IDENTIFYING COARSE-GRAINED INDEPENDENT CAUSAL MECHANISMS WITH SELF-SUPERVISION

Anonymous authors
Paper under double-blind review

ABSTRACT

Current approaches for learning disentangled representations assume that independent latent variables generate the data through a single data generation process. In contrast, this manuscript considers independent causal mechanisms (ICM), which, unlike disentangled representations, directly model multiple data generation processes in a coarse granularity. In this work, we aim to learn a model that isolates each mechanism and approximates the ground-truth ICM from observational data. We outline sufficient conditions under which the ICM can be learned and isolated using a single self-supervised generative model with a mixture prior, simplifying previous methods. Moreover, we implement a generative model with an identifiable structural latent space by combining the ICM with a shared latent space. We compare this ICM approach to disentangled representations on various downstream tasks, showing that the ICM is more robust to intervention, co-variant shift, and noise due to the isolation between the data generation processes.

1 INTRODUCTION

The past decade witnessed the great success of machine learning (ML) algorithms, which achieve record-breaking performance in various tasks. However, most of the successes are based on discovering statistical regularities that are encoded in the data, instead of causal structure. As a consequence, standard ML model performance may decrease significantly under minor changes to the data, such as color changes that are irrelevant for the task, but which affect the statistical associations. On the other hand, human intelligence is more robust against such changes (Szegedy et al., 2013). For example, if a baby learns to recognize a digit, he/she can recognize the digit regardless of color, brightness, or even some style changes. Previous works (Parascandolo et al., 2018; Schölkopf, 2019) argue that it is because human intelligence relies on *causal mechanisms*, which represent the data generation processes and are invariant across domains. To this end, existing work has explored methods to learn domain invariant data generation processes, e.g., through disentangled representations

Disentangled representations, as its name suggests, assume that the data is generated using a set of independent latent explanatory factors and aim to disentangle the underlying factors in the data representation (Bengio et al., 2013). Previous works (Higgins et al., 2017; Locatello et al., 2020) have shown that such disentanglement is beneficial for downstream tasks and is more robust against the co-variant shift. Technically, disentangled representations assume that the high-dimensional observed data $\boldsymbol{x}$ is generated through an unknown function g, which takes a low-dimensional set of independent ground-truth latent factors $\boldsymbol{z}$ as input. The goal of learning disentangled representations is to recover the posterior $p(\boldsymbol{z} \mid \boldsymbol{x})$ i such a way that each dimension $\boldsymbol{z}$ encodes distinct variations. However, disentangled representations assume there exists only one data generation process, which may be at odds with real-world scenarios. For example, in a digit dataset, the data generation processes of digit 1 and 2 are different because digit 1 and 2 both have distinct variations (see Figure 2(a)). In Section 3.2, we show that the disentangled representations can not disentangle such class-dependent variations with the disentangled prior $p(\boldsymbol{z}) = \prod_{i=1}^{d} p(z_i)$.

The independent causal mechanisms (ICM) principle (Schölkopf, 2019) assumes that the generative processes of a data sample are composed of autonomous modules that are isolated from each other. Different from disentangled representations, the ICM principle assumes there may exist multiple data generation functions as autonomous modules. Previous works have attempted to learn the ICM

mainly from three aspects. Parascandolo et al. (2018) first try to learn an inverse version of the mechanisms which are shared across the dataset. More specifically, they proposed a competitive training-based mixture of expert model to learn the (inverse) mechanisms like the rotation of an object. Later, Locatello et al. (2018) integrated deep generative models into the competitive training framework and aimed to learn the causal mechanisms associated with each cluster in the dataset. In the rest of this paper, we refer to such mechanisms as ICM-conditioned mechanisms. In their algorithm, each cluster may represent one type of data such as digit 0. Recently, a similar method (von Kügelgen et al., 2020) was applied to the generative scene models, where a sequence of mechanisms was learned to draw independent objects in the scene. In this work, we focus on the first two aspects, where the i.i.d. data samples are generated cooperatively by ICM-conditioned mechanisms and shared mechanisms.

Despite the progress above, jointly learning of ICM-conditioned and shared mechanisms without supervision remains a difficult and unsolved task. For the shared causal mechanisms, Parascandolo et al. (2018) assumes that two types of data—the original data and the data that is transformed by the mechanisms—are available at the same time and are separable from each other. This assumption may not always hold as we do not have such separation in most of the real-world datasets. For the ICM conditioned mechanisms, competitive training does not provide guarantees that mechanisms are isolated from each other because it does not directly enforce the isolation in the training objective and relies on the collapse of the deep generative model, which is hard to control and is unstable.

