Provably Learning Disentangled & Compositional Models

Prédoc 3 Examination

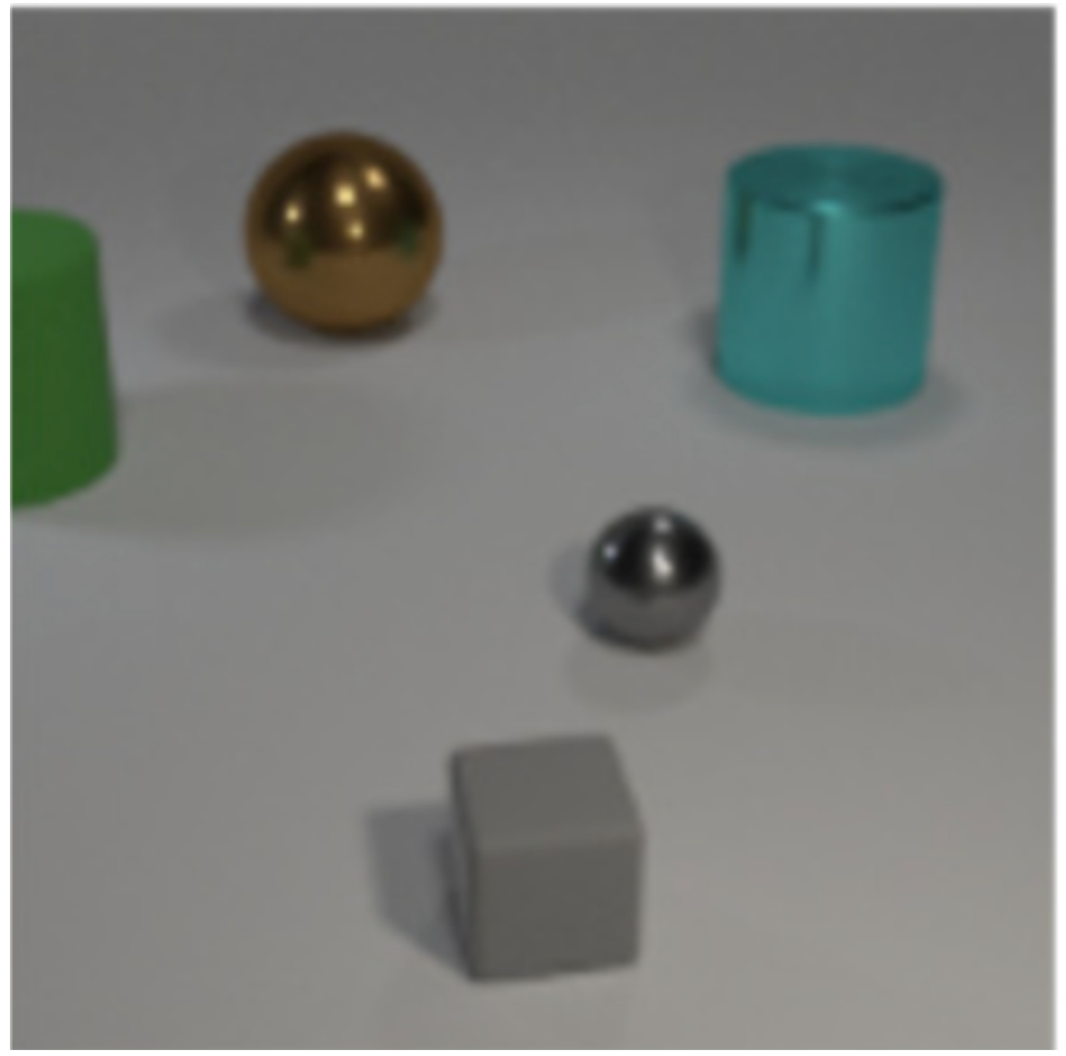
Student: Divyat Mahajan

Committee: Ioannis Mitliagkas, Simon Lacoste-Julien, Yoshua Bengio

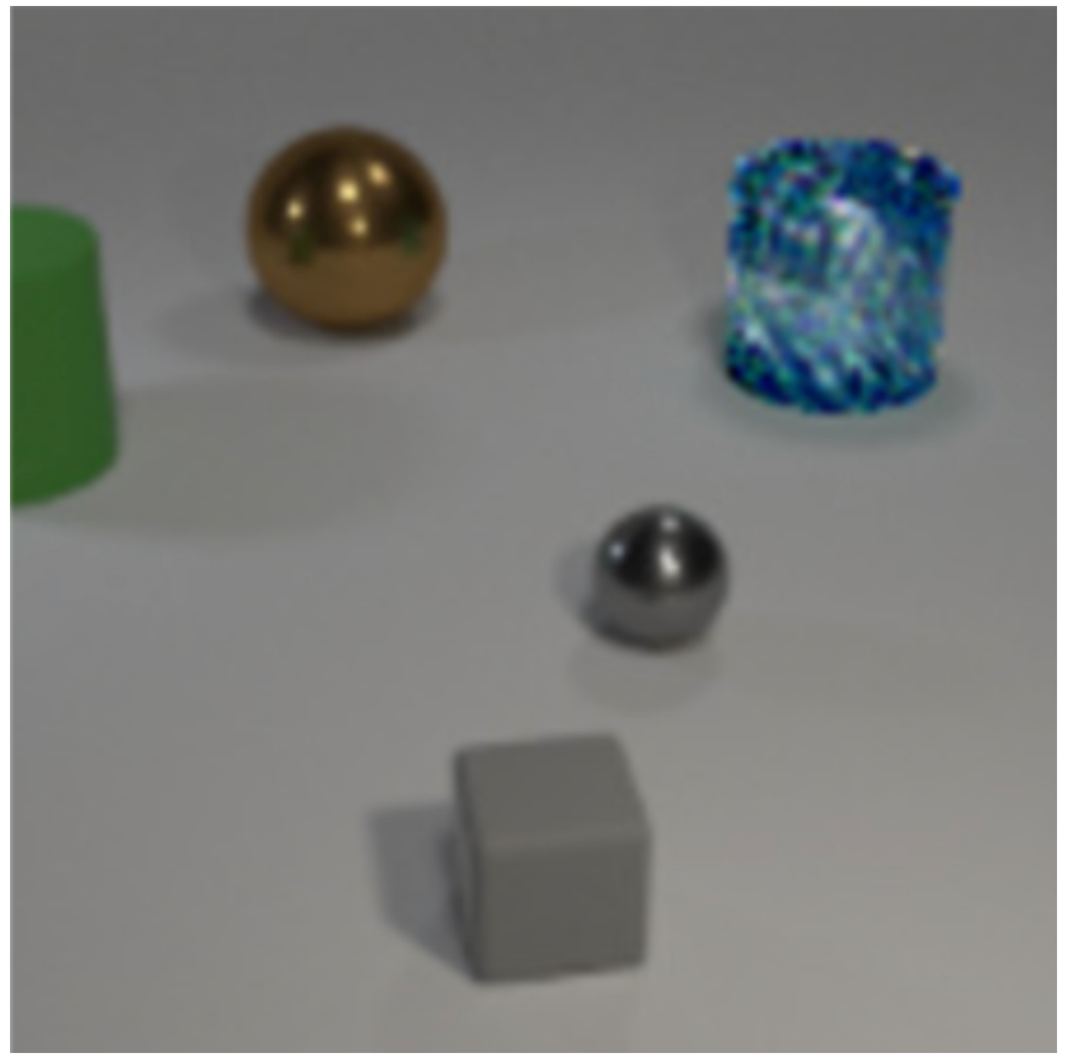
Université de Montréal

Covariate Shift

Training Samples



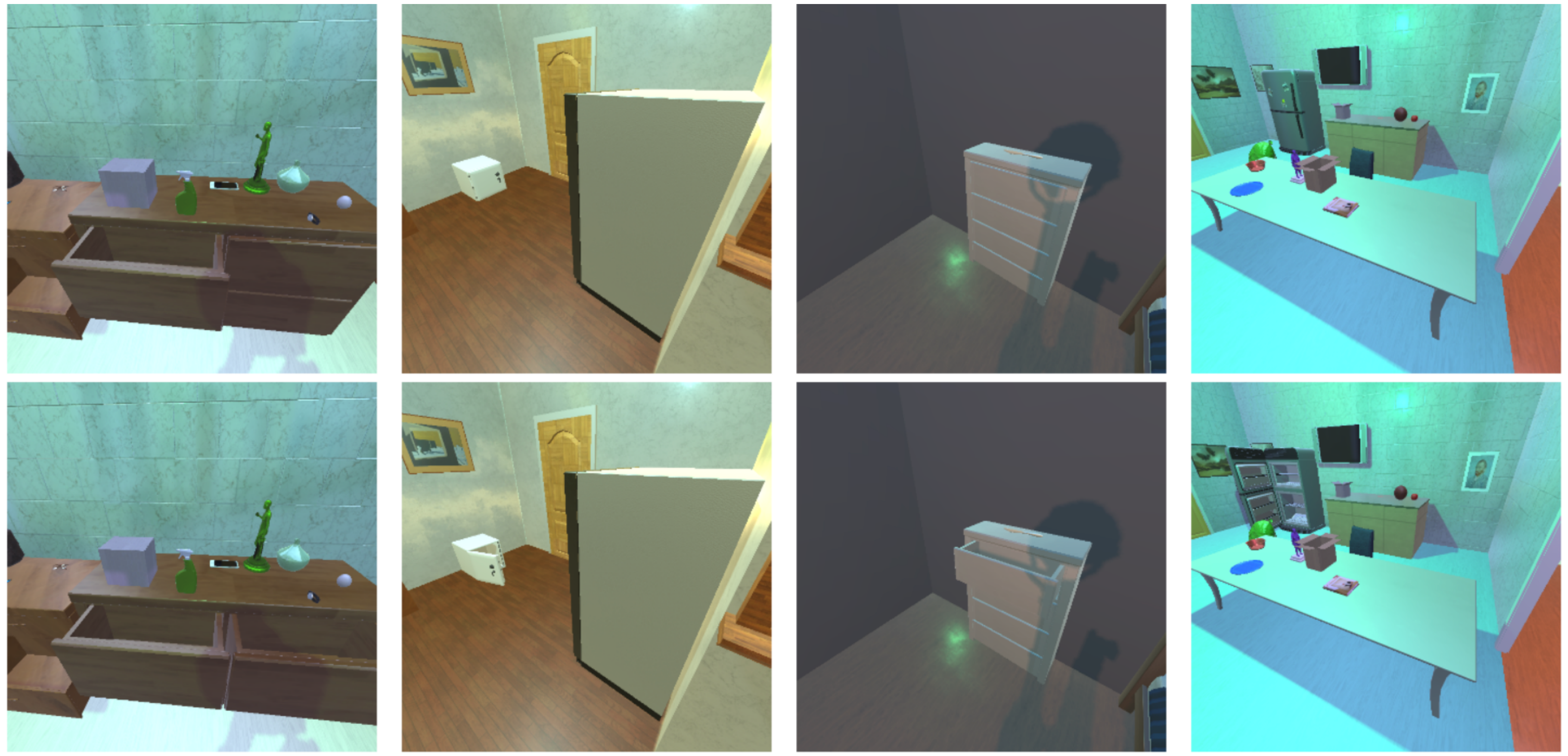
Test Samples

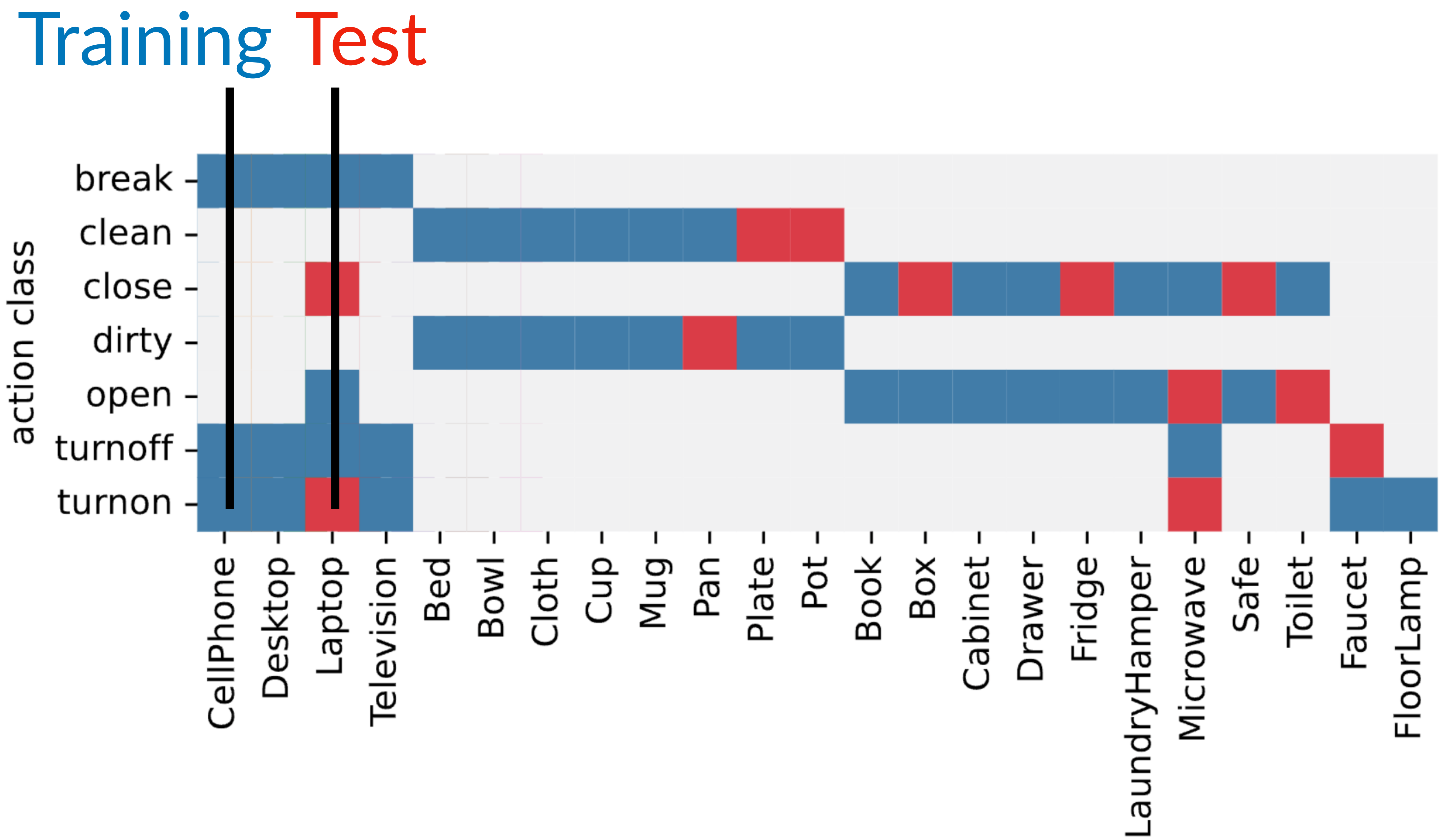


Source: Generalization and Robustness Implications in Object-Centric Learning by Dittadi et al. (2022)

Hypothesis: Learning disentangled representations can allow us to efficiently adapt to covariate shifts as it changes mechanisms in a sparse manner

Compositional Shift

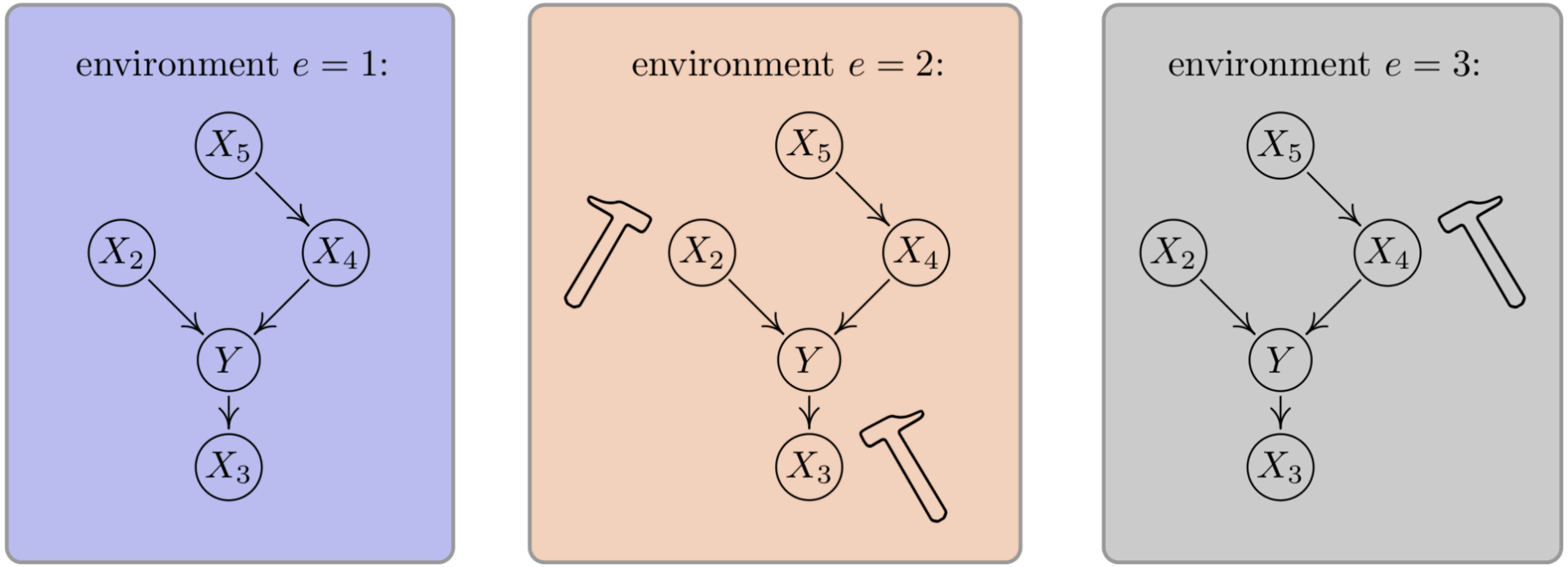




Source: Causal Triplet by Liu et al. (2023)

Hypothesis: Learning disentangled representations can allow us to efficiently extrapolate to novel compositions

