±0.01	0.08±0.05	0.02±0.01	0.91±0.04	0.95±0.02	0.84±0.07	0.95±0.02
	0.07±0.05	0.02±0.00	0.02±0.00	0.01±0.00	0.96±0.01	0.94±0.04	0.90±0.08	0.93±0.04
	0.09±0.06	0.01±0.00	0.02±0.00	0.01±0.00	0.97±0.01	0.96±0.03	0.95±0.03	0.92±0.06
8	0.05±0.01	0.12±0.05	0.05±0.03	0.06±0.04	0.97±0.00	0.92±0.04	0.95±0.04	0.95±0.04
	0.10±0.04	0.03±0.01	0.08±0.06	0.01±0.00	0.93±0.03	0.92±0.03	0.95±0.03	0.93±0.03
	0.07±0.03	0.02±0.00	0.04±0.01	0.03±0.01	0.91±0.02	0.96±0.02	0.94±0.03	0.91±0.04
	0.05±0.02	0.04±0.01	0.07±0.04	0.03±0.01	0.93±0.03	0.92±0.03	0.95±0.04	0.93±0.04
10	0.06±0.03	0.05±0.01	0.04±0.02	0.02±0.01	0.80±0.03	0.83±0.02	0.77±0.04	0.80±0.02
	0.05±0.01	0.09±0.03	0.05±0.02	0.05±0.03	0.95±0.01	0.94±0.03	0.98±0.00	0.95±0.01
	0.09±0.03	0.12±0.04	0.04±0.01	0.09±0.04	0.94±0.02	0.96±0.02	0.95±0.03	0.95±0.04
	0.08±0.03	0.03±0.00	0.04±0.01	0.07±0.04	0.97±0.00	0.96±0.03	0.99±0.00	0.98±0.01
12	0.04±0.02	0.03±0.01	0.02±0.00	0.01±0.00	0.70±0.02	0.79±0.01	0.72±0.02	0.69±0.02
	0.15±0.03	0.07±0.02	0.07±0.03	0.05±0.02	0.91±0.02	0.96±0.01	0.97±0.00	0.90±0.03
	0.08±0.02	0.09±0.03	0.06±0.03	0.10±0.03	0.97±0.00	0.95±0.01	0.95±0.02	0.96±0.02
	0.08±0.03	0.09±0.03	0.03±0.01	0.02±0.00	0.96±0.01	0.97±0.00	0.95±0.03	0.99±0.00
14	0.02±0.00	0.03±0.01	0.02±0.01	0.01±0.00	0.70±0.02	0.68±0.04	0.65±0.03	0.58±0.02
	0.14±0.01	0.08±0.02	0.17±0.02	0.07±0.02	0.95±0.01	0.95±0.00	0.96±0.01	0.92±0.02
	0.08±0.02	0.07±0.02	0.07±0.01	0.11±0.02	0.94±0.01	0.94±0.01	0.96±0.02	0.93±0.02
	0.05±0.00	0.09±0.03	0.09±0.03	0.03±0.01	0.95±0.02	0.97±0.00	0.96±0.02	0.97±0.01
6	0.35±0.03	0.13±0.04	0.28±0.01	0.33±0.00	0.97±0.03	0.97±0.02	1.00±0.00	1.00±0.00
	0.13±0.06	0.02±0.01	0.02±0.00	0.03±0.01	0.86±0.06	0.93±0.02	0.97±0.01	0.97±0.01
	0.10±0.05	0.02±0.00	0.01±0.00	0.03±0.01	0.90±0.03	0.93±0.03	0.98±0.01	0.93±0.04
	0.05±0.03	0.02±0.01	0.01±0.00	0.02±0.01	0.97±0.02	0.95±0.03	0.99±0.00	0.99±0.00
8	0.19±0.04	0.09±0.03	0.05±0.02	0.14±0.05	0.82±0.06	0.93±0.02	0.94±0.02	0.84±0.07
	0.08±0.03	0.14±0.05	0.08±0.04	0.05±0.03	0.87±0.02	0.92±0.03	0.96±0.02	0.94±0.04
	0.12±0.04	0.13±0.04	0.05±0.03	0.07±0.02	0.90±0.04	0.92±0.03	0.98±0.00	0.91±0.03
	0.05±0.01	0.13±0.04	0.03±0.01	0.01±0.00	0.94±0.02	0.97±0.02	0.97±0.03	0.88±0.04
10	0.08±0.04	0.05±0.01	0.14±0.03	0.07±0.03	0.77±0.05	0.75±0.03	0.67±0.05	0.76±0.04
	0.12±0.02	0.07±0.02	0.13±0.04	0.08±0.04	0.95±0.01	0.90±0.03	0.88±0.04	0.94±0.02
	0.15±0.01	0.08±0.04	0.12±0.03	0.15±0.03	0.93±0.02	0.95±0.01	0.92±0.03	0.90±0.03
	0.08±0.03	0.04±0.01	0.05±0.01	0.10±0.04	0.94±0.01	0.93±0.03	0.96±0.02	0.98±0.00
12	0.04±0.01	0.06±0.01	0.06±0.01	0.06±0.01	0.66±0.03	0.63±0.04	0.65±0.03	0.65±0.02
	0.16±0.03	0.13±0.02	0.15±0.02	0.15±0.03	0.89±0.02	0.91±0.02	0.88±0.02	0.86±0.02
	0.15±0.01	0.13±0.01	0.11±0.01	0.12±0.02	0.89±0.02	0.92±0.02	0.92±0.01	0.90±0.02
	0.13±0.02	0.11±0.03	0.12±0.02	0.11±0.03	0.93±0.02	0.91±0.03	0.93±0.02	0.96±0.01
14	0.07±0.02	0.04±0.01	0.07±0.01	0.08±0.01	0.54±0.02	0.51±0.04	0.52±0.01	0.51±0.03
	0.13±0.01	0.09±0.02	0.16±0.02	0.11±0.03	0.81±0.02	0.81±0.01	0.88±0.01	0.80±0.02
	0.10±0.01	0.07±0.01	0.10±0.02	0.13±0.02	0.83±0.01	0.83±0.02	0.91±0.01	0.85±0.02
	0.08±0.01	0.05±0.01	0.07±0.01	0.10±0.03	0.85±0.01	0.92±0.01	0.92±0.02	0.91±0.02