We propose a new method to simultaneously learn the ICM-conditioned and the shared independent causal mechanisms from i.i.d. data. Unlike competitive training-based methods, we use a single generative model with a mixture prior to learn the mechanisms. Using the mixture prior, we define a structural latent space where each mechanism lies in orthogonal subspaces. We achieve the isolation between the ICM-conditioned mechanisms through self-supervision, which is based on ClusterGAN (Mukherjee et al., 2019). The isolation between ICM-conditioned mechanisms and the shared mechanism will be guaranteed by the structural latent space itself. Furthermore, we theoretically prove that the subspaces for each mechanism are identifiable. The key contributions of this paper are:

- we propose a simpler method to learn the ICM with only self-supervision,
- we implement a structural latent space that separates each ICM-conditioned mechanism and the shared mechanism,
- we prove the identifiability of the structural latent space,
- we develop a novel method to quantitatively evaluate the robustness of ML models under co-variant shift using the co-variant that is naturally encoded in the data.
- we conduct extensive experiments to show that the ICM is more robust against intervention, co-variant shift, and noise compared to disentangled representations.

2 Related Work

In addition to the ICM and the disentangled representations, which Section 1 introduces, learning the data generation process has also been studied from the following three aspects.

Functional Causal Model Functional Causal Model (FCM) is similar to the Bayesian network except for the way one expresses the dependency between variables. In the Bayesian network, we use the posterior distribution $P(X \mid \mathbf{PA}_X)$ to describe the dependency. In FCM, the relationships are expressed through deterministic functions, $f(X \mid \mathbf{PA}_X)$, and the uncertainty is introduced via the assumption that certain variables in the functions are not observed (Pearl et al., 2000). If each function in FCM represents an autonomous mechanism, such FCM is called a structural model. Moreover, if each mechanism only determines the value of one and only one variable, such a structural model is called a structural causal model (SCM). There are many works (Cai et al., 2019; Monti et al., 2020) that have studied the problem of discovering the SCM from observational data. Taking the view from the SCM's perspective, we want to approximately learn a causal model whose inputs are pure latent variables and whose output is a single high-dimensional variable that describes complex data such as images. Different from other SCM approaches, where the unobserved variables only add uncertainty into the model, the latent variables in our model carries distinct variations in the dataset.

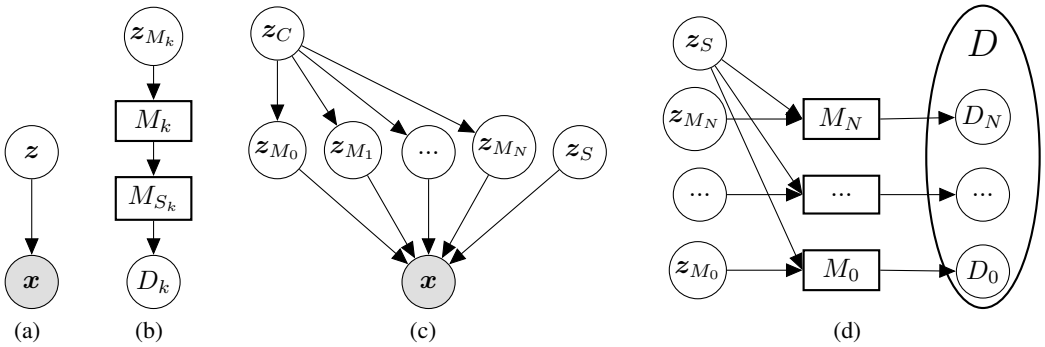


Figure 1: (a) The graphical model of deep generative models and disentangled representations. (b) The assumed data generation process in previous works (Parascandolo et al., 2018; Locatello et al., 2018). They assume the ICM-conditioned mechanism and the shared mechanism are applied sequentially. (c) The graphical model of our ICM approach, where $\boldsymbol{z_C}$ is a confounding variable that controls which ICM will take effect. (d) The assumed data generation processes in our approach. Each cluster D_k in the data set D is generated by different mechanisms, which simultaneously model the ICM-conditioned mechanism and the shared mechanism.

Independent Component Analysis Discovering independent components of the data generating process has been studied intensively (Hyvärinen & Oja, 2000; Hyvarinen et al., 2019). A recent work (Khemakhem et al., 2020) also bridges the gap between the independent component analysis (ICA) and the deep generative model. However, applying the ICA-based deep generative model to the complex dataset, even MNIST, is still difficult. This is because the current method requires a relatively large number of independent components (e.g. data from different classes, domains, etc) that exist in the dataset. For example, if the latent variable is Gaussian and the latent space has 10 dimensions, we would need 20 such components. Unfortunately, we only have 10 classes in the MNIST dataset.

3 Algorithms

3.1 Preliminaries

Many generative models (Kingma & Welling, 2013; Goodfellow et al., 2014; Locatello et al., 2020) assume that the data $\boldsymbol{x}$ is generated through a two-step procedure: (1) A sample $\boldsymbol{z}$ is drawn from an unobserved continuous prior distribution $p(\boldsymbol{z})$, which is usually assumed to be $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. (2) An observational data sample $\boldsymbol{x}$ is drawn from an unknown conditional distribution $p(\boldsymbol{x} \mid \boldsymbol{z})$. Figure 1(a) and 1(b) further visualize this procedure.