Distribution Shifts in SCMs

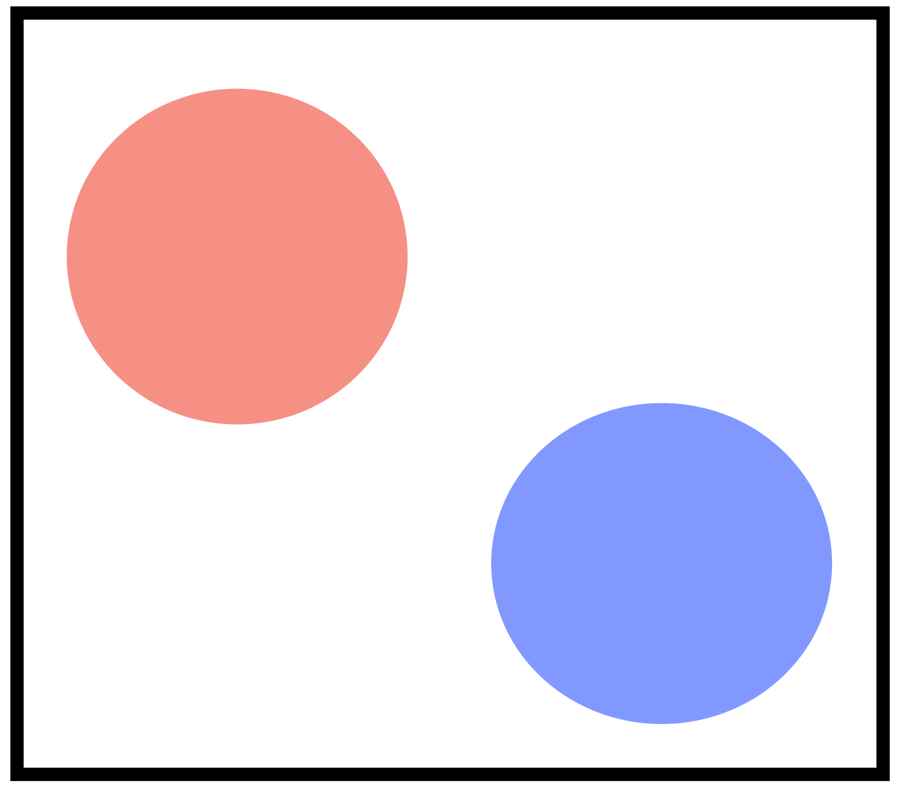


Source: Causal Inference using Invariant Prediction by Peters et al. (2015)

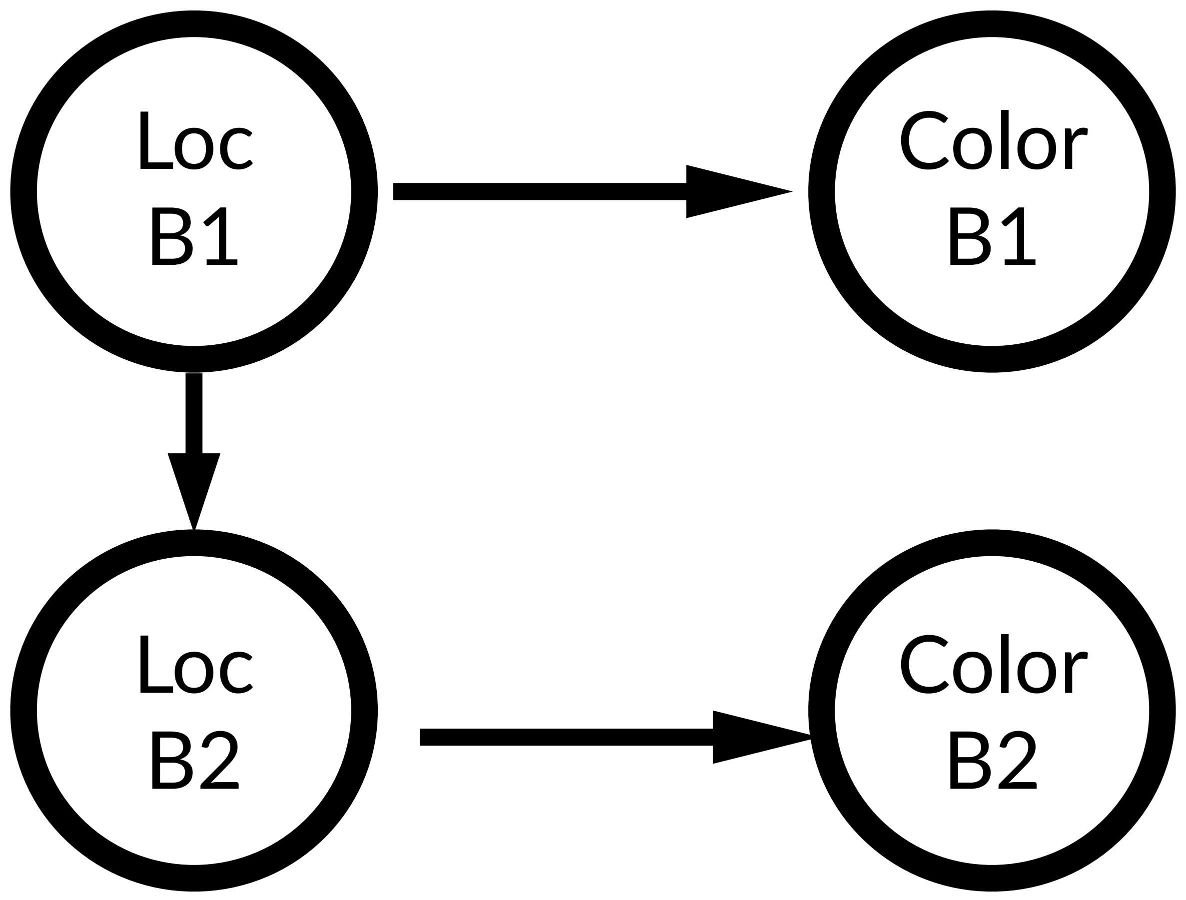
- **Independent Causal Mechanisms (ICM)**: Changing one causal mechanism leads to no change in the other causal mechanisms
- **Sparse Mechanism Shift**: Effect of interventions is modular in structural causal models

Disentangled Representation Learning

Input

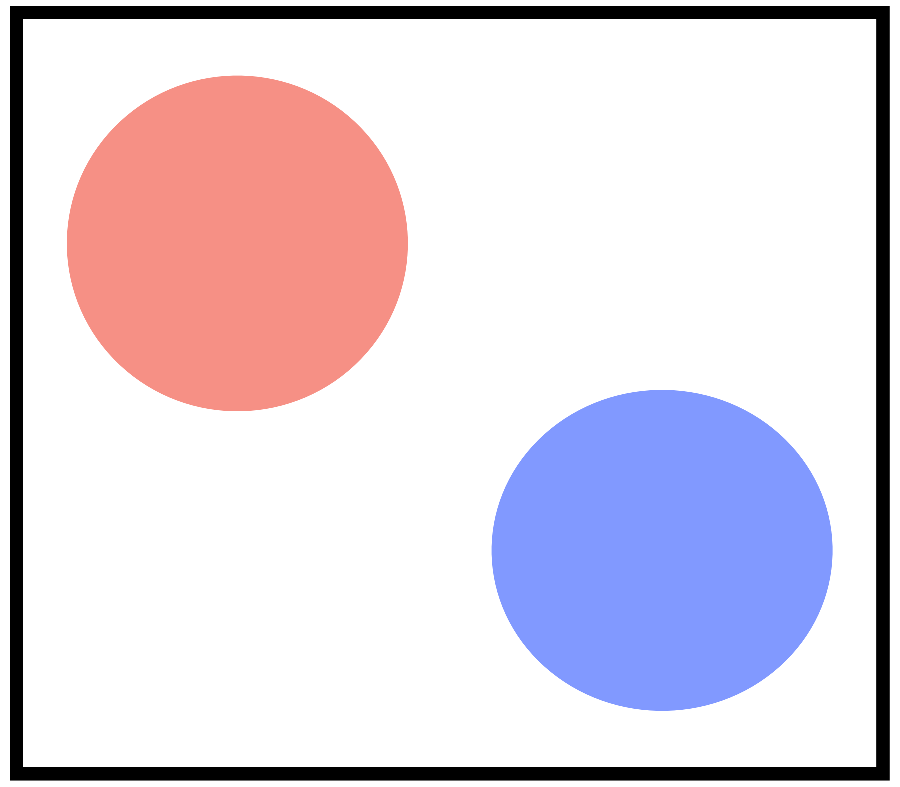


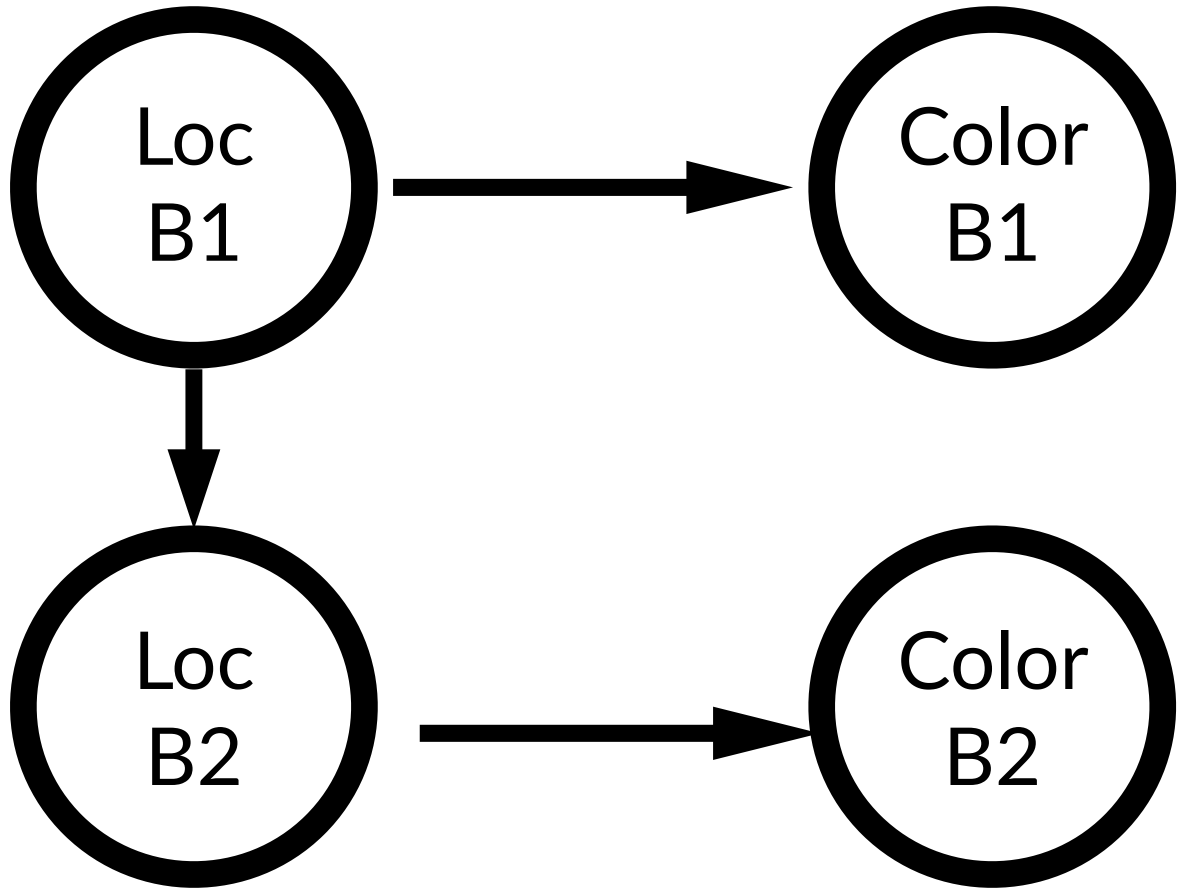
Latent Factors of Variations



- **Setup:** $x = g(z)$ where $z \in \mathbb{R}^{d_z}$ are the latent (causal) factors of the data generation process (DGP), that are transformed to observations $x \in \mathbb{R}^{d_x}$
- **Goal**: Invert the DGP to get latent factors (z) from observations (x)

Indeterminacy in Latent Recovery



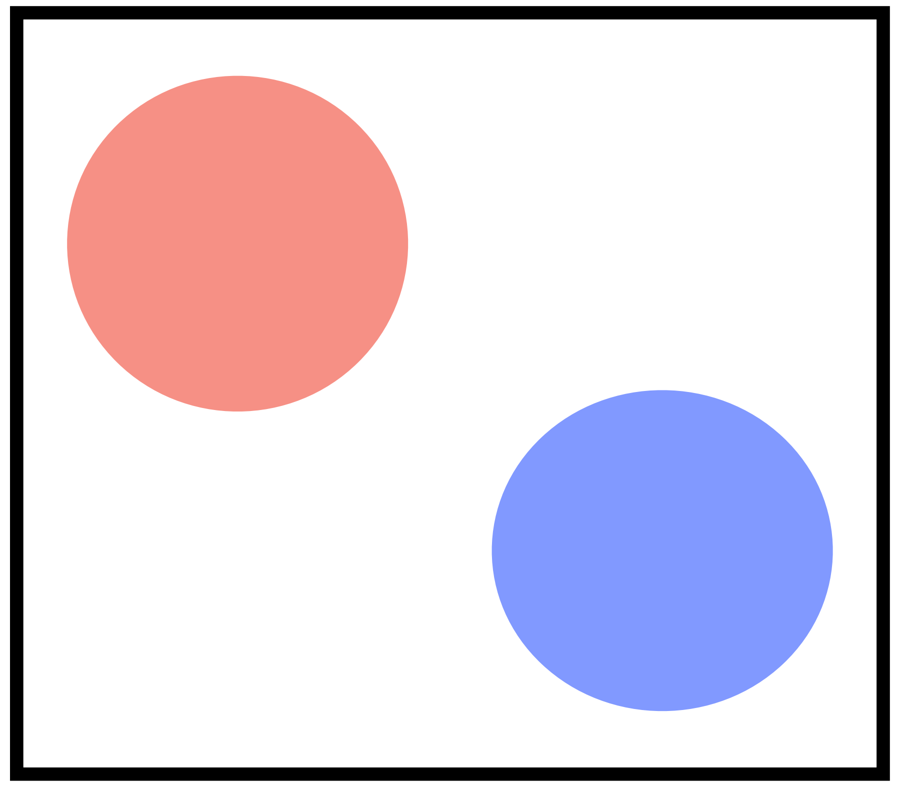


- **Reconstruction Objective:** Optimal encoder ($\hat{f} : \mathbb{R}^{d_x} \to \mathbb{R}^{d_z}$) and the optimal decoder ($\hat{g} : \mathbb{R}^{d_z} \to \mathbb{R}^{d_x}$) satisfy the following.

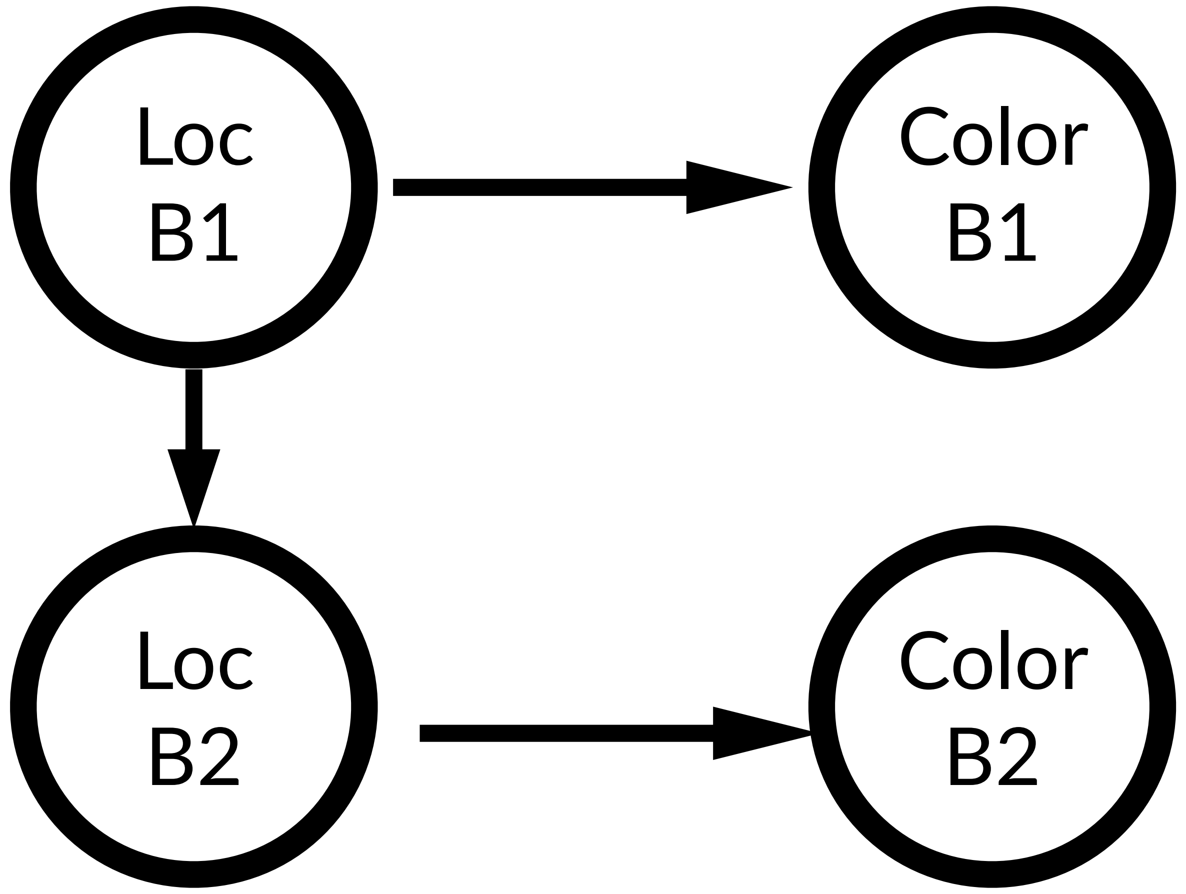
$$\mathbb{E}_{x \sim X} || x - \hat{g}(\hat{f}(x)) ||^2 = 0 \quad \implies \quad \hat{z} = v(z) \; ; v(z) = \hat{g}^{-1} \circ g(z)$$

Indeterminacy in Latent Recovery

Input



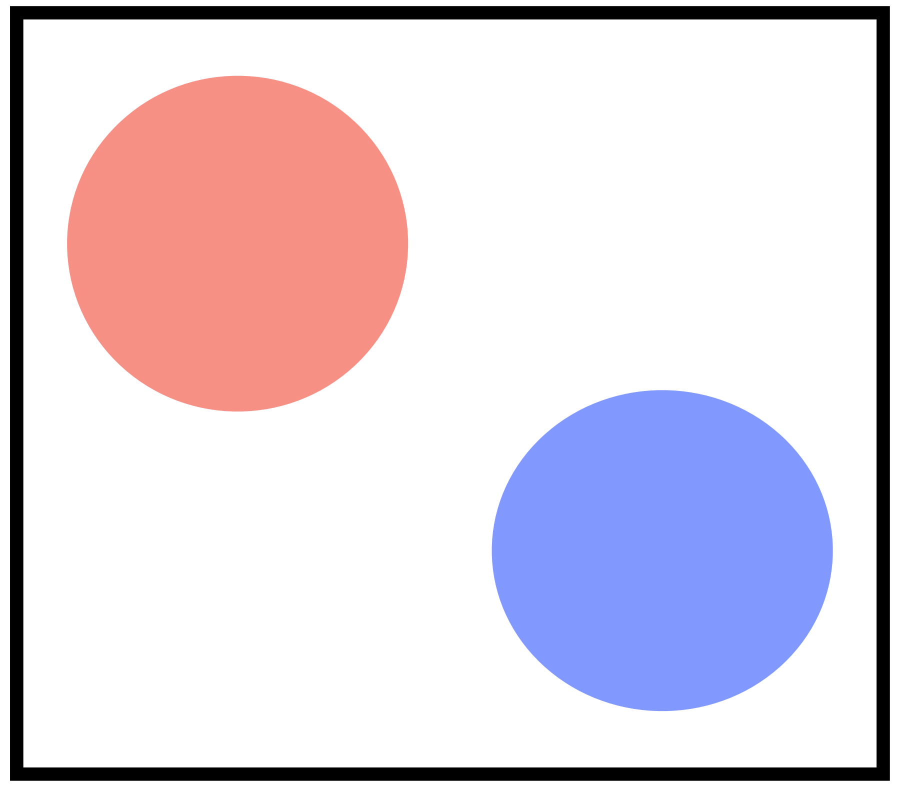
Latent Factors of Variations



- We need to constrain the indeterminacy in latent recovery; $\hat{z} = v(z) = \hat{g}^{-1} \circ g(z)$
- Both $(\hat{f}, \hat{g})$ and (f, g) explain data equally well, however, learned latents $\hat{z}$ might be a complex transformation of true latents z

Latent Identification

Input



Latent Factors of Variations

True Encoder: $f(x)$

$z = [Loc_{B_1}, Col_{B_1}, Loc_{B_2}, Col_{B_2}]$

Learned Encoder: $\hat{f}(x)$

$\hat{z} = [Loc_{B_2}, Col_{B_1}, Col_{B_2}, Loc_{B_1}]$

- **Permutation & Scaling Identification:** $\hat{z} = \Pi \circ \Lambda \, z + b$ where Π is permutation matrix and Λ is invertible diagonal matrix

- **Local Disentanglement**: $\hat{z} = v(z)$ where Jacobian of v is permuted diagonal matrix

How to achieve identification guarantees?

$$\hat{z} = v(z) = \hat{g}^{-1} \circ g(z)$$

- Constraints on the mixing function (g) and learned decoder ($\hat{g}$)
- Constrains on the latent distribution ($\mathbb{P}(Z)$) and enforcing learned latents ($\hat{z} = v(z)$) to satisfy them as well

How to achieve identification guarantees?

$$\hat{z} = v(z) = \hat{g}^{-1} \circ g(z)$$

- **Linear ICA:**
  - Constrain g, $\hat{g}$ to be linear functions
    - Leads to linear identification as $v(z) = \hat{g}^{-1} \circ g(z)$ is a linear function
  - Constrain z, $\hat{z}$ to have mutually independent components and *all components of z are non-gaussian*
    - Further restricts the linear $v(z)$ to permutation & scaling matrix.