Table 20: β-VAE - Polynomial Mixing Dataset R^2 scores, **Dynamic SCM** DGP. The results are averaged over 5 seeds. $\hat{z}, z \in R^d$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{d/2}$ and $x = g(z) \in R^{200}$. Top section and bottom section correspond to polynomial degrees of 2 and 3. Each set of 4 rows correspond to a specific d and from top to bottom denote the R^2 scores before training β-VAE, and after training with $\beta \in [0.1, 1.0, 10.0]$.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
d	$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
6	0.27±0.05	0.41±0.06	0.23±0.04	0.34±0.00	0.98±0.01	0.92±0.06	0.99±0.00	0.99±0.00
	0.05±0.01	0.19±0.04	0.07±0.03	0.07±0.03	0.87±0.06	0.91±0.03	0.92±0.04	0.92±0.02
	0.07±0.03	0.25±0.04	0.07±0.03	0.07±0.03	0.91±0.04	0.83±0.05	0.89±0.04	0.89±0.04
	0.05±0.01	0.21±0.06	0.08±0.04	0.13±0.05	0.85±0.07	0.81±0.05	0.87±0.04	0.90±0.04
8	0.18±0.04	0.05±0.01	0.05±0.02	0.03±0.01	0.89±0.02	0.96±0.01	0.96±0.02	0.97±0.01
	0.14±0.05	0.12±0.03	0.04±0.01	0.03±0.01	0.92±0.02	0.94±0.03	0.94±0.01	0.93±0.02
	0.14±0.04	0.12±0.03	0.08±0.04	0.05±0.02	0.86±0.04	0.91±0.05	0.92±0.03	0.90±0.04
	0.14±0.03	0.16±0.02	0.11±0.04	0.08±0.04	0.90±0.01	0.91±0.06	0.88±0.04	0.89±0.04
10	0.06±0.02	0.02±0.01	0.03±0.00	0.05±0.02	0.81±0.02	0.83±0.02	0.79±0.03	0.77±0.03
	0.17±0.03	0.10±0.03	0.11±0.03	0.08±0.02	0.92±0.01	0.93±0.02	0.92±0.03	0.96±0.01
	0.17±0.02	0.08±0.01	0.14±0.03	0.09±0.04	0.89±0.02	0.92±0.03	0.95±0.01	0.92±0.03
	0.17±0.03	0.15±0.02	0.14±0.04	0.09±0.04	0.89±0.02	0.88±0.04	0.94±0.02	0.91±0.04
12	0.04±0.00	0.03±0.01	0.02±0.00	0.03±0.01	0.69±0.02	0.79±0.02	0.71±0.01	0.69±0.02
	0.18±0.01	0.14±0.02	0.10±0.03	0.10±0.03	0.91±0.01	0.93±0.02	0.90±0.01	0.96±0.01
	0.14±0.02	0.12±0.03	0.09±0.03	0.11±0.03	0.91±0.02	0.93±0.02	0.90±0.08	0.94±0.03
	0.14±0.03	0.13±0.03	0.09±0.03	0.14±0.03	0.90±0.02	0.90±0.03	0.90±0.01	0.95±0.02