Generative Adversarial Network Generative Adversarial Network (GAN) learns the data generation process via a two-player minmax game between a generator G, which models $p(\boldsymbol{x} \mid \boldsymbol{z})$, and a discriminator D: $\min_G \max_D V(G, D) = \mathbb{E}_{\boldsymbol{x} \sim p(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]$. The goal is to minimize the divergence between the generated data and the real data. Recent research has shown that the Wasserstain distance (Arjovsky et al., 2017) is a good choice of divergence in practice. Thus, we use Wasserstain GAN with gradient penalty (WGAN-GP) (Gulrajani et al., 2017) as our generative model. GAN are less popular in unsupervised learning of disentangled representations as is difficult to approximate the posterior $p(\boldsymbol{z} \mid \boldsymbol{x})$. However, previous work (Locatello et al., 2018) showed its advantages in learning ICMs. We will again show its advantage in our method from another perspective.

3.2 The Mixture Prior and Structural Latent Space

Unlike most generative models which assume $p(\boldsymbol{z}) = \prod_{i=1}^{d} p(z_i)$, we use a mixture prior:

$$p(\boldsymbol{z}) = p(\boldsymbol{z}_S) \sum_{k=1}^{N} p(\boldsymbol{z}_{M_k}) \tag{1}$$

where $p(\boldsymbol{z}_S) = \prod_{i=1}^{d_S} p(z_{S_i})$ and $p(\boldsymbol{z}_{M_k}) = \prod_{i=1}^{d_{M_k}} p(z_{M_{k_i}})$. Figure 1(c) and 1(d) further visualize the graphical model and the assumed data generation process with the mixture prior. In this prior, $\boldsymbol{z}_S$ represents the input for the shared mechanisms and $\boldsymbol{z}_{M_k}$ represents the inputs for ICM-conditioned mechanisms. We encode all the shared mechanisms, such as rotation and brightness, in $\boldsymbol{z}_S$ and do not further separate them. Although we do not achieve isolation between each shared mechanism, it usually does not hurt the performance because most of the predictive tasks are associated with the ICM-conditioned mechanisms. Thus, we only need to isolate the shared mechanisms from ICM-conditioned mechanisms.

In addition, we pose an "isolation" constraint for the ICM-conditioned mechanisms such that $\boldsymbol{z}_{M_j} = \mathbf{0}$ if $\boldsymbol{z}_{M_i} \neq \mathbf{0}$ for all $i \neq j$. We pose this constraint because we know that if a data sample is generated by certain modules, it has nothing to do with the other modules. We note that NVAE (Antoran & Miguel, 2019) also used the same constraint in computing the posterior. Using this prior with the "isolation" constraint would allow us to approximate both the ICM-conditioned mechanisms and the shared mechanisms using a single generative model, which we will discuss in Section 3.3.

If we use one generative model to represent one ICM as the mixture-of-experts model (Locatello et al., 2018) does, it is easy to see that a single disentangled model can not learn all the ICM. However, as we only use a single generative model to approximate all the ICM, whether a disentangled model has such ability still worth discussion. To see this, consider that by definition, each z_i in $\boldsymbol{z}$, which admits a density $p(\boldsymbol{z}) = \prod_{i=1}^{d} p(z_i)$, represents one type of variations in disentangled representations and one causal mechanism represents one type of data generation processes. If we use each dimension or subspace in $\boldsymbol{z}$ to represent each causal mechanism, we can generate data through g by setting each z_i to an arbitrary value. Then, there exist $\boldsymbol{z} \in \mathcal{Z}$ such that $z_i \neq 0$ for all i. It further means that there exist data samples that are generated by multiple ICM-conditioned mechanisms. This contradicts the ICM principle that each ICM-conditioned mechanism represents a different data generation process. Technically, we will have $g^*(\boldsymbol{z}) \notin \mathcal{X}$ as the ground-truth generative model g^* only takes the $\boldsymbol{z}$ that satisfies the "isolation" constraint as input.