Solving Non-linear ICA

Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations

Francesco Locatello [1 2] **Stefan Bauer** [2] **Mario Lucic** [3] **Gunnar Rätsch** [1] **Sylvain Gelly** [3] **Bernhard Schölkopf** [2]
Olivier Bachem [3]

Unlike linear ICA, restricting $Z, \hat{Z}$ to have mutually independent components is **not sufficient** to guarantee disentanglement for non-linear ICA!

Today's Talk

Disentanglement with Auxiliary Information

Towards Efficient Representation Identification in Supervised Learning

Kartik Ahuja*, **Divyat Mahajan***, Vasilis Syrgkanis, Ioannis Mitliagkas

Conference on Causal Learning and Reasoning [CleaR 2022]

Unsupervised Disentanglement & Cartesian-Product Extrapolation

Additive Decoders for Latent Variables Identification and Extrapolation

Sébastien Lachapelle*, **Divyat Mahajan***, Ioannis Mitliagkas, Simon Lacoste-Julien

Advances in Neural Information Processing Systems [NeurIPS 2023 (Oral)]

Extrapolation with Discrete Factors

Compositional Generalization with Additive Energy Models

Ongoing work in collaboration with Kartik Ahuja, Ioannis Mitliagkas, Mohammad Pezeshki, Pascal Vincent

Other Contributions

Causal Inference with Observational Data

Empirical Analysis of Model Selection for Heterogeneous Causal Effect Estimation

Divyat Mahajan, Ioannis Mitliagkas, Brady Neal, Vasilis Syrgkanis

International Conference on Learning Representations [ICLR 2024 (Spotlight)]

Disentanglement with Interventional Data

Interventional Causal Representation Learning

Kartik Ahuja, **Divyat Mahajan**, Yixin Wang, Yoshua Bengio

International Conference on Machine Learning [ICML 2023 (Oral)]

Benefits of Disentanglement for Downstream Tasks

Synergies between Disentanglement and Sparsity in Multi-Task Learning

Sébastien Lachapelle*, Tristan Deleu*, **Divyat Mahajan**, Ioannis Mitliagkas, Yoshua Bengio, Simon Lacoste-Julien, Quentin Bertrand

International Conference on Machine Learning [ICML 2023]

Towards Efficient Representation Identification in Supervised Learning

Kartik Ahuja*, Divyat Mahajan*, Vasilis Syrgkanis & Ioannis Mitliagkas

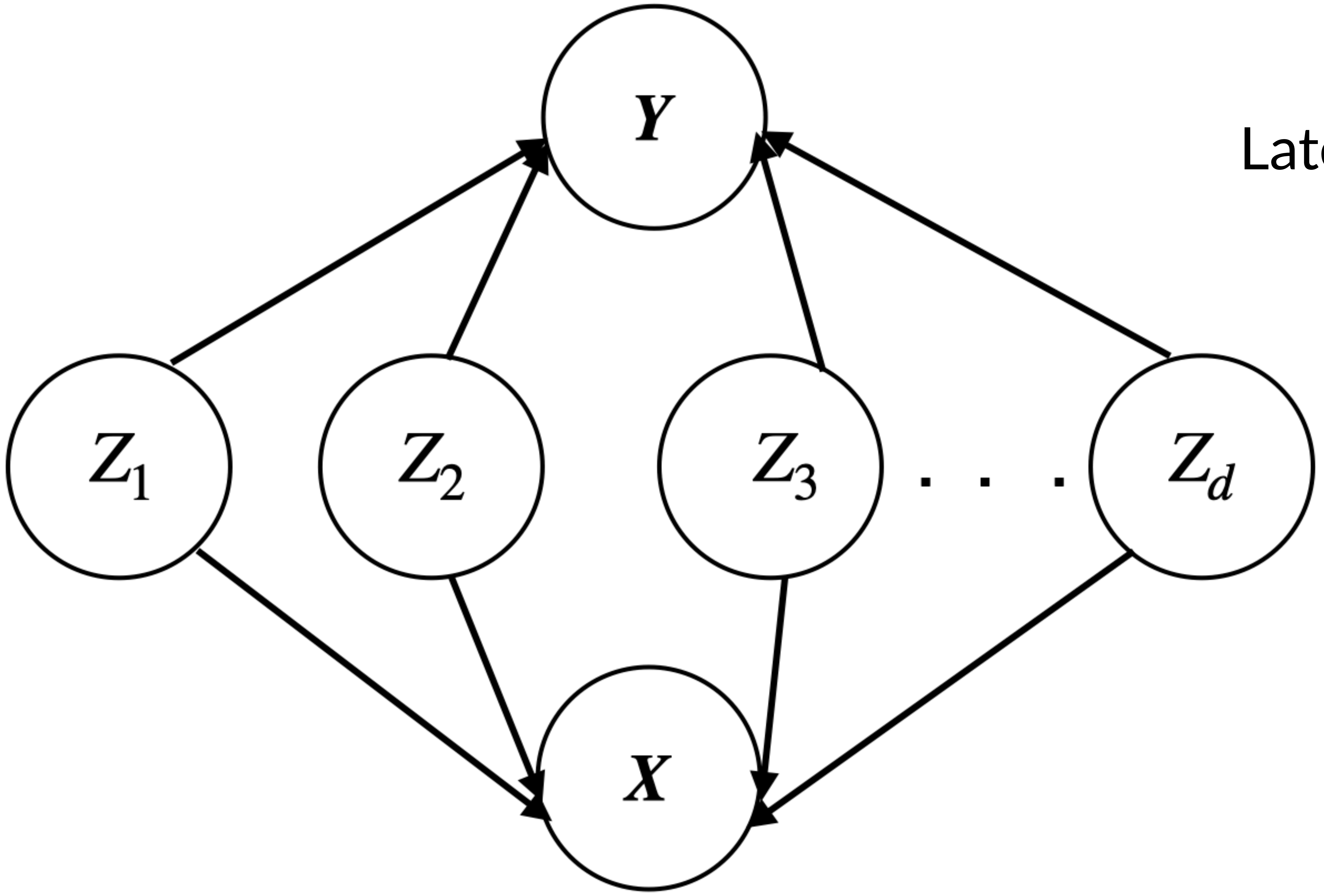
Conference on Causal Learning and Reasoning (CL*ea*R) 2022

*Equal contribution

Mila

Université de Montréal

Non Linear ICA with Auxiliary Information



$$Z = (Z_1, \cdots, Z_d)$$

Latent Variable: Mutually independent & Non-Gaussian

$$Y \leftarrow \Gamma Z + N$$

Auxiliary Information: $Y \in \mathbf{R}^k$ & $Z \perp N$

$$X \leftarrow g(Z)$$

Observed non-linear mixing of latents, g is bijection

Independence Constrained ERM

Model: $W \circ \Phi$

$W \in \mathbb{R}^{d \times k}$: Linear Classifier
$\Phi \in \mathscr{H}_{\Phi}$: Non-Linear Representation

Empirical Risk Minimization (ERM): $$\min_{W \in \mathbb{R}^{d \times k}, \Phi \in \mathscr{H}_{\Phi}} \sum_{i=1}^{N} \ell\Big(W \circ \Phi(x_i), y_i\Big)$$

IC-ERM: $\min_{W \in \mathbb{R}^{d \times k}, \Phi} \sum_{i=1}^{N} \ell\Big(W \circ \Phi(x_i), y_i\Big)$ **s.t. Components of $\Phi(x)$ are i.i.d.**

Identification with IC-ERM

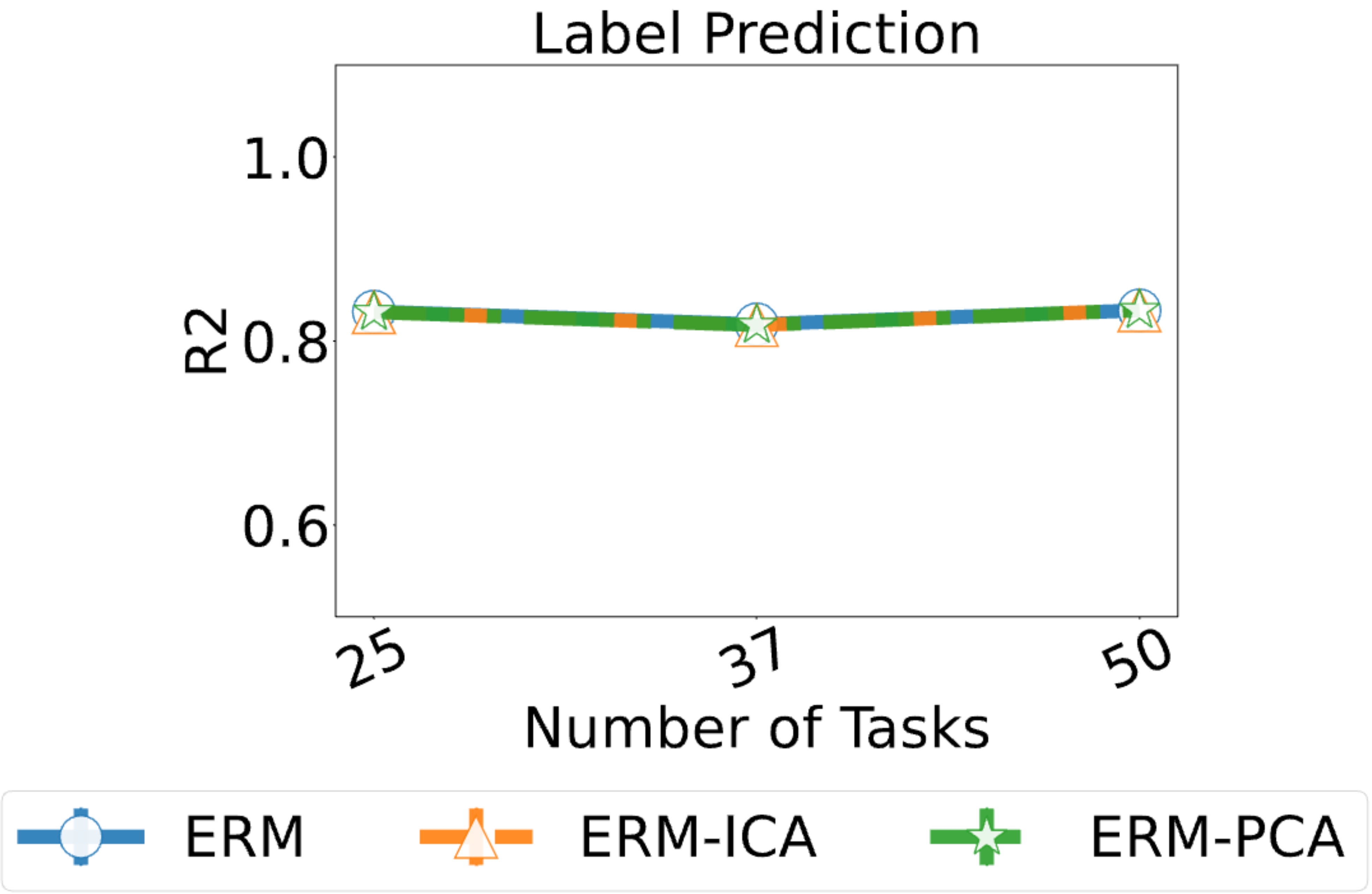
Assumption: Dimension of the **label (k)** is equal to the dimension of the **latent (d)**

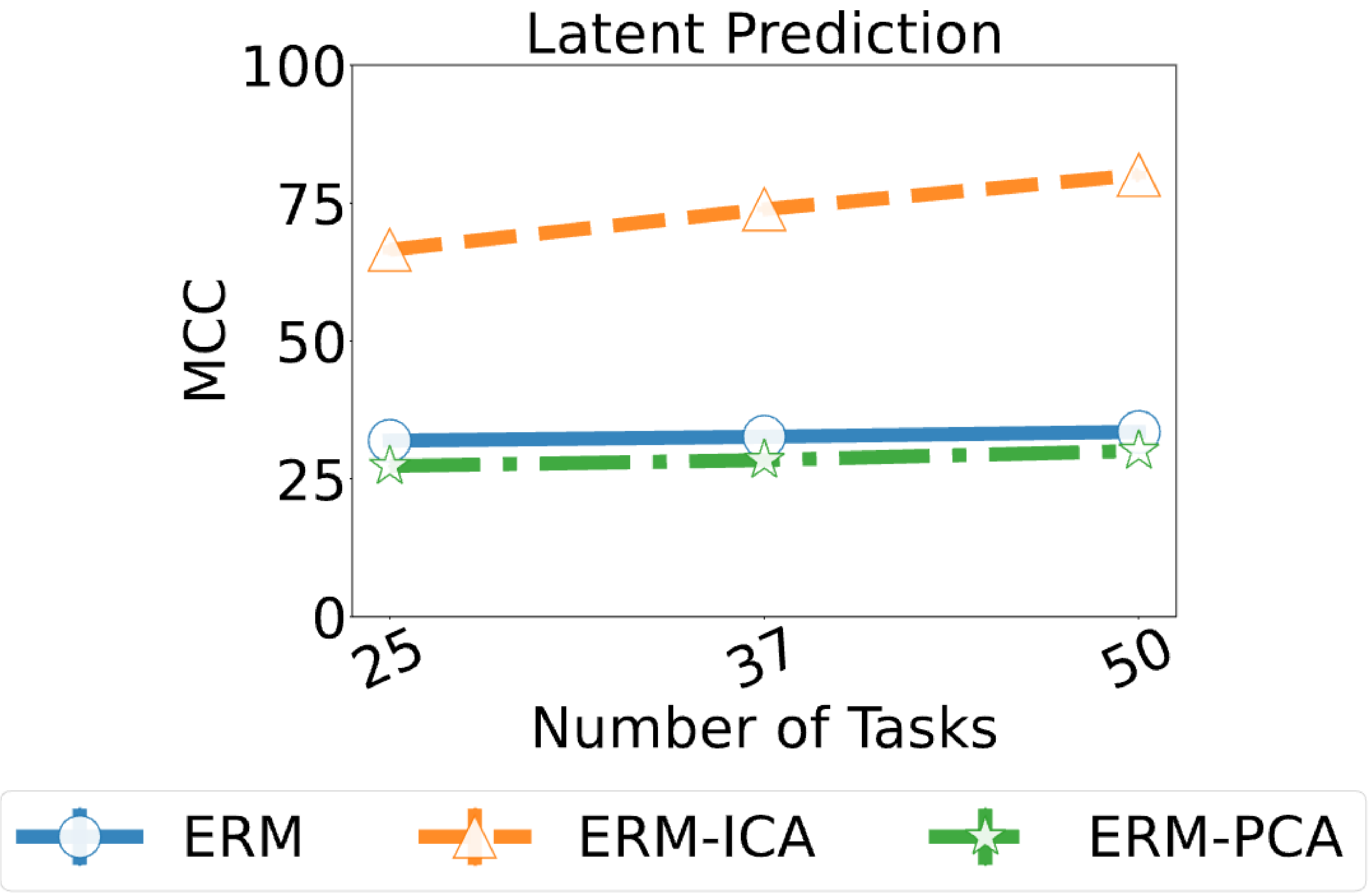
Theorem (Informal): Under the above **assumption** as well as those on the data generation process (**mutual independence** of Z), we have the following:

- **ERM**: Optimal solutions identify true latents up to **linear transformation**
- **IC-ERM:** Optimal solutions identify true latents up to **permutation & scaling**

Note: We also present identification results in the paper when $k < d$

Experiments





Results for regression task with latent dimension $d = 50$.

Disentanglement performance (MCC) improves as we observe more tasks.

Extending theory beyond mutual independence

Assumption: Latent Variables Z are mutually independent

Solution: Assume Γ to be sparse where $Y \leftarrow \Gamma Z + N$

Synergies between Disentanglement & Sparsity: Generalization & Identifiability in Multi-Task Learning

Sébastien Lachapelle*, Tristan Deleu*, Divyat Mahajan, Ioannis Mitliagkas, Yoshua Bengio, Simon Lacoste-Julien & Quentin Bertrand

*Equal contribution

International Conference on Machine Learning (*ICML*) 2023

Can we identify latents without auxiliary information?