14	0.03±0.01	0.03±0.01	0.02±0.00	0.01±0.00	0.68±0.01	0.69±0.03	0.64±0.01	0.58±0.02
	0.17±0.01	0.15±0.03	0.13±0.02	0.13±0.03	0.91±0.01	0.94±0.01	0.93±0.02	0.90±0.01
	0.19±0.03	0.16±0.03	0.15±0.03	0.15±0.03	0.91±0.01	0.91±0.02	0.88±0.02	0.87±0.03
	0.21±0.02	0.17±0.03	0.14±0.02	0.18±0.03	0.91±0.01	0.88±0.02	0.85±0.03	0.84±0.03
6	0.37±0.04	0.33±0.01	0.35±0.01	0.34±0.00	0.95±0.04	0.99±0.00	0.99±0.00	1.00±0.00
	0.07±0.04	0.10±0.05	0.09±0.06	0.03±0.01	0.94±0.02	0.94±0.02	0.95±0.02	0.90±0.04
	0.11±0.06	0.09±0.05	0.08±0.06	0.04±0.02	0.94±0.02	0.91±0.04	0.93±0.03	0.89±0.05
	0.06±0.03	0.09±0.05	0.08±0.05	0.06±0.03	0.92±0.05	0.92±0.03	0.96±0.02	0.91±0.04
8	0.17±0.04	0.10±0.04	0.05±0.02	0.11±0.04	0.86±0.06	0.92±0.03	0.95±0.02	0.87±0.05
	0.13±0.05	0.10±0.04	0.07±0.02	0.09±0.03	0.87±0.04	0.90±0.02	0.84±0.03	0.87±0.03
	0.16±0.05	0.09±0.04	0.09±0.02	0.08±0.02	0.87±0.03	0.90±0.03	0.87±0.03	0.85±0.03
	0.21±0.04	0.12±0.05	0.12±0.04	0.13±0.04	0.87±0.02	0.92±0.02	0.88±0.03	0.84±0.05
10	0.08±0.03	0.06±0.01	0.10±0.02	0.05±0.02	0.80±0.03	0.74±0.03	0.73±0.02	0.76±0.03
	0.19±0.03	0.17±0.02	0.16±0.03	0.09±0.02	0.88±0.02	0.82±0.04	0.85±0.03	0.85±0.03
	0.21±0.02	0.20±0.03	0.18±0.04	0.16±0.03	0.84±0.03	0.83±0.03	0.82±0.04	0.79±0.02
	0.22±0.02	0.20±0.02	0.16±0.04	0.19±0.02	0.81±0.04	0.81±0.03	0.79±0.04	0.79±0.03
12	0.05±0.01	0.06±0.01	0.07±0.02	0.04±0.00	0.65±0.03	0.65±0.03	0.64±0.03	0.67±0.02
	0.17±0.02	0.18±0.01	0.20±0.01	0.13±0.03	0.84±0.01	0.82±0.02	0.82±0.03	0.83±0.03
	0.17±0.02	0.18±0.01	0.22±0.01	0.13±0.02	0.81±0.01	0.84±0.02	0.82±0.02	0.80±0.02
	0.16±0.02	0.19±0.01	0.23±0.01	0.12±0.02	0.80±0.02	0.82±0.02	0.81±0.02	0.80±0.02
14	0.06±0.02	0.05±0.01	0.06±0.01	0.08±0.01	0.51±0.07	0.55±0.07	0.53±0.01	0.50±0.03
	0.18±0.03	0.20±0.01	0.21±0.01	0.18±0.01	0.82±0.03	0.82±0.02	0.83±0.02	0.79±0.01
	0.19±0.03	0.22±0.02	0.25±0.02	0.23±0.03	0.78±0.03	0.80±0.03	0.78±0.02	0.75±0.02
	0.21±0.02	0.22±0.02	0.25±0.01	0.25±0.03	0.75±0.03	0.80±0.02	0.75±0.03	0.75±0.02