3.3 The Self-supervised Model

As is discussed in Section 3.1, we use a single WGAN-GP with the mixture prior to learn the ICM. Fortunately, using a mixture distribution as the prior has already been explored by ClusterGAN (Mukherjee et al., 2019). We extend their method by replacing the continuous part of their discrete-continuous prior, which includes a categorical variable and an isotropic Gaussian, by our mixture prior. The self-supervision is achieved by predicting which mechanism does the generated data come from using the encoder E. Such self-supervision would encourage the data samples from the same ICM-conditioned mechanism to be similar to each other and vice versa. Thus, we can isolate the mechanisms in the sense that their sources are predictable. The loss function of our method is:

$$\begin{aligned}\mathcal{L} = W(\mathbb{P}_g||\mathbb{P}_r) + \beta_C \mathbb{E}_{\boldsymbol{z}\sim P(\boldsymbol{z})}\mathcal{H}(\boldsymbol{z}_C, E(G(\boldsymbol{z}))_C) + \beta_M \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}||\boldsymbol{z}_M - E(G(\boldsymbol{z}))_M||_2^2 \\ + \beta_S \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}||\boldsymbol{z}_S - E(G(\boldsymbol{z}))_S||_2^2 + \beta_r \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}||\boldsymbol{x} - G(E(\boldsymbol{x}))||_2^2\end{aligned} \tag{2}$$

where $\boldsymbol{z} = [\boldsymbol{z}_C, \boldsymbol{z}_M, \boldsymbol{z}_S]$, $\boldsymbol{z}_C$ is a categorical variable represents the index of the mechanisms, $\boldsymbol{z}_M = [\boldsymbol{z}_{M_0}, ..., \boldsymbol{z}_{M_N}]$, $\mathbb{P}_g$ denotes the distribution covered by the generator network G, $\mathbb{P}_r$ denotes the real data distribution, $W(\mathbb{P}_g||\mathbb{P}_r)$ denotes the Wasserstain distance between two distributions, $\mathcal{H}$ denotes the cross-entropy loss, and E denotes an encoder network that inverts G. We note that we have tried to replace the Euclidean distance with Cosine distance in the loss function, but both methods yield similar results.

We use $\boldsymbol{z}_C$ to add ICM-conditioned bias terms to the generator's first activation layer, which eliminates the ambiguity when $\boldsymbol{z}_M$ is close to $\mathbf{0}$ (Antoran & Miguel, 2019). Moreover, we want to encourage the isolation between each ICM-conditioned mechanism through self-supervision, which is expressed through $\mathcal{H}(\boldsymbol{z}_C, E(G(\boldsymbol{z}))_C)$. As the categorical variable only serves as the bias term and the self-supervision label, we leave it as an implementation detail and keep the mixture prior unchanged. For the interested reader, we show that our method can implicitly achieve the isolation between ICM-conditioned mechanisms and the shared mechanism by mapping them to disjoint manifolds in Theorem 3, Section 4

Using this loss function above, we want our generative model to achieve three objectives: 1) Approximating the real data distribution using the generator 2) Making the data generated by different ICM-conditioned mechanisms separable. 3) Constructing a bijective mapping between the latent space $\mathcal{Z}$ and the data space $\mathcal{X}$ using the generator G and the encoder E. It is easy to see the first objectives in the loss function. For the third objective, the latter three terms in Equation 2 can be seen as the cycle-consistent loss (Zhu et al., 2017), which encourages the generative model $G : \mathcal{Z} \to \mathcal{X}$ to be bijective. A little bit different from the CycleGAN (Zhu et al., 2017), we only use the bijection property to enforce the generated data to encode the information of the corresponding $\boldsymbol{z}$ and make sure we can use $E = G^{-1}$ in the downstream predictive tasks. Besides, our encoder directly inverts the generator. Thus it better fits the definition of FCM as it directly outputs a value, instead of estimating a posterior distribution like the encoder in the variational auto-encoder (VAE).

4 IDENTIFIABILITY

A model is identifiable if it is theoretically possible to learn the ground-truth model with an infinite number of observations. In some cases, we can only identify a subset of model parameters. We say such a model is partially-identifiable. In this section, we show the ICM is partial-identifiable in the sense that we can learn the mechanisms and isolate each mechanism in orthogonal subspaces. Different from identifying disentangled representations (Locatello et al., 2020), we focus on the subspace level identifiability instead of the dimension level identifiability. Before showing our result, we first introduce the impossible result of identifying the deep latent variable model without supervision and inductive bias.

Theorem 1. *(Locatello et al., 2019) For $d > 1$, let $\boldsymbol{z} \sim P$ denote any distribution which admits a density $p(\boldsymbol{z}) = \prod_{i=1}^{d} p(z_i)$. Then, there exists an infinite family of bijective functions $h : \mathrm{supp}(\boldsymbol{z}) \to \mathrm{supp}(\boldsymbol{z})$ such that $\frac{\partial h_i(\boldsymbol{u})}{\partial u_j} \neq 0$ almost everywhere for all i and j (i.e., $\boldsymbol{z}$ and $h(\boldsymbol{z})$ are completely entangled) and $P(\boldsymbol{z} \leq \boldsymbol{u}) = P(h(\boldsymbol{z}) \leq \boldsymbol{u})$ for all $\boldsymbol{u} \in \mathrm{supp}(\boldsymbol{z})$ (i.e., they have the same marginal distribution).*