Additive Decoders for Latent Variables Identification and Cartesian-Product Extrapolation

Sébastien Lachapelle*, Divyat Mahajan*, Ioannis Mitliagkas & Simon Lacoste-Julien

Neural Information Processing Systems (*NeurIPS*) 2023 (*Oral*)

*Equal contribution

Mila

Université de Montréal

Samsung Advanced Institute of Technology AI Lab Montreal

Additive Decoders

$$x = g(z) = \sum_{B \in \mathcal{B}} g^{(B)}(z_B)$$

Observation
e.g. an image

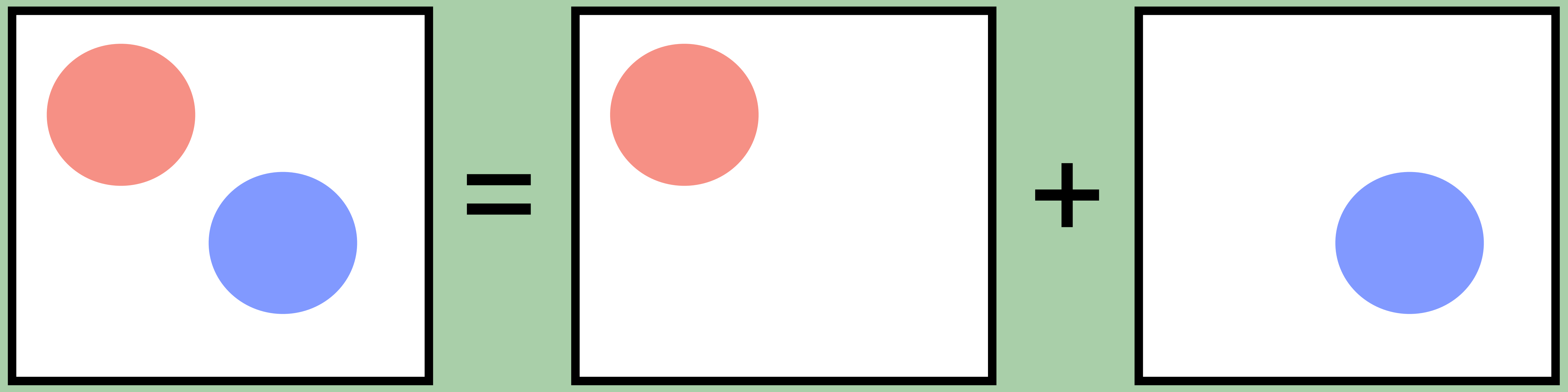
Latent
Factors

Partition of $\{1, \ldots, d_z\}$
e.g. $\mathcal{B} = \{\{1,2\}, \{3,4\}\}$

Sub-blocks of z

Example: Images of moving balls

$x = g(z)$ $\quad$ $g^{(B_1)}(z_1)$ $\quad$ $g^{(B_2)}(z_2)$



$\mathcal{B} = \{\{1,2\}, \{3,4\}\}$

$z_{B_1} = (z_1, z_2)$ Coordinates of

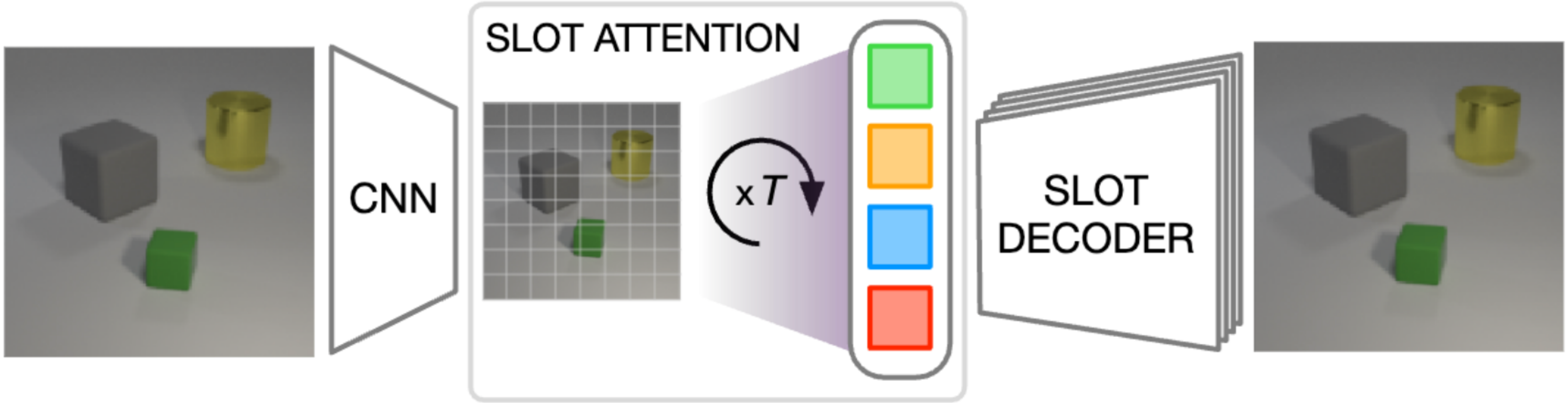
$z_{B_2} = (z_3, z_4)$ Coordinates of

$g^{(B)}$ Block-specific Decoder

Contribution

We introduce **additive decoders**: a simple architecture similar to object-centric decoders for which we can prove both **disentanglement** and **extrapolation** guarantees.

Decoder Architecture in Object-Centric Learning



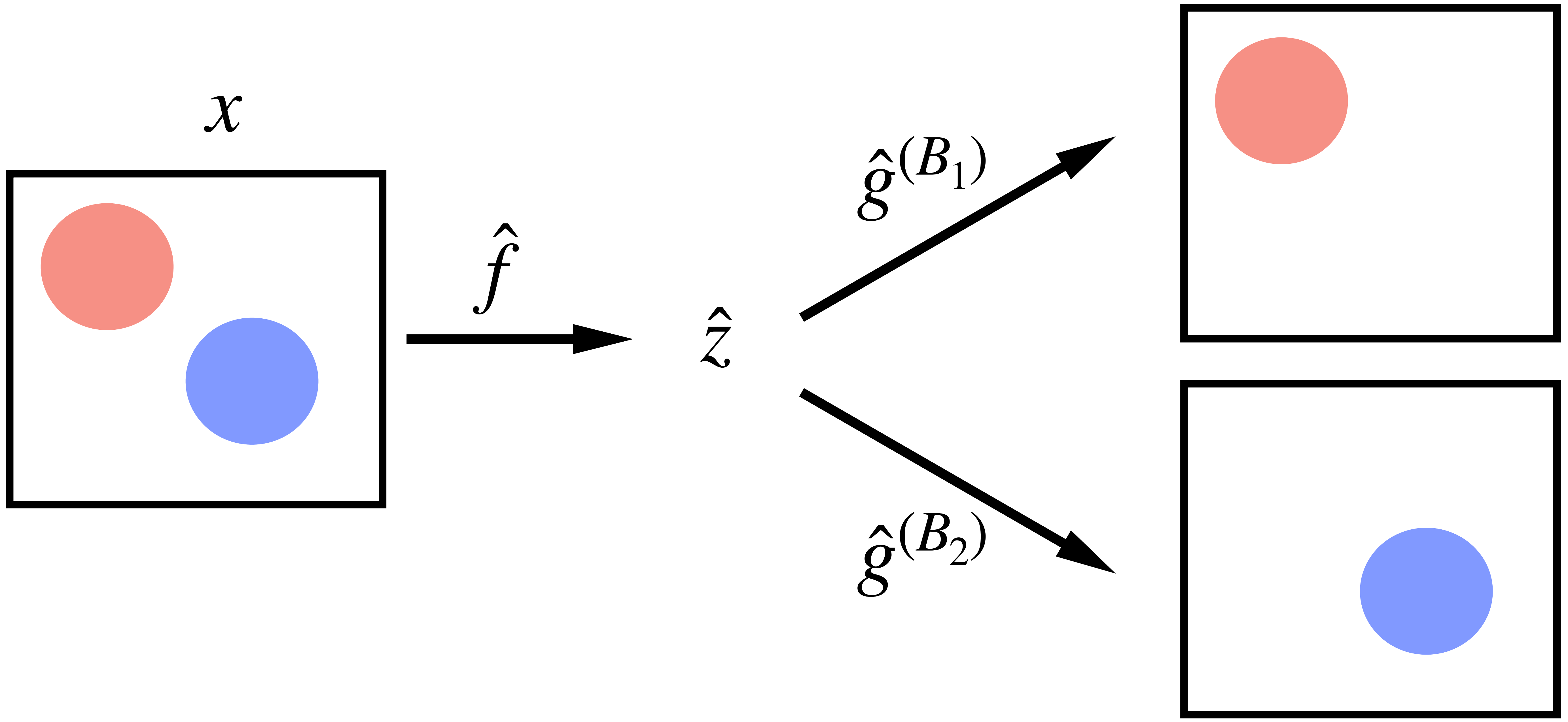
$x = g(z) = \sum_{B \in \mathscr{B}} m^{(B)}(z) \odot g^{(B)}(z_B)$ where $m^{(B)}$ denote masking mechanism

Object-centric learning approaches have shown impressive performance at disentanglement without using any weak supervision!

Block Disentanglement

- Learned Encoder: $\hat{f}(x)$
- Learned Additive Decoder: $\hat{g}(z) = \sum_{B \in \mathscr{B}} \hat{g}^{(B)}(z_B)$

If we optimise reconstruction loss perfectly, i.e., $\mathbb{E}[\,||x - \hat{g}(\hat{f}(x))||\,] = 0$, can we guarantee disentanglement of latent blocks?



Disentangled ✓

Definition of Block Disentanglement

Learned decoder $\hat{g}$ is disentangled w.r.t ground-truth decoder g
if the learned block-specific decoders "imitate" the ground-truth ones

Definition of Block Disentanglement

Learned decoder $\hat{g}$ is disentangled w.r.t ground-truth decoder g
if the learned block-specific decoders "imitate" the ground-truth ones

Precisely, for all $B \in \mathscr{B}$ we have $v_{\pi(B)}(z) = \bar{v}_{\pi_B}(z_B)$

$$\hat{g}^{(B)}(z_B) = g^{(\pi(B))}(v_{\pi(B)}(z)) + c^{(B)}$$

Permutation that sends blocks to blocks, i.e., $\pi(B) \in \mathscr{B}$

Invertible Transformation

$$\sum_{B \in \mathscr{B}} c^{(B)} = 0$$

Local & Global Disentanglement

Local Disentanglement: $\pi(B)$ depends on z
Global Disentanglement: $\pi(B)$ independent of z

Local Disentanglement: $D_{i,j} v_{\pi(B)}(z) = 0 \ \forall \ i \in \pi(B), j \notin B$

The permutation map can vary with samples in the case of local disentanglement
No unique mapping between the learned and true block decoders!

Latent Identification Guarantee

Theorem (Informal): Under the following assumptions

- Data Generation Process is additive, i.e, $x = \sum_{B \in \mathscr{B}} g^{(B)}(z_B)$
- Learned decoder is additive as well with total latent partitions as $|\mathscr{B}|$
- Ground-truth decoder is sufficiently non-linear (see paper)
- Block-specific decoders $g^{(B)}, \hat{g}^{(B)}$ are injective (for global disentanglement)

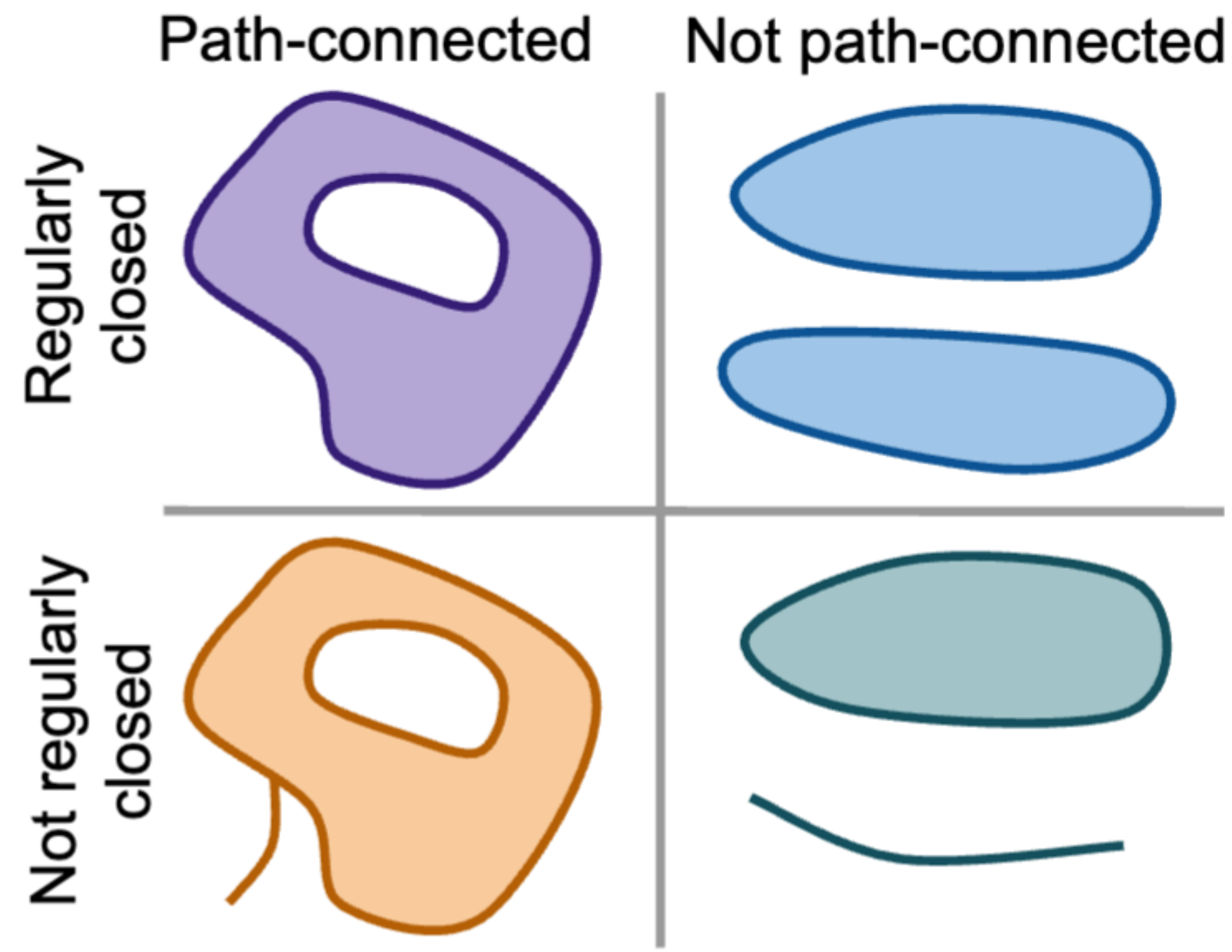
Then optimal reconstruction loss ($\mathbb{E}[||x - \hat{g}(\hat{f}(x))||]$) implies block disentanglement

We make no distributional assumptions on latent factors!

General Support for Latents

The assumptions made on the support of the distribution of latent factors

- Regularly Closed (For both local and global disentanglement)
  - Need this to define derivative uniquely over the support of training data
- Path-Connected (Only for global disentanglement)



Extrapolation with Additive Decoders

Cartesian Product Extrapolation

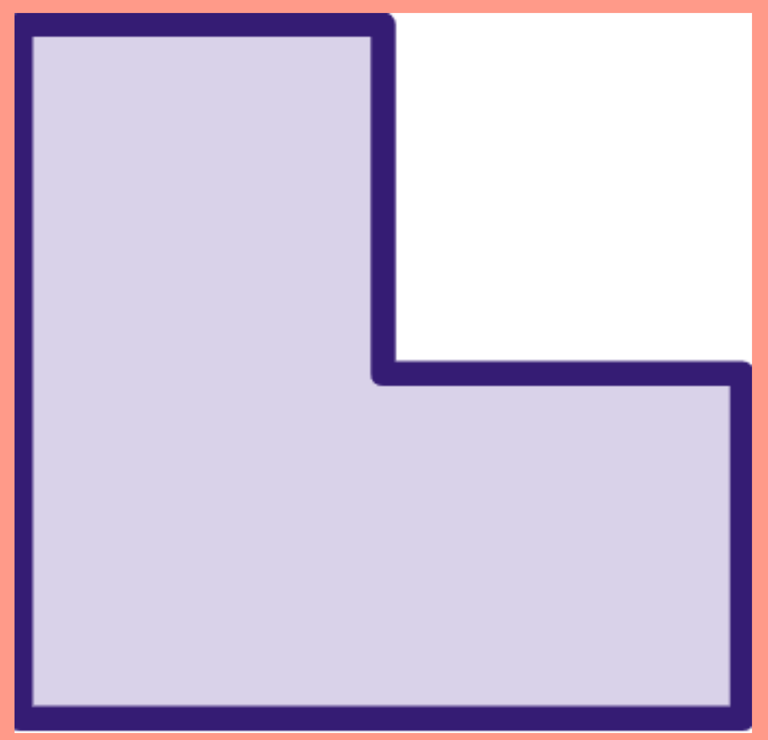
$\hat{\mathbb{Z}}^{\,train}$ = Support of learned latent factors observed during training

Cartesian Product Extrapolation:

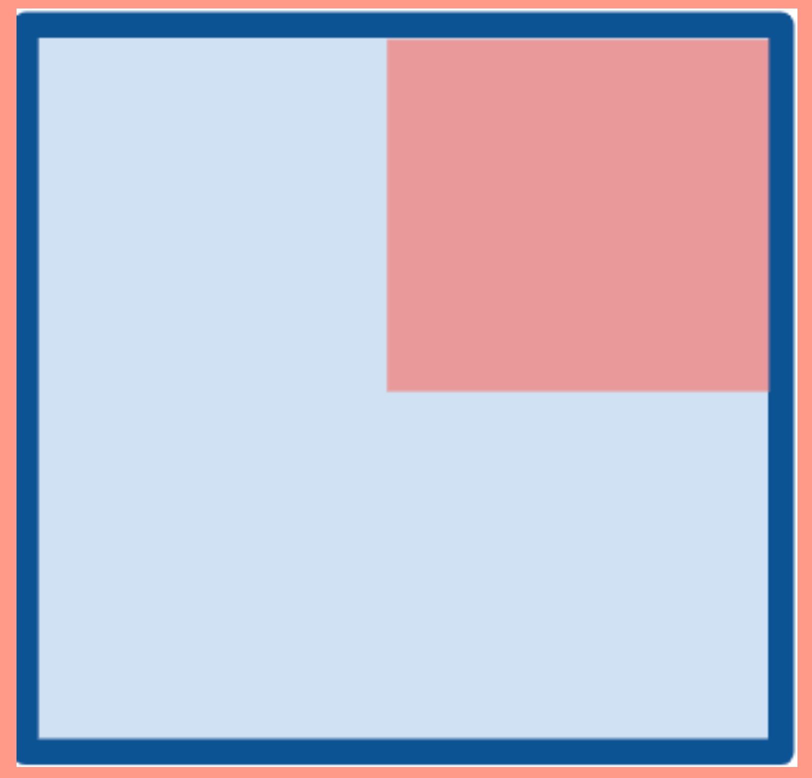
$$CPE_{\mathscr{B}}(\hat{\mathbb{Z}}^{\,train}) = \prod_{B \in \mathscr{B}} \hat{\mathbb{Z}}_B^{\,train} \quad \text{where} \quad \hat{\mathbb{Z}}_B^{\,train} = \{\hat{z}_B \mid \hat{z} \in \hat{\mathbb{Z}}^{train}\}$$

Example:

$\hat{\mathbb{Z}}^{train}$



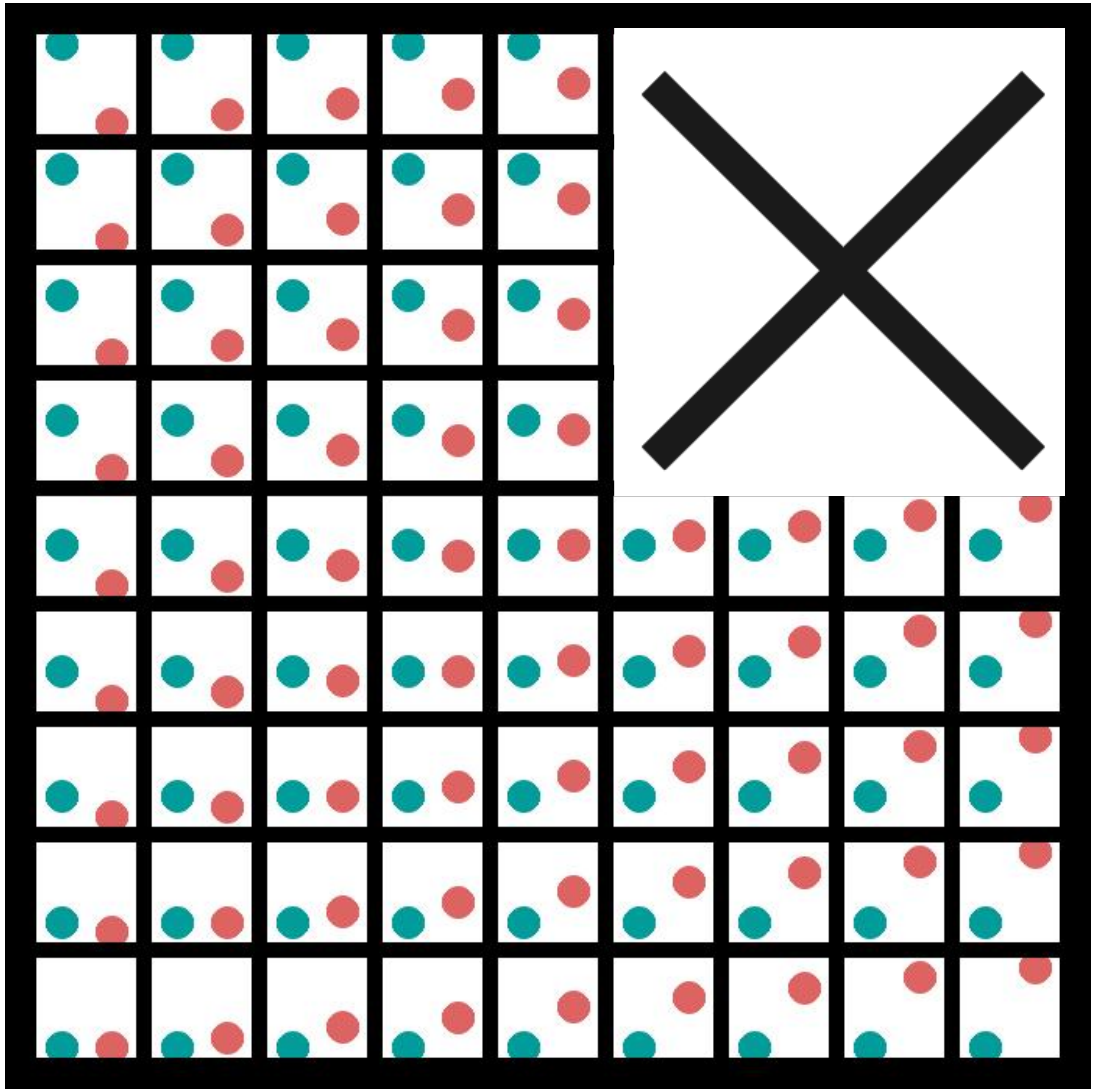
$CPE_{\mathscr{B}}(\hat{\mathbb{Z}}^{\,train})$



Corollary (Informal): Under same assumptions as previous theorem, the learned decoders imitate ground-truth decoders not only over $\hat{\mathbb{Z}}^{train}$ but also over $CPE_{\mathscr{B}}(\hat{\mathbb{Z}}^{\,train})$

Experiments

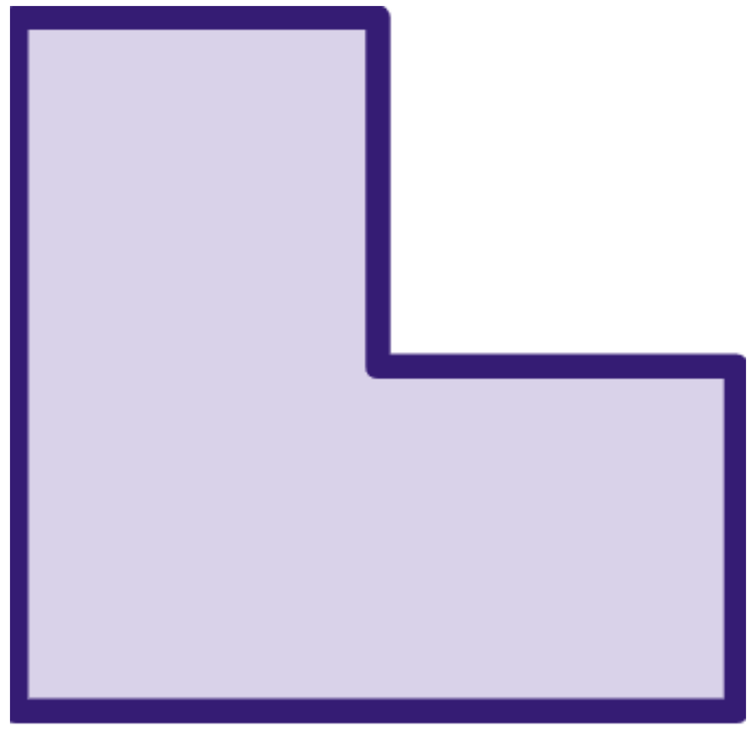
Extrapolation



Scalar Latent Dataset:

- Balls move only along y-axis
- Remove images where both balls have high y-coordinate to get L-shaped training support