Table 21: Autoencoder architecture for stage 1. First section presents the encoder layers, and the second section presents the decoder layers.

Layer	Input Size	Output Size	Bias	Activation	BatchNorm
Linear (1)	784	256	True	ReLU	True
Linear (2)	256	256	True	ReLU	True
Linear (3)	256	128	True	ReLU	True
Linear (1)	128	256	True	ReLU	True
Linear (2)	256	256	True	ReLU	True
Linear (3)	256	784	True	-	False

Table 22: Autoencoder architecture for stage 2. First section presents the encoder layers, and the second section presents the decoder layers.

Layer	Input Size	Output Size	Bias	Activation	BatchNorm
Linear (1)	128	200	True	LeakyReLU(0.2)	True
Linear (2)	200	200	True	LeakyReLU(0.2)	True
Linear (3)	200	200	True	LeakyReLU(0.2)	True
Linear (3)	200	128	True	-	False
Linear (1)	128	200	True	LeakyReLU(0.2)	True
Linear (2)	200	200	True	LeakyReLU(0.2)	True
Linear (3)	200	200	True	LeakyReLU(0.2)	True
Linear (3)	200	128	True	-	False

Table 23: Balls Dataset R^2, **Independent SCM** DGP. For each penalty, the top row corresponds to the scores after training the autoencoder, and the bottom row denotes the scores after enforcing distributional invariances. The results are averaged over 5 seeds. $\hat{z} \in R^{128}$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{64}$. The underlying latent $z \in R^4$. The sections are Min-Max, MMD, and the combination, respectively.

$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
0.99	0.99	0.99	0.99	0.99	0.98	0.98	0.97
0.94±0.00	0.89±0.01	0.85±0.04	0.65±0.01	0.88±0.00	0.66±0.02	0.56±0.04	0.19±0.01
0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.97
0.76±0.01	0.83±0.03	0.67±0.05	0.63±0.04	0.65±0.02	0.72±0.04	0.57±0.01	0.27±0.05
0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.97
0.77±0.01	0.91±0.01	0.86±0.03	0.81±0.04	0.30±0.01	0.11±0.01	0.14±0.01	0.18±0.02

Table 24: Balls Dataset R^2, **Dynamic SCM** DGP. For each penalty, the top row corresponds to the scores after training the autoencoder, and the bottom row denotes the scores after enforcing distributional invariances. The results are averaged over 5 seeds. $\hat{z} \in R^{128}$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{64}$. The underlying latent $z \in R^4$. The sections are Min-Max, MMD, and the combination, respectively.

$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
$k=2$	$k=4$	$k=8$	$k=16$	$k=2$	$k=4$	$k=8$	$k=16$
0.99	0.99	0.99	0.99	0.97	0.95	0.99	0.99
0.93±0.00	0.77±0.03	0.42±0.01	0.61±0.03	0.92±0.01	0.84±0.00	0.67±0.04	0.22±0.01
0.99	0.99	0.99	0.99	0.97	0.95	0.99	0.99
0.55±0.01	0.46±0.01	0.31±0.01	0.55±0.12	0.67±0.01	0.46±0.01	0.32±0.01	0.15±0.04
0.99	0.99	0.99	0.99	0.97	0.95	0.99	0.99
0.73±0.01	0.71±0.03	0.77±0.02	0.82±0.02	0.35±0.02	0.22±0.01	0.19±0.01	0.20±0.04