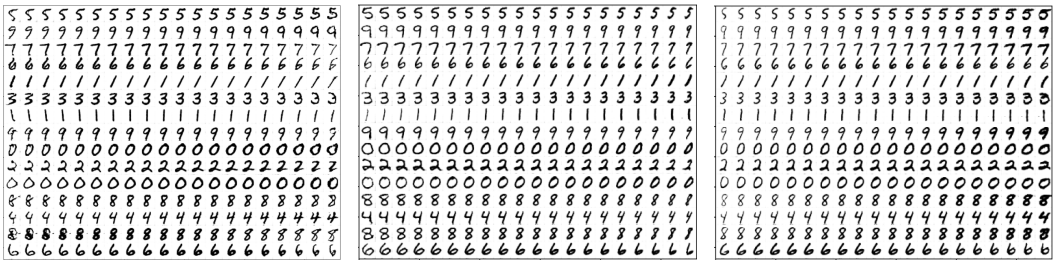
The theorem above indicates that even if you can learn a generative model g that maps the latent space $\mathcal{Z}$ to $\mathcal{X}$ in the same way as the ground-truth disentangled g^*, you can not tell whether the learned model g is the ground-truth disentangled g^* or an entangled $g^* \circ h$ using only the observational data. The following lemma, whose proof is in Appendix A, further shows that using $g = g^* \circ h$ is sufficient to represent all the possible generative model g we can learn.

Lemma 2. *Let $\mathcal{G}$ be the space of smooth invertable functions with smooth inverse (i.e., a diffeomorphism) that map $\mathcal{Z}$ to $\mathcal{X}$, and $h : \mathcal{Z} \to \mathcal{Z}$ is a smooth invertable function. Then, any function $g \in \mathcal{G}$ can be represented as $g = g^* \circ h$, where $g^* : \mathcal{Z} \to \mathcal{X}$ is the ground-truth generative model in the function space $\mathcal{G}$*

Together with the lemma above, the following theorem shows that even if we can not tell whether the learned generative model $g = g^* \circ h$ is the ground truth generative model g^* or not, the function h will not entangle two mechanisms. To this end, we employ the constraint which is enforced by the estimation, that if $\boldsymbol{z}_{M_i} \neq \mathbf{0}$, $\boldsymbol{z}_{M_j} = \mathbf{0}$ for all $i \neq j$.

Theorem 3. *For $N > 1$, let $\boldsymbol{z} \sim P$ denote the mixture prior which admits a density $p(\boldsymbol{z}) = p(\boldsymbol{z}_S) \sum_{k=1}^{N} p(\boldsymbol{z}_{M_k})$. Moreover, we assume that if $\boldsymbol{z}_{M_i} \neq \mathbf{0}$, $\boldsymbol{z}_{M_j} = \mathbf{0}$ for all $i \neq j$. We use $\mathcal{M}$ to denote the manifold where the ground-truth latent variable $\boldsymbol{z}$ lies on. We assume that each ICM-conditioned mechanisms M_k is associated with $\boldsymbol{z}_{M_k}$, which lies on $\mathcal{M}_{M_k}$ and the shared mechanisms S is associated with $\boldsymbol{z}_S$, which lies on $\mathcal{M}_S$. Then, if there exists an smooth invertable function $h : \mathcal{Z} \to \mathcal{Z}$ such that $g = g^* \circ h$ maps $\mathcal{Z}$ to $\mathcal{X}$, we have h maps each $\mathcal{M}_{M_k}$ to disjoint sub-manifold $\hat{\mathcal{M}}_{M_k}$ and maps $\mathcal{M}_S$ to $\hat{\mathcal{M}}_S$, which is disjoint from all $\hat{\mathcal{M}}_{M_k}$.*

We leave the proof in Appendix B and provide two examples for ICM-conditioned mechanisms in this paragraph. Assume we have two ICM-conditioned mechanisms M_i and M_j and $h : \mathcal{Z} \to \mathcal{Z}$ is an transformation that entangles M_i and M_j. We further assume $\boldsymbol{z}_{M_i} \in \mathbb{R}^2$ and $\boldsymbol{z}_{M_j} \in \mathbb{R}^2$. Then, we can construct a $\boldsymbol{z}$ where $\boldsymbol{z}_{M_i} \neq \mathbf{0}$ and $\boldsymbol{z}_{M_j} = \mathbf{0}$ but $h(\boldsymbol{z})_{M_i} \neq \mathbf{0}$ and $h(\boldsymbol{z})_{M_j} \neq \mathbf{0}$. For example, we can let $[\boldsymbol{z}_{M_i}, \boldsymbol{z}_{M_j}] = [1.0, 0.0, 0.0, 0.0]$ and $h([\boldsymbol{z}_{M_i}, \boldsymbol{z}_{M_j}]) = [0.7, 0.0, 0.5, 0.0]$. We can also let $[\boldsymbol{z}_{M_i}, \boldsymbol{z}_{M_j}] = [1.0, 1.0, 0.0, 0.0]$ and $h([\boldsymbol{z}_{M_i}, \boldsymbol{z}_{M_j}]) = [0.0, 1.0, 1.0, 0.0]$. The existence of such