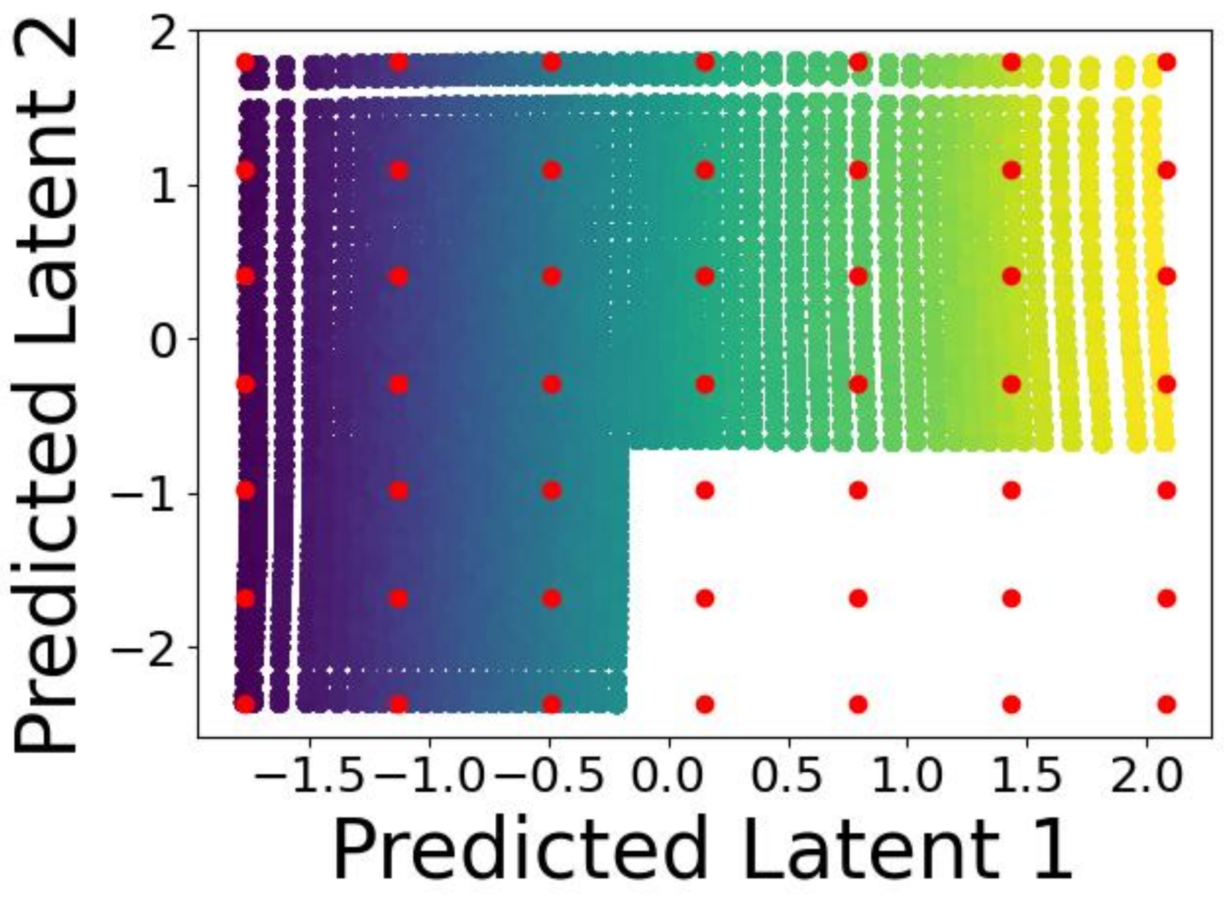
$\hat{\mathbb{Z}}^{train}$

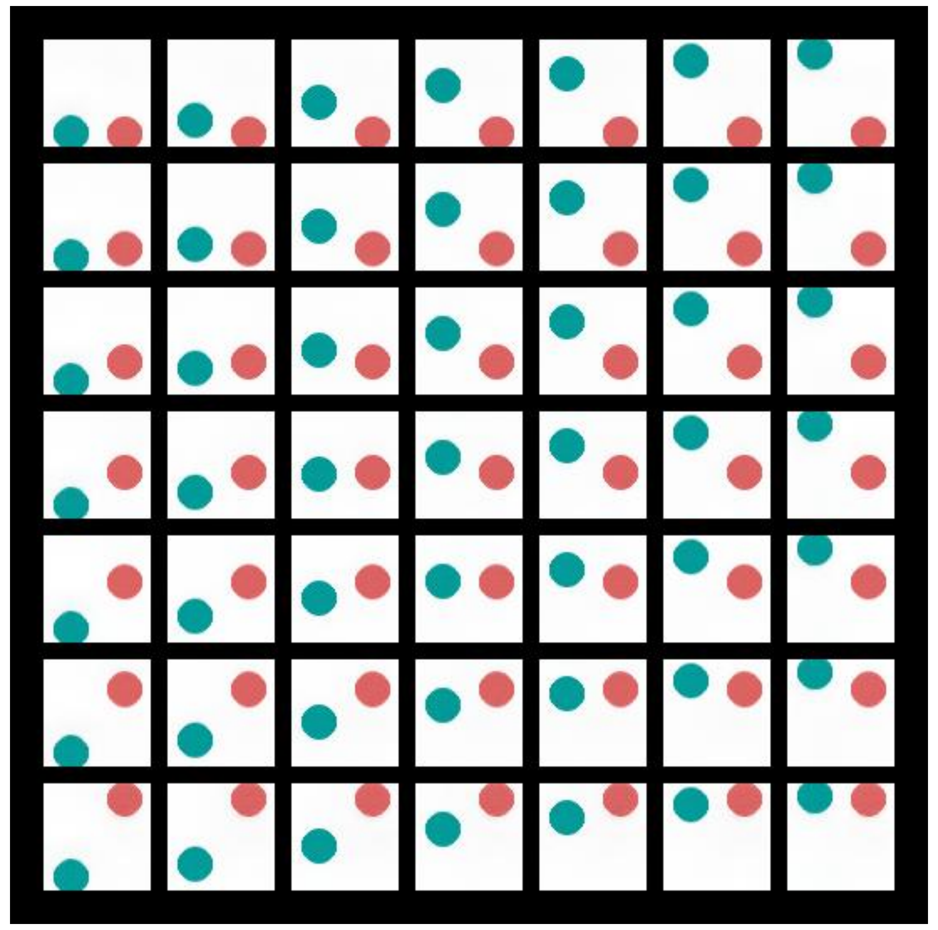


Extrapolation

Learned Latent Space | **Generated Images**

Additive Decoder

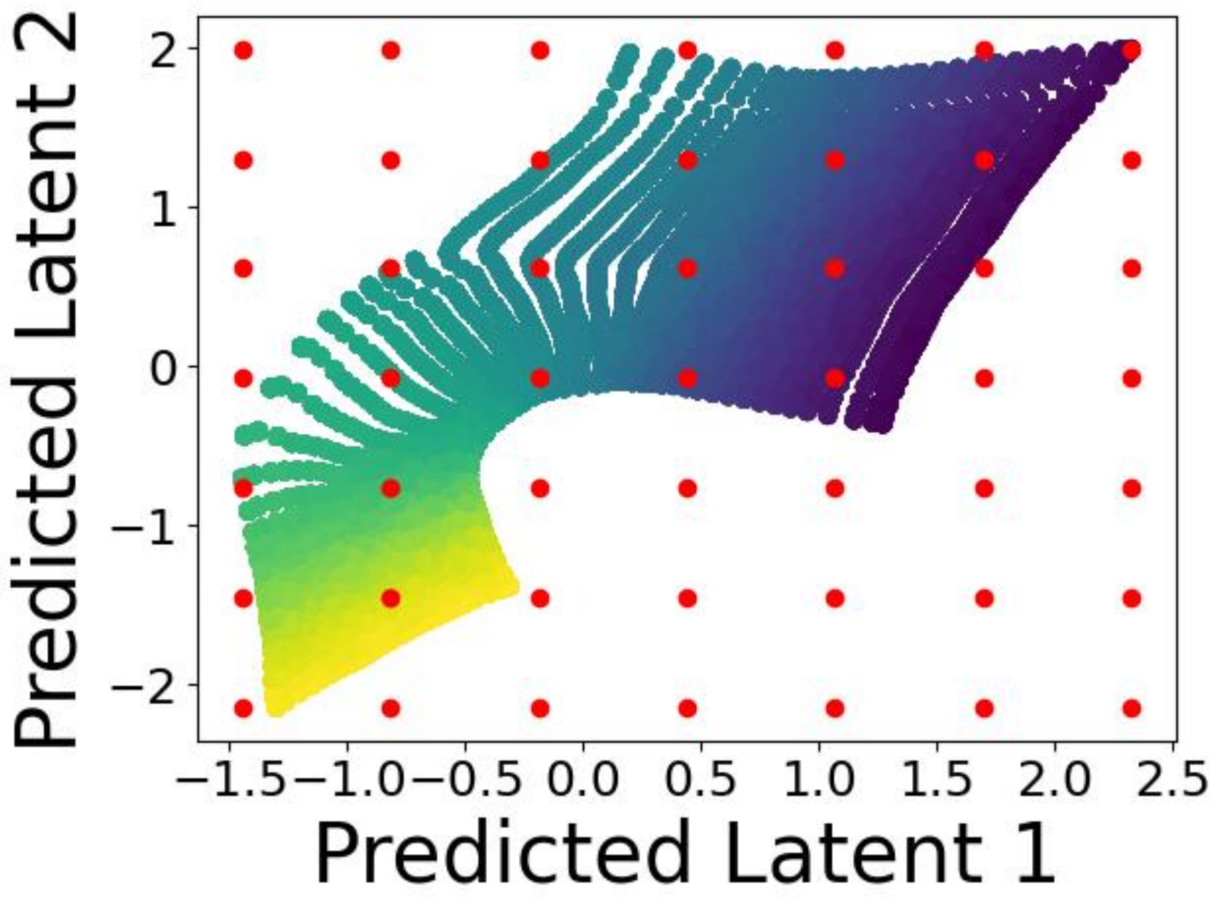


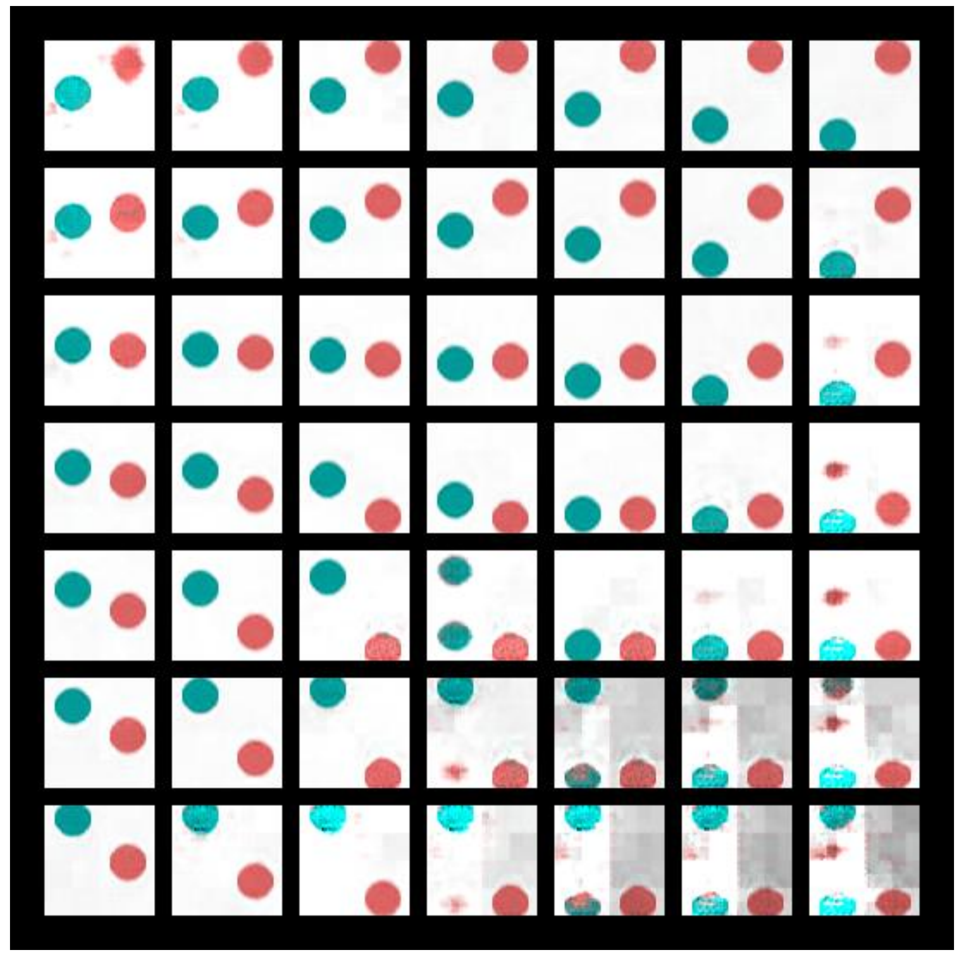


Disentangled

Changing Latent 1 only changes the blue ball

Non-Additive Decoder





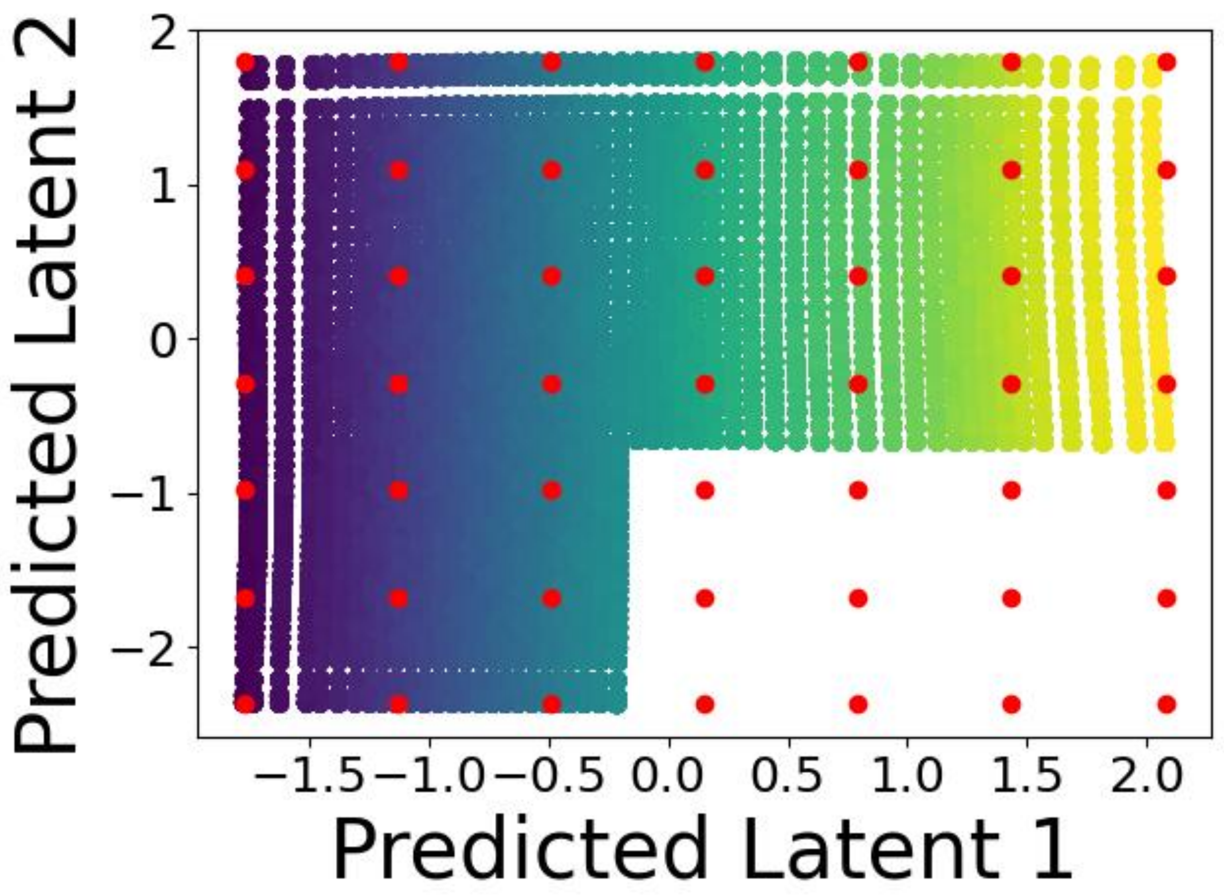
Entangled

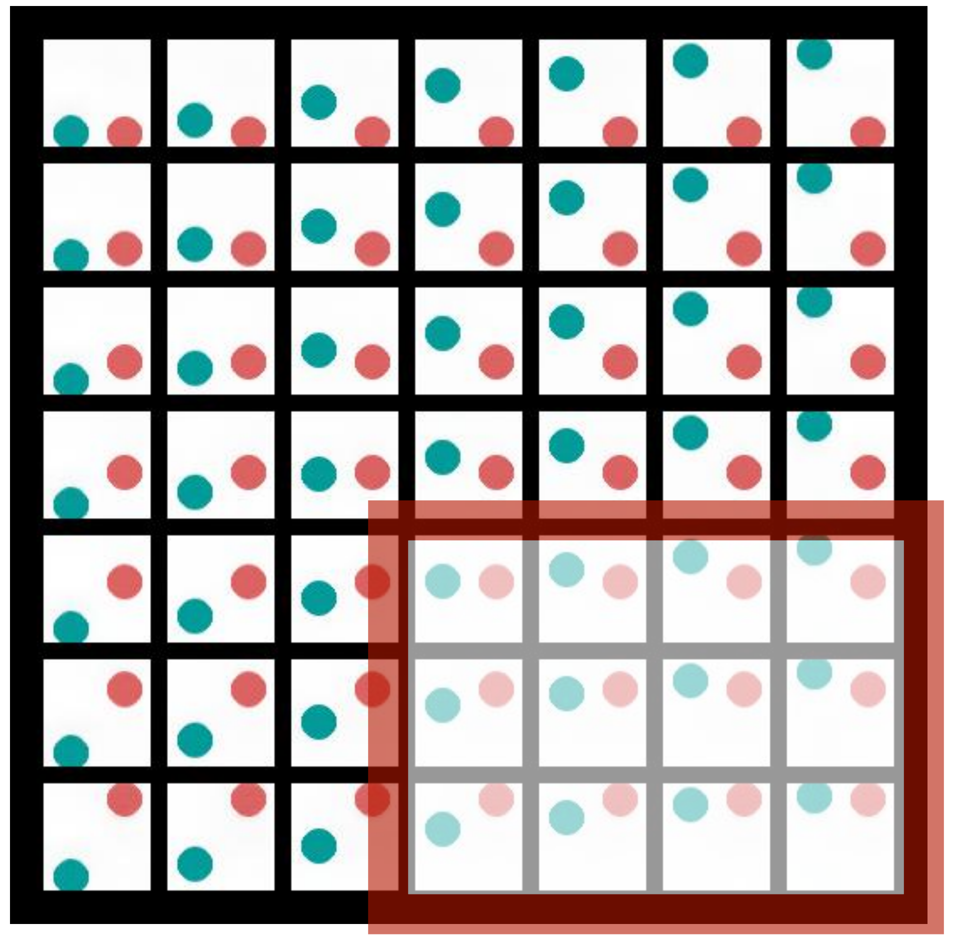
Changing Latent 1 changes both the balls