Table 25: β-VAE - Balls Dataset R^2, **Independent SCM** DGP. The top row corresponds to the scores after training the autoencoder and before enforcing β-VAE's KL divergence constraint, and the rest denote the disentanglement performance after training the β-VAE for each value of β. β-VAE The results are averaged over 5 seeds. $\hat{z} \in R^{128}$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{64}$. The underlying latent $z \in R^4$.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
β	$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 2$	$k = 4$	$k = 8$	$k = 16$
	0.28±0.02	0.24±0.04	0.25±0.02	0.25±0.03	0.29±0.06	0.25±0.03	0.34±0.04	0.26±0.04
0.1	0.97±0.00	0.97±0.00	0.96±0.00	0.96±0.00	0.98±0.00	0.98±0.00	0.97±0.00	0.93±0.01
1.0	0.96±0.00	0.94±0.01	0.93±0.00	0.92±0.01	0.95±0.01	0.97±0.00	0.97±0.00	0.93±0.01
10.0	0.83±0.03	0.82±0.03	0.77±0.02	0.77±0.02	0.92±0.01	0.93±0.01	0.94±0.00	0.87±0.01

Table 26: β-VAE - Balls Dataset R^2, **Dynamic SCM** DGP. The top row corresponds to the scores after training the autoencoder and before enforcing β-VAE's KL divergence constraint, and the rest denote the disentanglement performance after training the β-VAE for each value of β. β-VAE The results are averaged over 5 seeds. $\hat{z} \in R^{128}$ and $z_{\mathcal{S}}, z_{\mathcal{U}} \in R^{64}$. The underlying latent $z \in R^4$.

	$R^2_{\mathcal{S}}$				$R^2_{\mathcal{U}}$			
β	$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 2$	$k = 4$	$k = 8$	$k = 16$
	0.29±0.03	0.32±0.03	0.29±0.04	0.37±0.03	0.34±0.03	0.29±0.03	0.26±0.04	0.37±0.05
0.1	0.96±0.00	0.96±0.00	0.95±0.00	0.95±0.00	0.94±0.01	0.93±0.00	0.98±0.00	0.98±0.00
1.0	0.94±0.00	0.94±0.01	0.92±0.01	0.90±0.01	0.93±0.00	0.93±0.01	0.97±0.00	0.98±0.00
10.0	0.82±0.02	0.81±0.05	0.77±0.02	0.74±0.03	0.88±0.01	0.89±0.01	0.94±0.00	0.94±0.01

Table 27: MNIST, **Coloured Digits**, **Independent SCM** DGP. The top row corresponds to the scores after training the autoencoder, and the following rows denote the scores after enforcing distributional invariances through Min-Max penalty, MMD, and the combination, respectively. The results are averaged over 5 seeds. $\hat{z} \in R^{128}$ and $\hat{z}_{\hat{\mathcal{S}}}, \hat{z}_{\hat{\mathcal{U}}} \in R^{64}$.

Digits Classification Accuracy				Colours $R^2_{\mathcal{U}}$			
$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 2$	$k = 4$	$k = 8$	$k = 16$
0.87	0.33	0.33	0.32	0.76	0.67	0.73	0.74
0.71±0.02	0.59±0.01	0.58±0.01	0.66±0.01	0.72±0.02	0.55±0.01	0.51±0.03	0.49±0.02
0.72±0.01	0.70±0.01	0.73±0.01	0.73±0.01	0.77±0.01	0.64±0.01	0.64±0.02	0.63±0.02
0.73±0.02	0.70±0.02	0.74±0.00	0.74±0.01	0.73±0.02	0.54±0.02	0.38±0.01	0.28±0.01

Table 28: MNIST, **Coloured Digits**, **Dynamic SCM** DGP. The top row corresponds to the scores after training the autoencoder, and the following rows denote the scores after enforcing distributional invariances through Min-Max penalty, MMD, and the combination, respectively. The results are averaged over 5 seeds. $\hat{z} \in R^{128}$ and $\hat{z}_{\hat{\mathcal{S}}}, \hat{z}_{\hat{\mathcal{U}}} \in R^{64}$.

Digits Classification Accuracy				Colours $R^2_{\mathcal{U}}$			
$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 2$	$k = 4$	$k = 8$	$k = 16$
0.84	0.90	0.70	0.75	0.81	0.55	0.74	0.77
0.56±0.01	0.78±0.01	0.48±0.02	0.53±0.01	0.72±0.02	0.16±0.01	0.56±0.02	0.43±0.02
0.70±0.01	0.80±0.01	0.71±0.01	0.75±0.01	0.80±0.01	0.16±0.01	0.63±0.02	0.65±0.02
0.70±0.02	0.79±0.02	0.64±0.01	0.72±0.02	0.58±0.03	0.13±0.01	0.46±0.01	0.31±0.03