(a) ICM-conditioned Subspace, Distinct Variation for Each Row

(b) Shared Subspace, Dim #0, Width and Rotation

(c) Shared Subspace, Dim #1, Stroke Thickness and Width

Figure 2: Latent Space Traversal of ICM model on MNIST

$\boldsymbol{z} \in \mathcal{Z}$, which let $h(\boldsymbol{z}) \notin \mathcal{Z}$, contradicts the assumption of $h : \mathcal{Z} \to \mathcal{Z}$. Thus, h needs to map each mechanism to disjoint manifold to avoid violating the assumption.

5 Experiment

In this section, we first examine the robustness of our model against interventions. As we only apply interventions on the root node, such interventions are equivalent to conditional generation, which can be visualized by a latent space traversal. We then evaluate the robustness of our model under the co-variant shift by measuring the downstream performance. The co-variate shift is the change in the distribution of the co-variates, that is, the independent variables. More specifically, we consider disentangled single co-variant shifts (e.g. rotation) and entangled multiple co-variant shifts (e.g. brightness and width). Unlike previous works (Arjovsky et al., 2019; Locatello et al., 2020), which rely on manual annotation, we propose a new method to extract the latent co-variants from the datasets and use them to create experiments. Finally, we measure the robustness of our model against uniform noise and Gaussian noise by again measuring the downstream performance. Through our experiments, we use 1) Two datasets, which are MNIST and FashionMNIST. 2) Three competitors, which are VAE (Kingma & Welling, 2013), β-VAE (Higgins et al., 2017), and Ada-GVAE (Locatello et al., 2020). 3) Four co-variant shifts, which are implicitly carried by the datasets.

5.1 Interventional Robustness and Latent Space Traversal

We show the latent space traversal of our ICM model on MNIST in Figure 2. In the ICM model, each ICM-conditioned subspace has one dimension and the shared subspace has four dimensions. For the ICM-conditioned subspace traversal, we set the $\boldsymbol{z}_C$ to the corresponding index and manually change z_{M_k}. During this process, all the irrelevant dimensions are set to 0. As Figure 2(a) shows, each row represents a traversal of an ICM-conditioned mechanism. For the shared mechanism, we first assign an index 0 to $\boldsymbol{z}_C$. Then, we do a traversal for a shared dimension. After this traversal, we set $\boldsymbol{z}_C$ to the next index and repeat the procedure. In Figures 2(b) and 2(c), each figure represents a traversal for a shared dimension and each row represents a traversal of a shared dimension with a fixed $\boldsymbol{z}_C$. Due to the limited space, we leave more visualization results in the Appendix.

We say our ICM model is robust against intervention in the sense that no matter how we change $\boldsymbol{z}_{M_k}$ and $\boldsymbol{z}_S$ while keeping $\boldsymbol{z}_C$ and $\boldsymbol{z}_{\setminus M_k}$ fixed, the generator does not generate data that does not belong to mechanism M_k (e.g. digit 0 does not change to other digits during the traversal). Furthermore, we show that the traversal of each ICM-conditioned mechanism yields a distinct type of variation that is associated with the mechanism, and the traversal of the shared mechanism yields the same type of variation. Compared with conditional generative models (Kingma et al., 2014; Klys et al., 2018), our work do not pose any requirement on the labelling.

Finally, we note that the number of ICM-conditioned mechanisms is chosen to be larger than the number of classes, which means one ground-truth mechanism may be split into two estimated mechanisms. Previous works (Parascandolo et al., 2018; Locatello et al., 2018) also adopt the same setting and we do not find it triggers any issue in the following experiment.

Table 1: Average Accuracy of Downstream Classifiers under Different Shift Strength

MODEL	MNIST (T)	MNIST (W&R)	MNIST (R)	FASHIONMNIST (D&W)
ICM	**74.63%**	**56.91%**	43.71%	**42.92%**
VAE	55.91%	41.92%	39.90%	37.08%
β-VAE	73.34%	46.33%	46.05%	41.60%
ADA-GVAE	68.09%	47.16%	**47.58%**	40.85%

Table 2: Normalized Accuracy Variations of Downstream Classifiers under Different Shift Strength

MODEL	MNIST (T)	MNIST (W&R)	MNIST (R)	FASHIONMNIST (D&W)
ICM	**2.43%**	**3.70%**	**7.10%**	**5.02%**
VAE	4.32%	6.30%	7.33%	9.02%
β-VAE	2.71%	4.98%	7.80%	7.83%
ADA-GVAE	2.96%	4.90%	7.31%	7.98%