Extrapolation

Learned Latent Space | **Generated Images**

Additive Decoder

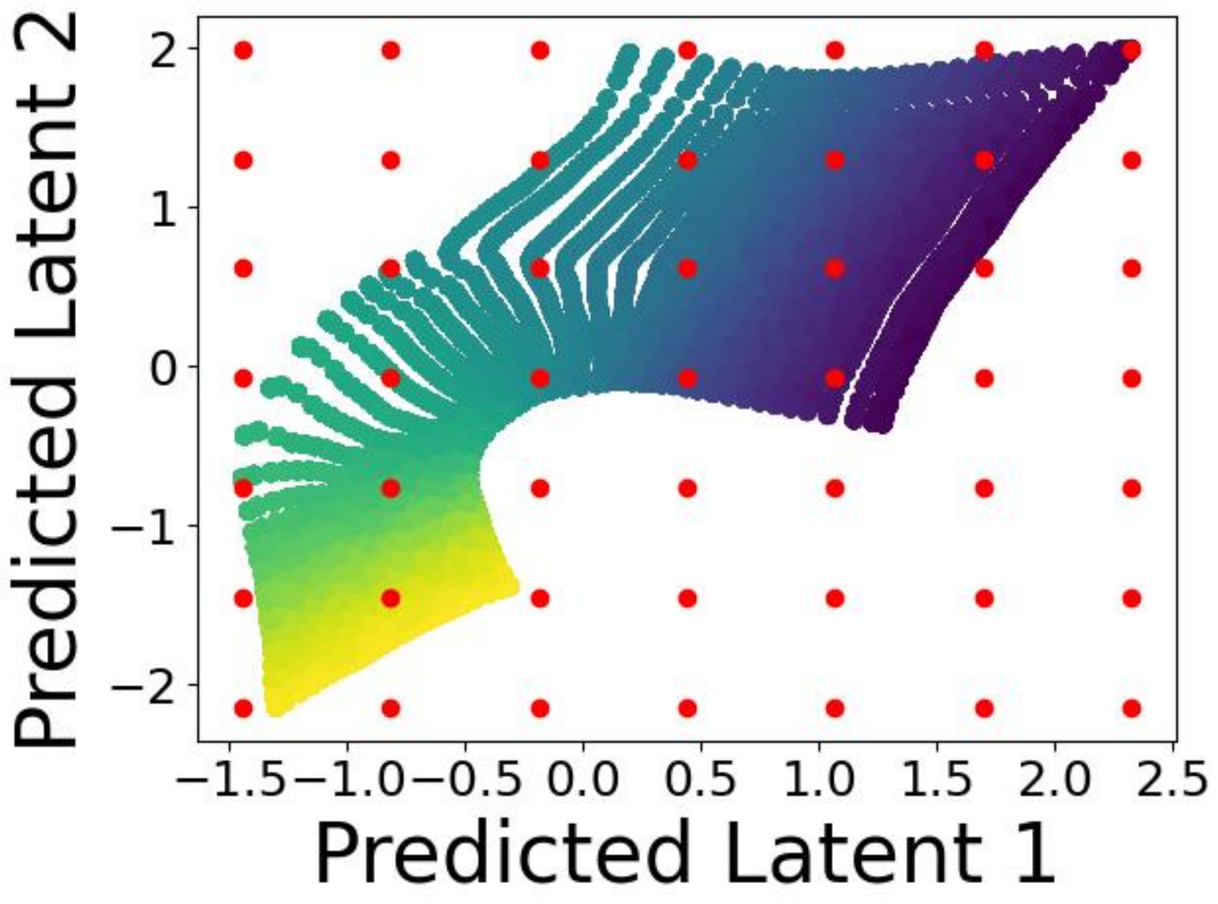


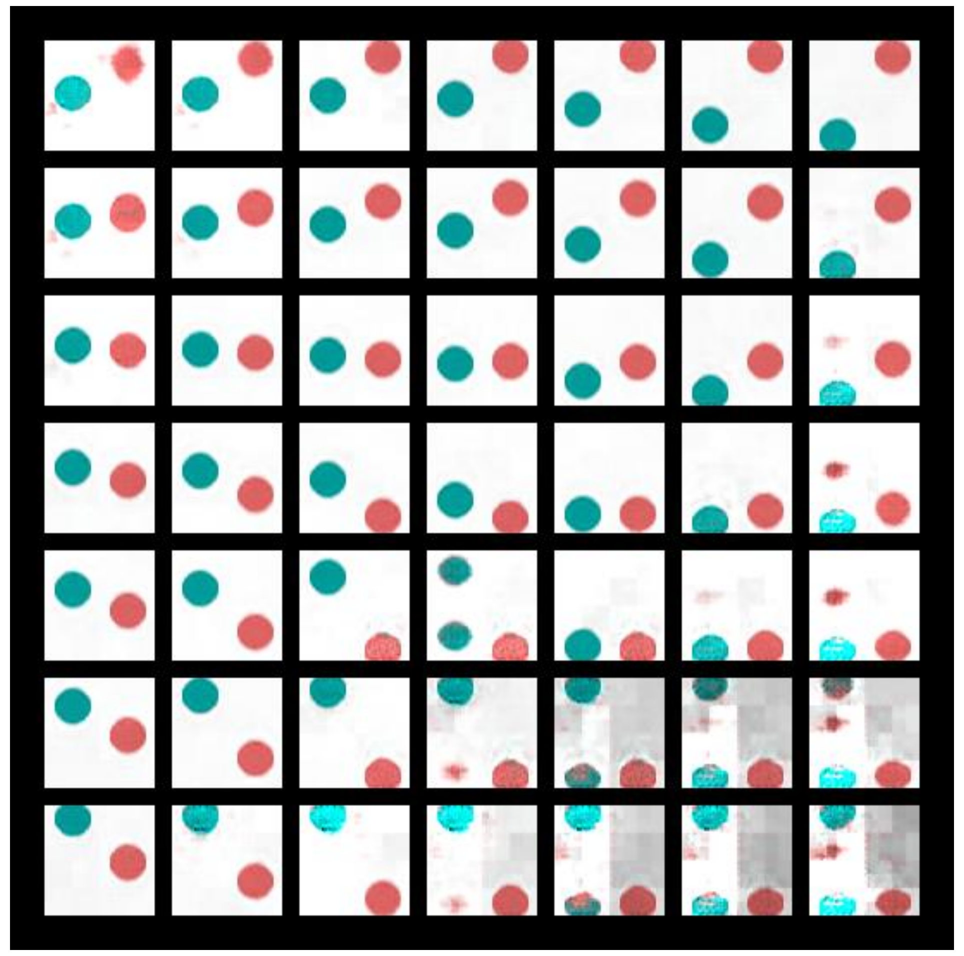


Disentangled

These samples were never seen during training

Non-Additive Decoder





Entangled

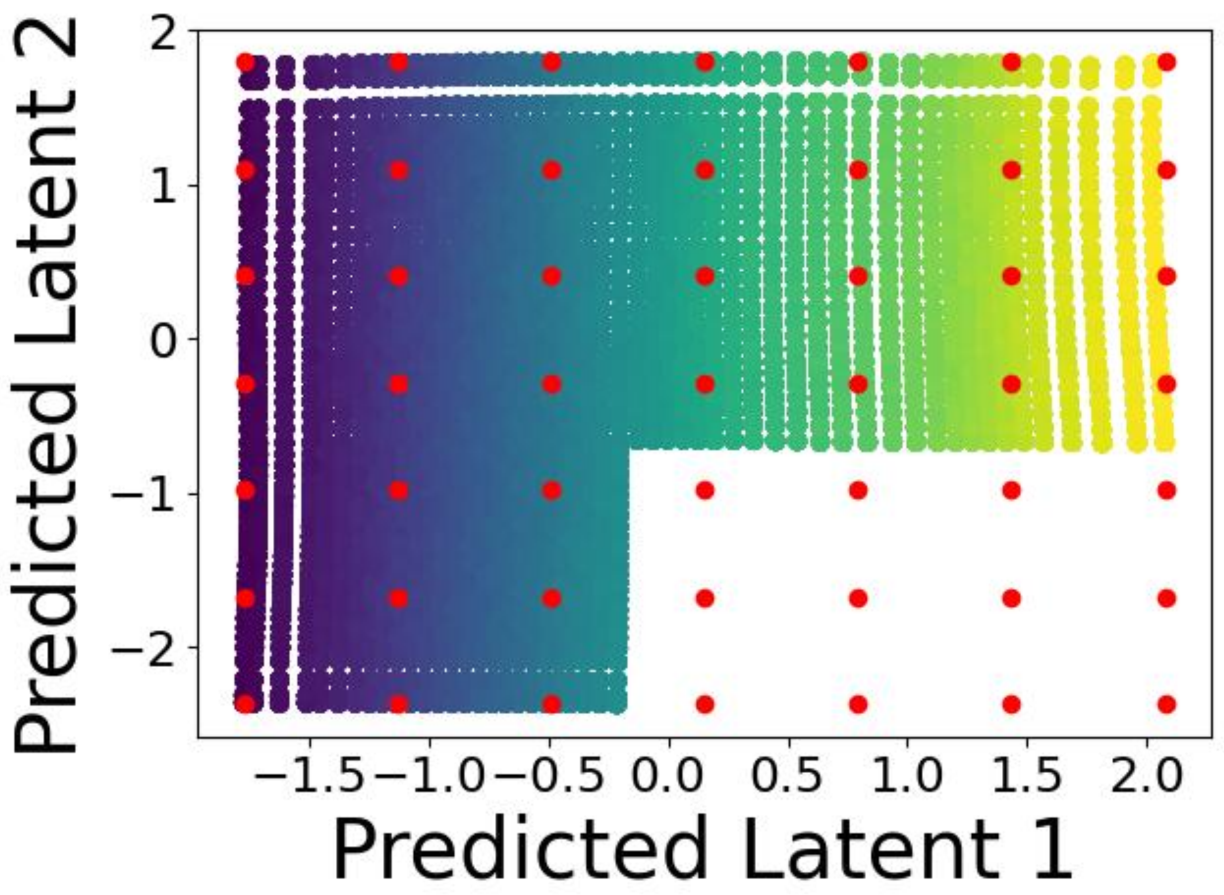
Cannot generate unseen samples

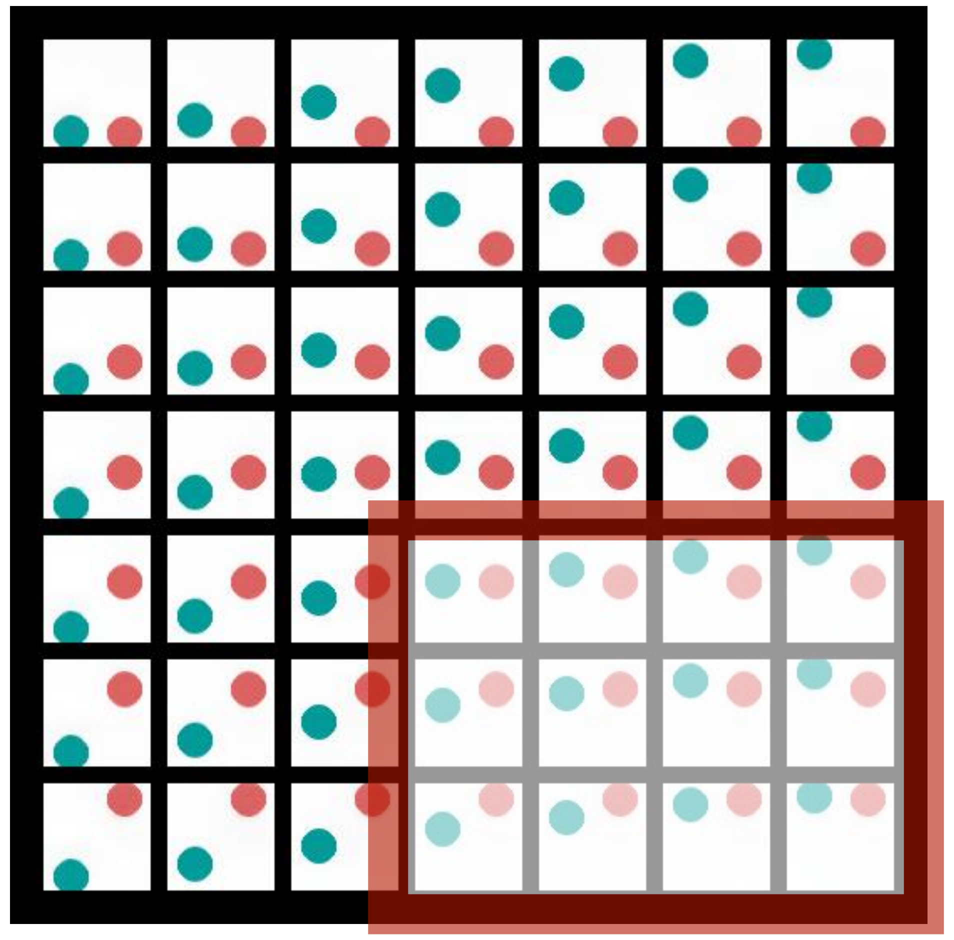
Extrapolation

Learned Latent Space

Generated Images

Additive Decoder

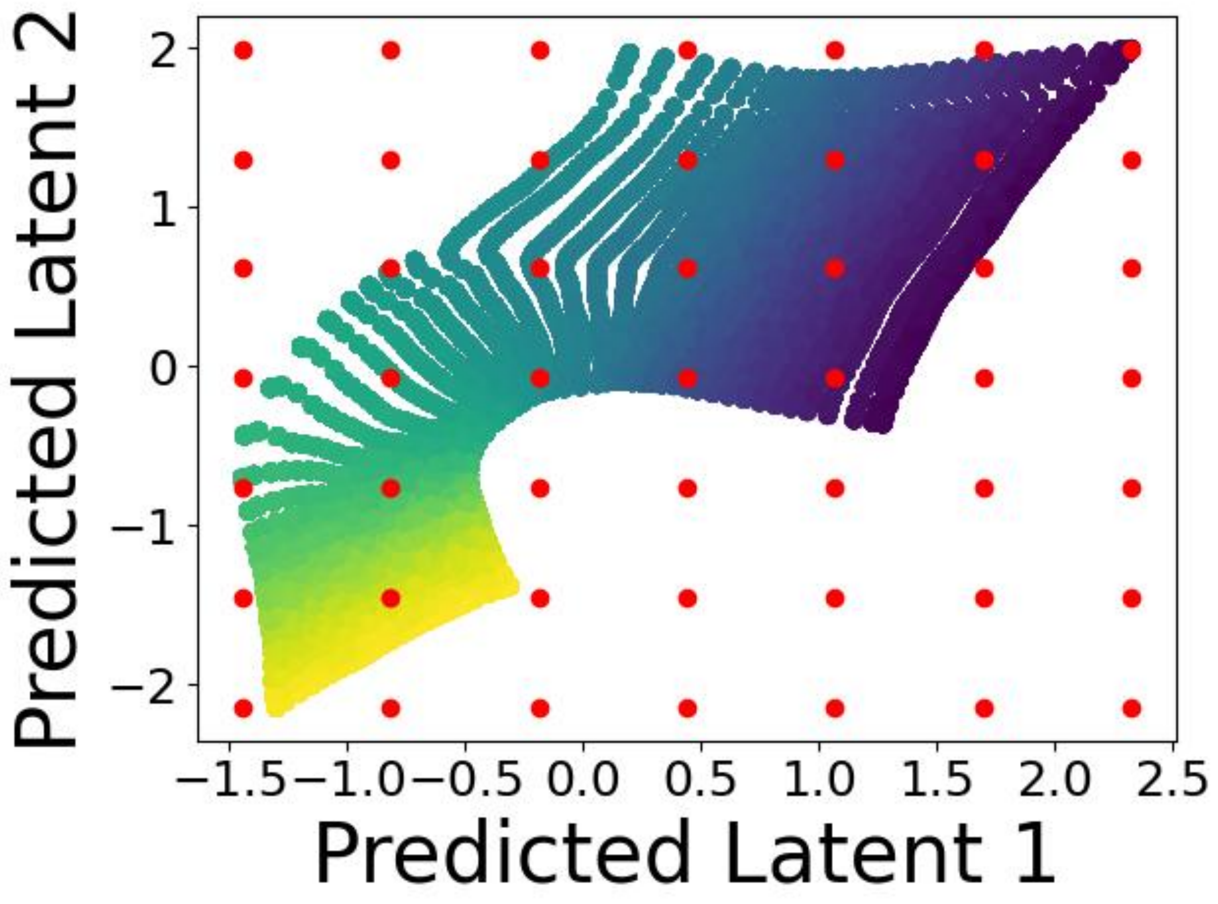


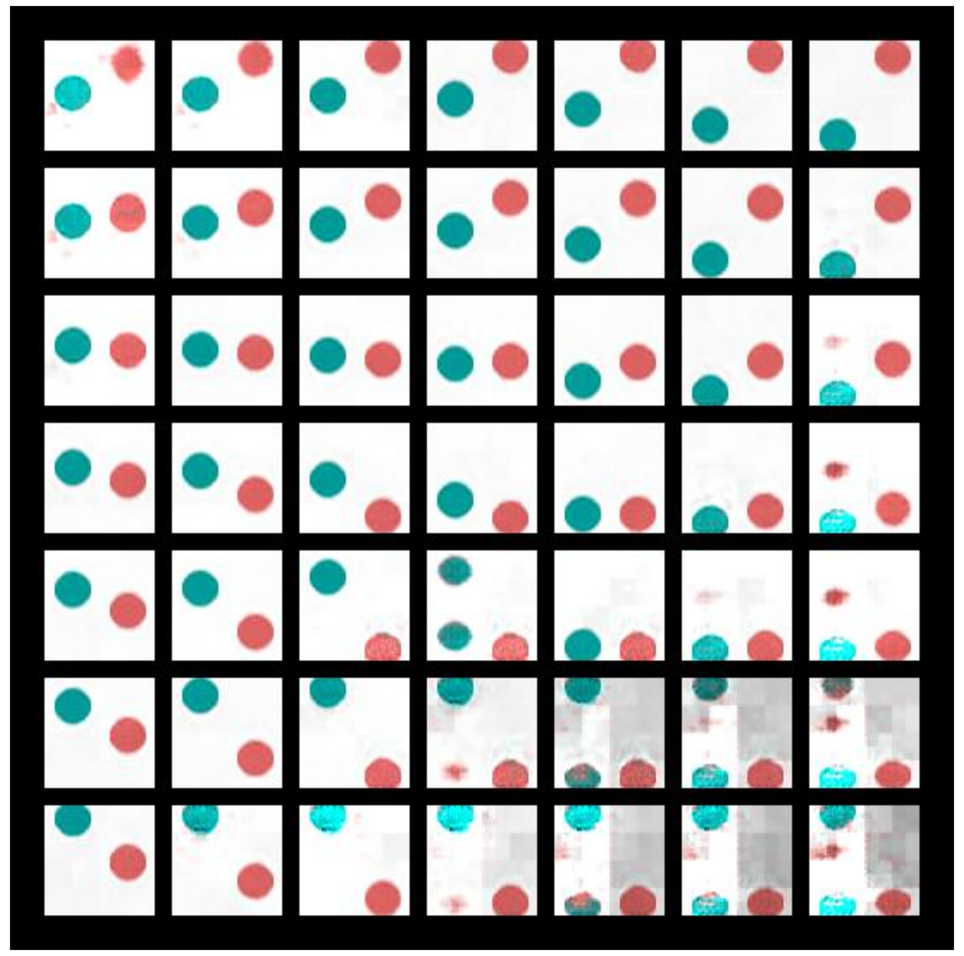


Disentangled

These samples were never seen during training

Non-Additive Decoder



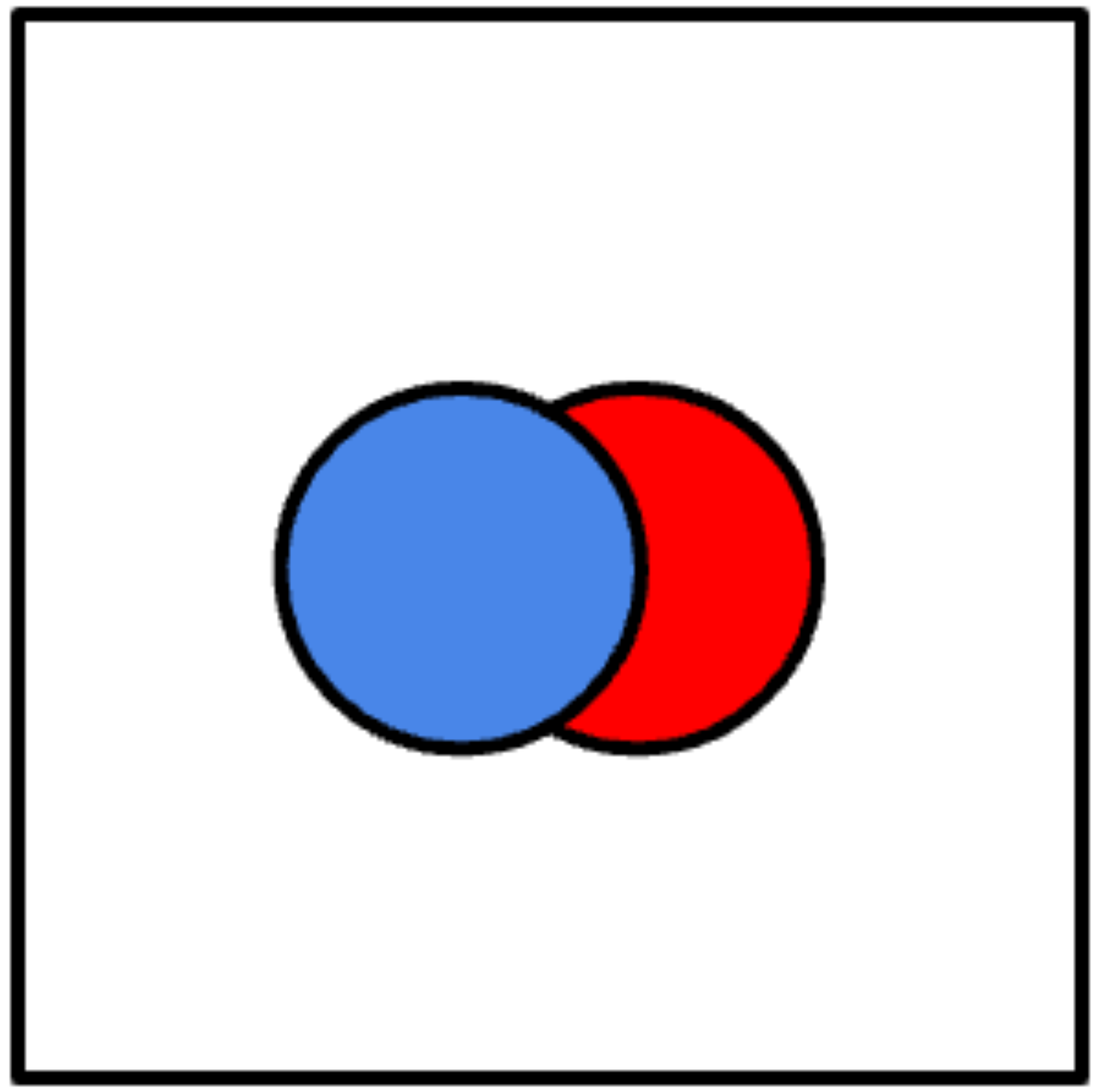


Entangled

Cannot generate unseen samples

Limitations of Additive Decoders

No interaction between latent factors



$$x = g(z_1) + g(z_2)$$

$$x = m^1(z) \odot g(z_1) + m^2(z) \odot g(z_2)$$

Additive Decoders cannot model images with occlusions!

Can we consider more expressive function classes for provable extrapolation?

Compositional Generalization with Additive Energy Models

Ongoing work with *Kartik Ahuja, Ioannis Mitliagkas, Mohammad Pezeshki, Pascal Vincent*

Additive Energy Models

$$p(x \mid z) = \frac{1}{B(z)} \exp\Big(- < 1 \, , \mathbf{E}(x, z) > \Big) \text{ where } < 1 \, , \mathbf{E}(x, z) > = \sum_{i=1}^{d_z} E_i(x, z_i)$$

Conditional distribution of data given factors

Partition Function

Energy Function

Energy Function for each component

- **Assumption**: The energy function can be decomposed as addition of energies with different components of z
- More expressive than additive decoders; can model interaction between components of z via the partition function $B(z) = \int \exp\Big(- < 1 \, , \mathbf{E}(x, z) > \Big) dx$

Contribution

We prove extrapolation guarantees for **discrete factors** with **additive energy models**

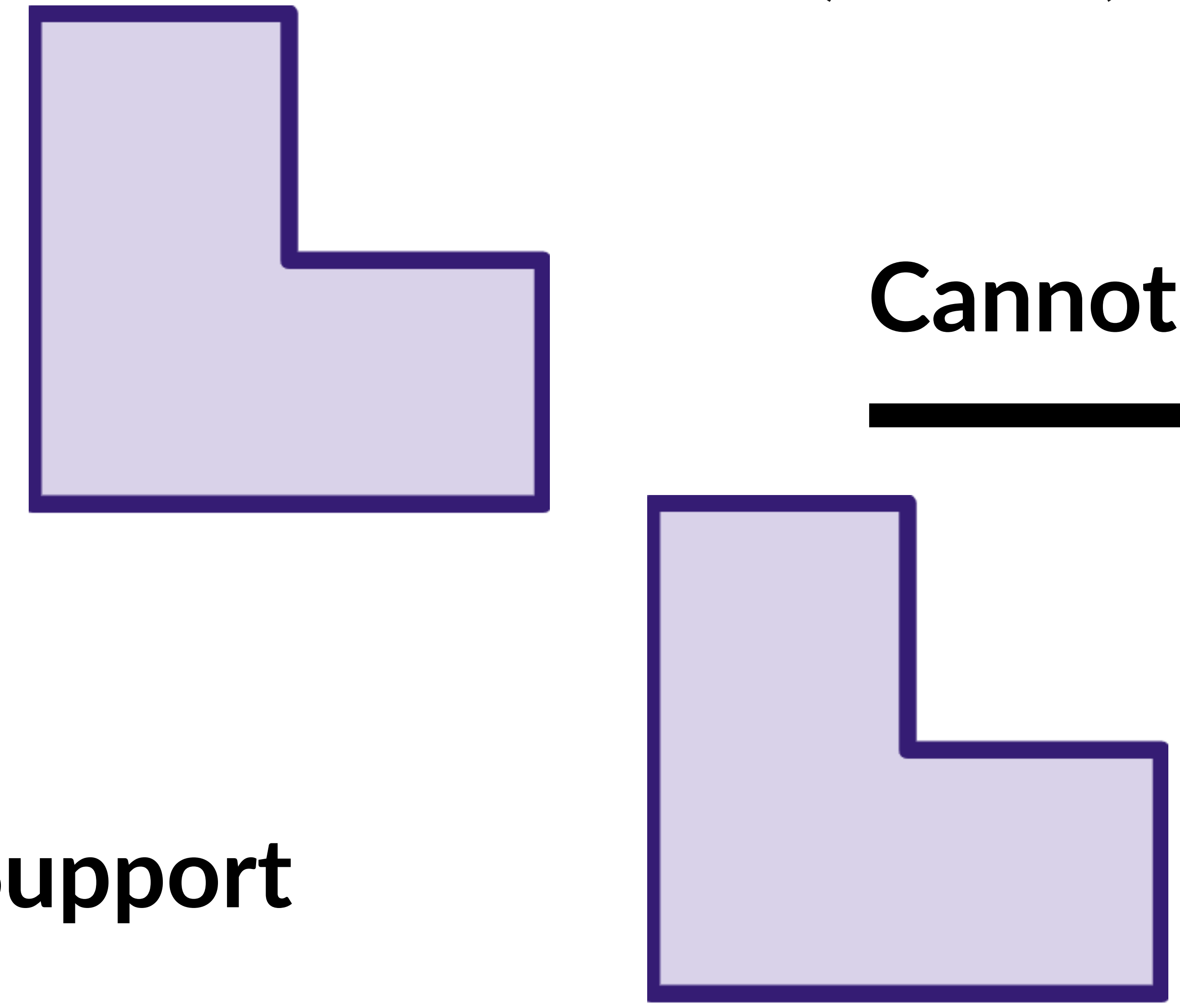
Note: We assume the factors of variations z are observed to focus on extrapolation aspect of additive energy models.

Challenges with Disconnected Support

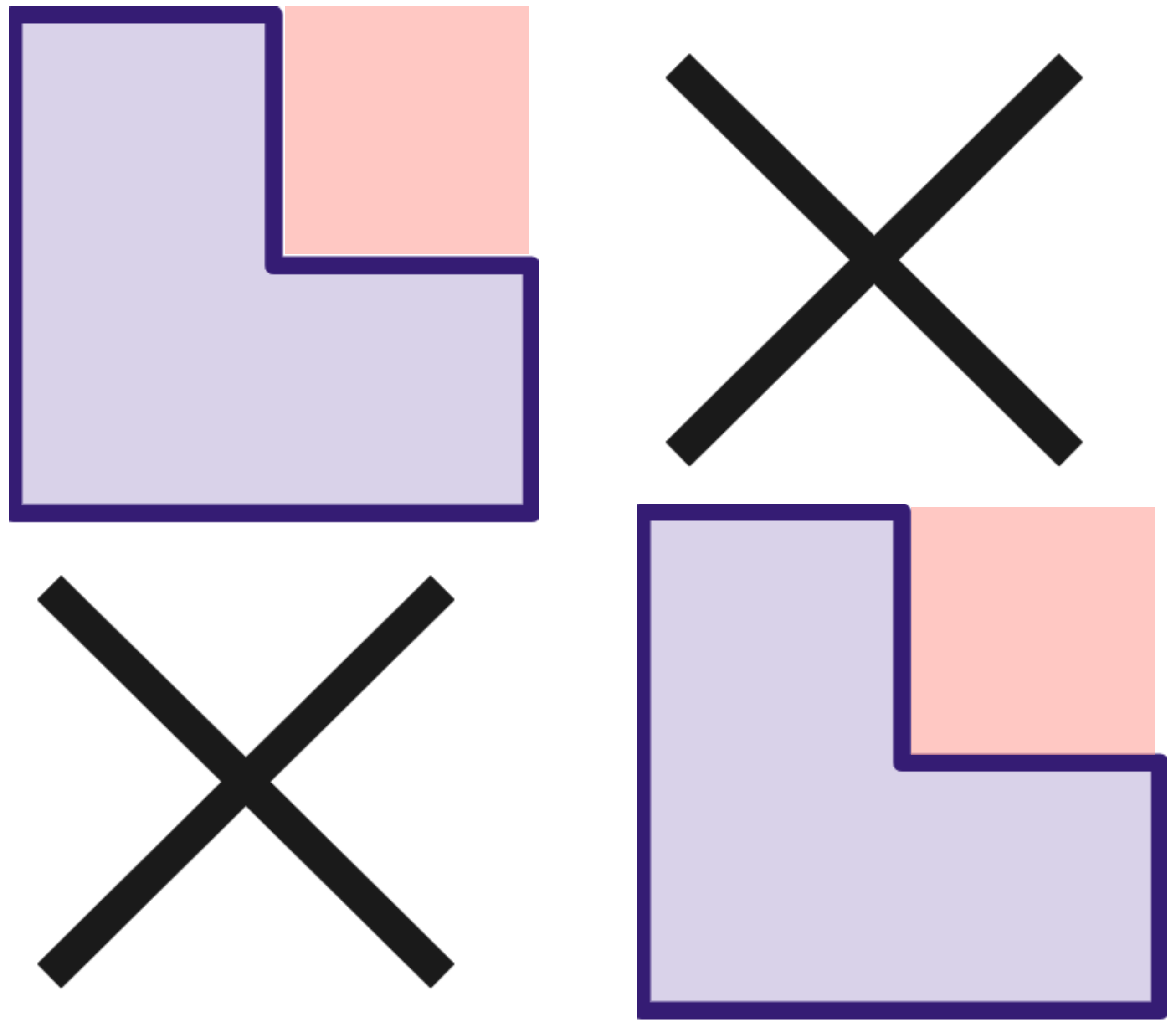
Lets revisit Cartesian-Product Extrapolation with additive functions $g(z) = \sum_{i=1}^{d_z} g_i(z_i)$

where each function $g_i : \mathbb{R} \rightarrow \mathbb{R}^{d_x}$ takes component z_i as input.

$$CPE(\mathbb{Z}^{train}) = \mathbb{Z}_1^{train} \times \mathbb{Z}_2^{train} \times \cdots \mathbb{Z}_{d_z}^{train}$$