Table 3: Shift Distance Needed for 10% Relative Accuracy Drop under Co-variant Shift

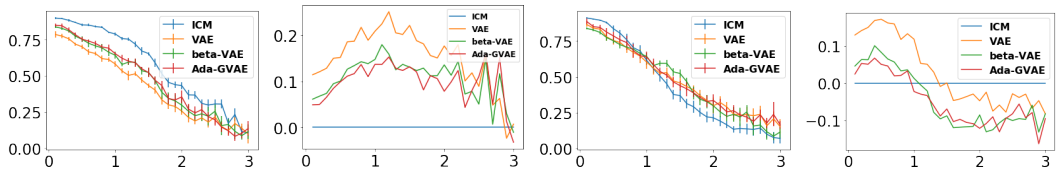
MODEL	MNIST (T)	MNIST (W&R)	MNIST (R)	FASHIONMNIST (D&W)
ICM	**1.4**	**0.8**	**0.6**	**0.4**
VAE	0.6	0.4	0.4	0.2
β-VAE	1.1	0.5	0.4	0.3
ADA-GVAE	1.1	0.5	0.4	0.3

5.2 ROBUSTNESS UNDER CO-VARIANT SHIFT

To quantitatively measure the robustness under the co-variant shift, we first partition the dataset using the learned representations as the co-variant. For convince, we use the encoder of the NVAE model (Antoran & Miguel, 2019) to do that. But using the shared mechanism in our ICM model would yield the same result. More specifically, we partition the dataset into subsets $\{\boldsymbol{x} \mid \boldsymbol{x} \in \mathcal{X}, \hat{\boldsymbol{z}}_i = E(\boldsymbol{x})_i \in [C_{lower}, C_{upper})\}$ using its inferred value in dimension i. For the training set, we set $[C_{lower}, C_{upper})$ to $[0, \infty)$. For each test set, we set $[C_{lower}, C_{upper})$ to $[-0.1, 0)$, $[-0.2, -0.1)$, ..., $[-3.0, -2.9)$. We use the $C_{lower}^{train} - C_{upper}^{test}$ to represent the strength of the co-variant shift, which is also called the shift distance in Table 3. In this experiment, we consider stroke thickness (T), width and rotation (W&R), rotation (R) and, darkness and width (D&W) as co-variant. Figure 4 in the Appendix further visualizes these co-variant.

Then, we use a gradient boosting classifier (GBT) from Scikit-learn with default parameters as the downstream ML model, which is the same as previous works (Locatello et al., 2019; 2020). We train the classifier with 1000 samples, which is sufficient for producing good accuracy, and test its accuracy on a sequence of test sets. For each experiment, we evaluate the classifier 10 times and collect the average accuracy as well as the accuracy variations. Tables 1 and 2 show the average accuracy and the normalized variation of the classifier using each model's representation as input under different co-variant shift, respectively. Our ICM model achieves the best average accuracy across all the experiments except for MNIST (R). For the normalized accuracy variations, which is the standard deviation over the accuracy, our ICM model always achieves the lowest variation. It means that the ICM model is less sensitive to the choice of training samples as the GBT classifier does not contribute many variations.

We further investigate the issue of our model under MNIST (R) co-variant shift. Figure 3 shows the accuracy changes as the shift strength increases. In both experiments, our method shows more robustness when the shift strength is low as its accuracy decreases slower and its advantage grows bigger. However, after a threshold, our method begins to lose its advantage. There are two possible reasons: 1) The test data shifts too far from the training data and the base generative model can not generalize to the test sets. After the test set shifts too far away, none of the methods perform



(a) Accuracy under Width&Rotation Shift (b) Accuracy Differences between ICM and Others (c) Accuracy under Rotation Shift (d) Accuracy Differences between ICM and Others

Figure 3: Accuracy and Accuracy Differences of Downstream Classifiers under Co-variant Shift on MNIST dataset. If ICM outperforms another method, the accuracy difference is positive.

Table 4: Accuracy of Downstream Classifiers under Noise $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$

MODEL	MNIST (T)	MNIST (W&R)	MNIST (R)	FASHIONMNIST (D&W)
ICM	**52.64% ± 2.03%**	39.36% ± 2.58%	**53.30% ± 2.08%**	**46.69% ± 1.75%**
VAE	34.13% ± 1.48%	32.40% ± 2.07%	27.14% ± 2.32%	33.19% ± 2.52%
β-VAE	39.98% ± 2.74%	39.93% ± 2.23%	37.35% ± 3.16%	34.79% ± 2.86%
ADA-GVAE	46.41% ± 1.66%	**43.10% ± 0.86%**	46.25% ± 0.77%	43.92% ± 1.61%

Table 5: Accuracy of Downstream Classifiers under Noise $\epsilon * \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ on MNIST (W&R)

MODEL	$\epsilon = 0.25$	$\epsilon = 0.5$	$\epsilon = 0.75$
ICM	**85.04% ± 0.57%**	**69.59% ± 2.05%**	51.61% ± 4.68%
VAE	74.97% ± 0.66%	58.57% ± 1.44%	42.90% ± 1.42%
β-VAE	80.08% ± 0.77%	64.94% ± 2.53%	49.51% ± 2.02%
ADA-GVAE	81.64% ± 1.29%	68.38% ± 1.76%	**53.90% ± 1.68%**

well. It's hard to conclude that a model with $\sim 40\%$ accuracy is better than a model with $\sim 35\%$ accuracy. 2) The test set there contains too few samples, which is just around tens or hundreds, and makes the evaluation inaccurate. As we can see from Figure 3, the larger the shift, the bigger the accuracy variations. To eliminate this interference, we instead measure how much distance the co-variant shift needs to go to decrease the accuracy by a percentage relatively. Such evaluation will put more weight on the test sets which have more samples and yield reasonable accuracy. Table 3 and Tables 9, 10 in Appendix D.2show our method can tolerate more co-variant shift before the accuracy relatively drops by 10%, 20%, and 40%.

5.3 ROBUSTNESS AGAINST NOISE

We evaluate the robustness of our model under noise using a similar setting as the previous section. In this experiment, we use the whole train set as the test set and train the GBT with 1000 samples. Table 5.3 shows our ICM model is generally more robust against Gaussian noise $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Table 5.3 further shows that although there are methods that perform slightly better than our method on MNIST (W&R), our method still performs better before the noise becomes too large and decrease the accuracy by a high percentage. We further show the results under uniform noise $\mathrm{Uniform}(\mathbf{0}, \boldsymbol{I})$ in Appendix D.3.

6 CONCLUSION

In this paper, we present a self-supervised method to learn the independent causal mechanisms. We show the presents of ICM-conditioned mechanisms and shared mechanisms. Our model can learn these two types of mechanisms through a structural latent space. Furthermore, we outline the sufficient conditions for theoretically identifying the mechanisms from observational data. Experiments show that our ICM model is generally more robust against interventions, co-variant shifts, and noise.

REFERENCES

Javier Antoran and Antonio Miguel. Disentangling and learning robust representations with natural clustering. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pp. 694–699. IEEE, 2019.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

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

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Ruichu Cai, Feng Xie, Clark Glymour, Zhifeng Hao, and Kun Zhang. Triad constraints for learning causal structure of latent variables. In *Advances in Neural Information Processing Systems*, pp. 12883–12892, 2019.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pp. 2672–2680. Curran Associates, Inc., 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pp. 5767–5777, 2017.

I. Higgins, Loïc Matthey, A. Pal, C. Burgess, Xavier Glorot, M. Botvinick, S. Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430, 2000.

Aapo Hyvarinen, Hiroaki Sasaki, and Richard Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 859–868, 2019.

Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pp. 2207–2217, 2020.

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

Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pp. 3581–3589, 2014.

Jack Klys, Jake Snell, and Richard Zemel. Learning latent subspaces in variational autoencoders. In *Advances in Neural Information Processing Systems*, pp. 6444–6454, 2018.

Francesco Locatello, Damien Vincent, Ilya Tolstikhin, Gunnar Rätsch, Sylvain Gelly, and Bernhard Schölkopf. Competitive training of mixtures of independent deep generative models. *arXiv preprint arXiv:1804.11130*, 2018.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, pp. 4114–4124, 2019.

Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. *International Conference on Machine Learning*, 2020.

Ricardo Pio Monti, Kun Zhang, and Aapo Hyvärinen. Causal discovery with general non-linear relationships using non-linear ica. In *Uncertainty in Artificial Intelligence*, pp. 186–195. PMLR, 2020.

Sudipto Mukherjee, Himanshu Asnani, Eugene Lin, and Sreeram Kannan. Clustergan: Latent space clustering in generative adversarial networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4610–4617, 2019.

Giambattista Parascandolo, Niki Kilbertus, Mateo Rojas-Carulla, and Bernhard Schölkopf. Learning independent causal mechanisms. In *International Conference on Machine Learning*, pp. 4036–4044. PMLR, 2018.

Judea Pearl et al. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 2000.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pp. 2234–2242. Curran Associates, Inc., 2016.

Bernhard Schölkopf. Causality for machine learning. *arXiv preprint arXiv:1911.10500*, 2019.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Julius von Kügelgen, Ivan Ustyuzhaninov, Peter Gehler, Matthias Bethge, and Bernhard Schölkopf. Towards causal generative scene models via competition of experts. *arXiv preprint arXiv:2004.12906*, 2020.

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