Cannot extrapolate to $CPE(\mathbb{Z}^{train})$ $\longrightarrow$

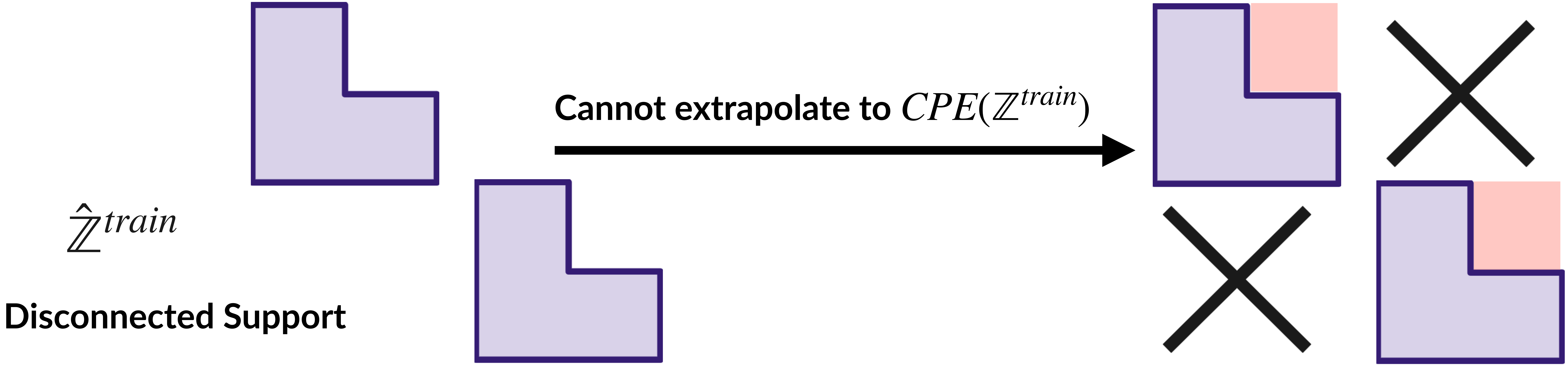


$\hat{\mathbb{Z}}^{train}$

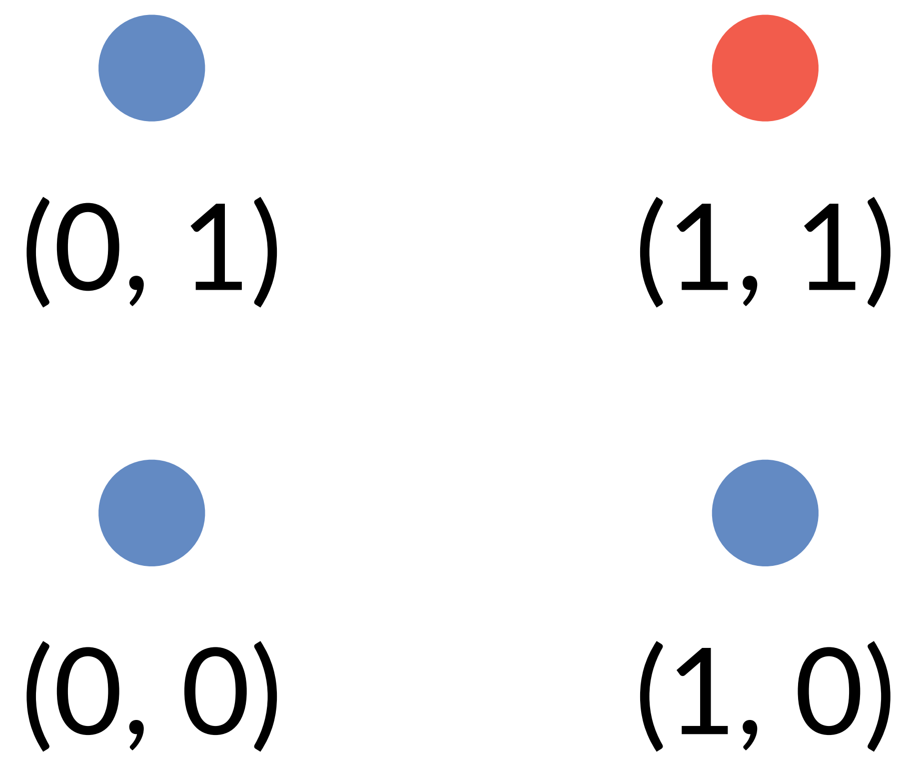
Disconnected Support

Challenges with Disconnected Support

- Disconnected support makes it hard to extrapolate to $CPE(\mathbb{Z}^{train})$
- This is a fundamental challenge when the factors z are discrete!



Affine Hull Extension



$$g(1,1) = g(1,0) + g(0,1) - g(0,0)$$

Extrapolation to novel discrete factors possible in this case!

Training Factors   Novel Factor

Affine Hull Extension of Training Support: $Aff(\mathbb{Z}^{train})$

Let each discrete component z_i takes one of m possible values.

Denote by $\tau(z_i) \in \mathbb{R}^m$ as the one-hot transformation of z_i ; $\tau(z) = [\tau(z_1), \cdots, \tau(z_{d_z})] \in \mathbb{R}^{d_z \times m}$

Then $\forall\ z \in Aff(\mathbb{Z}^{train})$ we have $\tau(z') = \sum_{z \in \mathbb{Z}^{train}} \alpha_z \tau(z)$ where $\sum_{z \in \mathbb{Z}^{train}} \alpha_z = 1$

Can Affine Hull equal Cartesian Product?

- As we add more factors to $\mathbb{Z}^{train}$, then $Aff(\mathbb{Z}^{train})$ would increase as well
- Can we show that after enough samples $Aff(\mathbb{Z}^{train})$ spans the full grid $\times_{i=1}^{d_z} [m]$?

Theorem: Assume $d_z = 2$, i.e, $z = (z_1, z_2)$ where each z_i has m possible values.

If $|\mathbb{Z}^{train}| > 8c * m \log m$, then $Aff(\mathbb{Z}^{train}) = [m] \times [m]$ with probability $\geq 1 - \frac{1}{c}$

Affine Hull Extrapolation with Additive Functions

For all $z' \in Aff(\mathbb{Z}^{train})$, we have $g(z') = \sum_{z \in \mathbb{Z}^{train}} \alpha_z g(z)$ as $\tau(z') = \sum_{z \in \mathbb{Z}^{train}} \alpha_z \tau(z)$

True Function: $g(z) = \sum_{i=1}^{d_z} g_i(z_i)$ Learned Function: $\hat{g}(z) = \sum_{i=1}^{d_z} \hat{g}_i(z_i)$

Corollary: If $g(z) = \hat{g}(z)\ \forall z \in \mathbb{Z}^{train}$ then $g(z) = \hat{g}(z)\ \forall z \in Aff(\mathbb{Z}^{train})$

Affine Hull Extrapolation with Additive Energy Models

True Model: $p(x \mid z) = \dfrac{1}{B(z)} \exp\Big(- < 1 \, , \mathbf{E}(x, z) > \Big)$

Learned Model: $p(x \mid z) = \dfrac{1}{\hat{B}(z)} \exp\Big(- < 1 \, , \hat{\mathbf{E}}(x, z) > \Big)$

Theorem: If $p(x \mid z) = \hat{p}(x \mid z) \; \forall z \in \mathbb{Z}^{train}$ then $p(x \mid z) = \hat{p}(x \mid z) \; \forall z \in Aff(\mathbb{Z}^{train})$ under the assumption of invariant support of $p(x \mid z)$

Affine Hull Extrapolation for Discriminative Case

True Model:

$$p(z \mid x) = Softmax(\log p(x \mid z) + \log p(z)) \quad \text{where} \quad p(x \mid z) = \frac{1}{B(z)} \exp(- < 1, \mathbf{E}(x, z) >)$$

Learned Model:

$$\hat{p}(z \mid x) = Softmax(\log \hat{p}(x \mid z) + \log p(z)) \quad \text{where} \quad \hat{p}(x \mid z) = \frac{1}{\hat{B}(z)} \exp(- < 1, \hat{\mathbf{E}}(x, z) >)$$

Corollary: If $p(z \mid x) = \hat{p}(z \mid x) \ \forall z \in \mathbb{Z}^{train}$ then $p(z \mid x) = \hat{p}(z \mid x) \ \forall z \in Aff(\mathbb{Z}^{train})$ under the assumption of invariant support of $p(x \mid z)$

Inferring partition function $\hat{B}(z) = \int \exp\Big(- < 1 \ , \hat{\mathbf{E}}(x, z) > \Big) dx$ is challenging!

Affine Hull Extrapolation for Discriminative Case

Learned Model: $\hat{p}(z \mid x) = Softmax\Big(- < 1, \hat{\mathbf{E}}(x, z) > - \log \hat{M}(z) + \log p(z) \Big)$

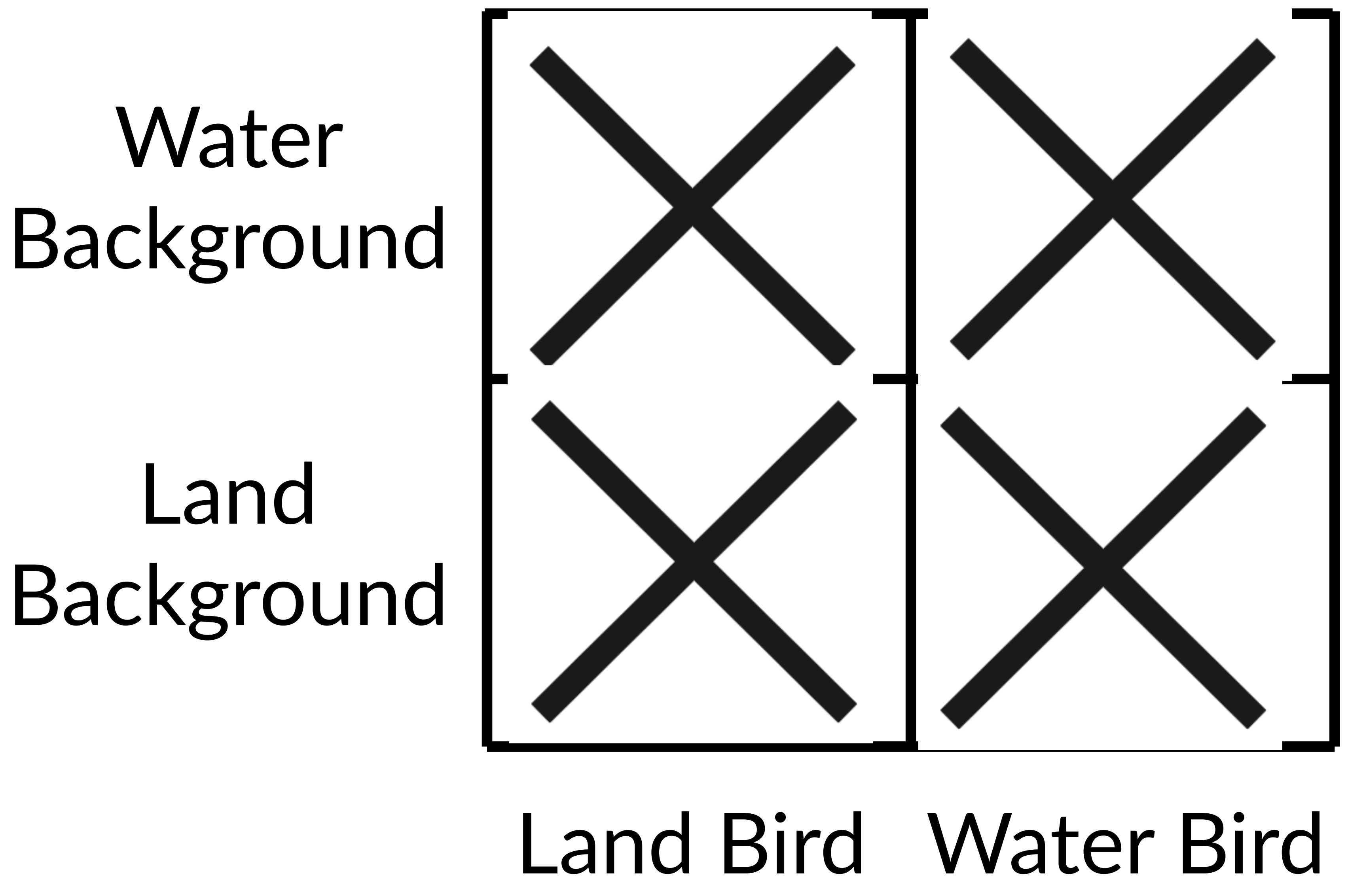
where $\hat{M}(z)$ is not constrained to be the partition function.

Theorem: If $p(z \mid x) = \hat{p}(z \mid x) \ \forall z \in \mathbb{Z}^{train}$ then $p(z \mid x) = \tilde{p}(z \mid x) \ \forall z \in Aff(\mathbb{Z}^{train})$ under the assumption of invariant support of $p(x \mid z)$, where $\tilde{p}(z \mid x)$ is defined as

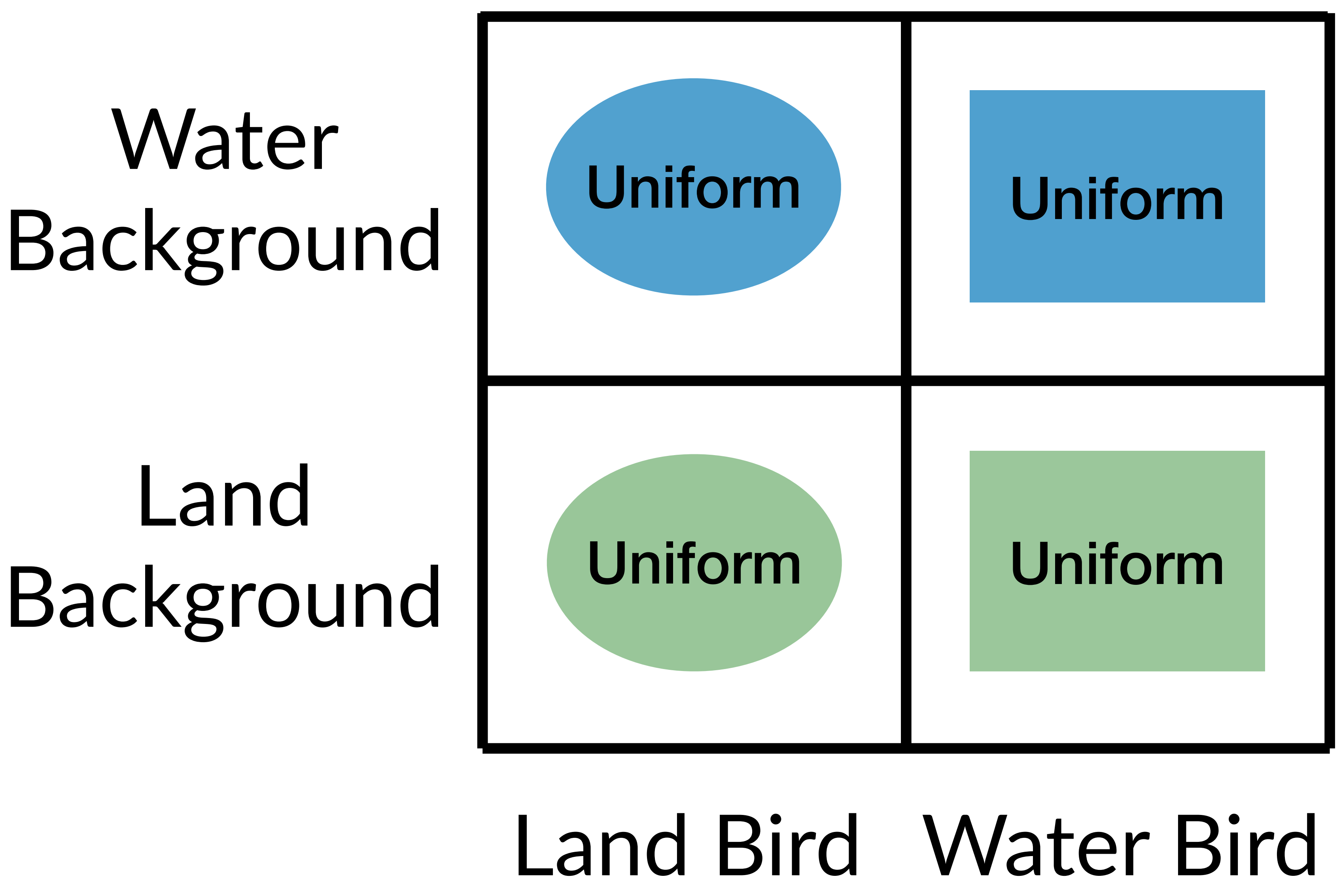
$$\tilde{p}(z \mid x) = Softmax\Big(- < 1, \hat{\mathbf{E}}(x, z) > - \log \hat{Q}(z) + \log p(z) \Big)$$

$$\hat{Q}(z) = \mathbb{E}_{x \sim p^{train}(x)} \left[\frac{exp\Big(- < 1, \hat{\mathbf{E}}(x, z) > \Big)}{\sum_{\tilde{z} \in \mathbb{Z}^{train}} \exp\Big(- < 1, \hat{\mathbf{E}}(x, \tilde{z}) > - \log \hat{M}(\tilde{z}) + \log p(\tilde{z}) \Big)} \right]$$

Experiments: Compositional Distribution Shift



Train Distribution



Test Distribution

- Factors $z = (y, a)$ where y denotes the class label and a denotes the spurious attribute
- Compositional Shift: $\mathbb{Z}^{train} \neq \mathbb{Z}^{test}$ but $\mathbb{Z}^{test} = Aff(\mathbb{Z}^{train})$

Implementation of Proposed Approach

$$\hat{p}(z \mid x) = Softmax\big(- < 1, \hat{\mathbf{E}}(x, z) > - \log \hat{M}(z) + \log p(z) \big)$$

$$\downarrow$$

$$\hat{p}(z \mid x) = Softmax\big(- < W_y, \phi(x) > - < W_a, \phi(x) > - \log \hat{M}(z) + \log p(z) \big)$$

$\phi(x)$: Representations via pretrained ResNet-50 architecture

Learnable Parameters: $W_y, W_a, \hat{M}$

Learning Objective: $\min_{W_y, W_a, \hat{M}} \mathbb{E}_{p(y,a,x)} - \log \hat{p}(y, a \mid x)$

Results

Removed (y, a)	Method	Average Acc	Worst Group Acc
(0, 0)	ERM	0.76 (0.0)	0.69 (0.0)
(0, 0)	GroupDRO	0.86 (0.0)	0.78 (0.0)
(0, 0)	AddEnergy	0.88 (0.0)	0.86 (0.0)
(0, 1)	ERM	0.71 (0.0)	0.38 (0.0)
(0, 1)	GroupDRO	0.79 (0.01)	0.41 (0.05)
(0, 1)	AddEnergy	0.87 (0.0)	0.79 (0.01)
(1, 0)	ERM	0.81 (0.01)	0.1 (0.02)
(1, 0)	GroupDRO	0.92 (0.0)	0.74 (0.04)
(1, 0)	AddEnergy	0.88 (0.0)	0.85 (0.0)
(1, 1)	ERM	0.89 (0.0)	0.53 (0.0)
(1, 1)	GroupDRO	0.91 (0.0)	0.77 (0.04)
(1, 1)	AddEnergy	0.89 (0.0)	0.86 (0.0)

Better worst group accuracy than baselines

Results for the **Waterbirds** benchmark. The performance for both the metrics is denoted as mean ± standard error over 3 random seeds on the test dataset

Future Work

Planned experiments for the current method

$$\hat{p}(z|x) = Softmax\Big(- < 1, \hat{\mathbf{E}}(x, z) > - \log \hat{M}(z) + \log p(z)\Big)$$

- Experiment on more complex subpopulation shift benchmarks
  - CivilComments, MultiNLI, NICO++, Causal Triplet

- Empirically verify the $O(m \log m)$ bound for the case of 2-dimensional factors
  - Design synthetic image datasets where we can control factors $z = (y, a)$

Future Work

Extrapolation without labelled factors z?

- A reasonable assumption is that we observe the class label (y) but not the spurious attributes ($(a_1, \cdots, a_{d_z-1})$ where $z = (y, a_1, \cdots, a_{d_z-1})$
    - Similar setup explored in recent works; XRM (Pezeshki et al.), ULA(Tsirigotis et al.)
    - However, their setup did not consider the extreme case of compositional shift

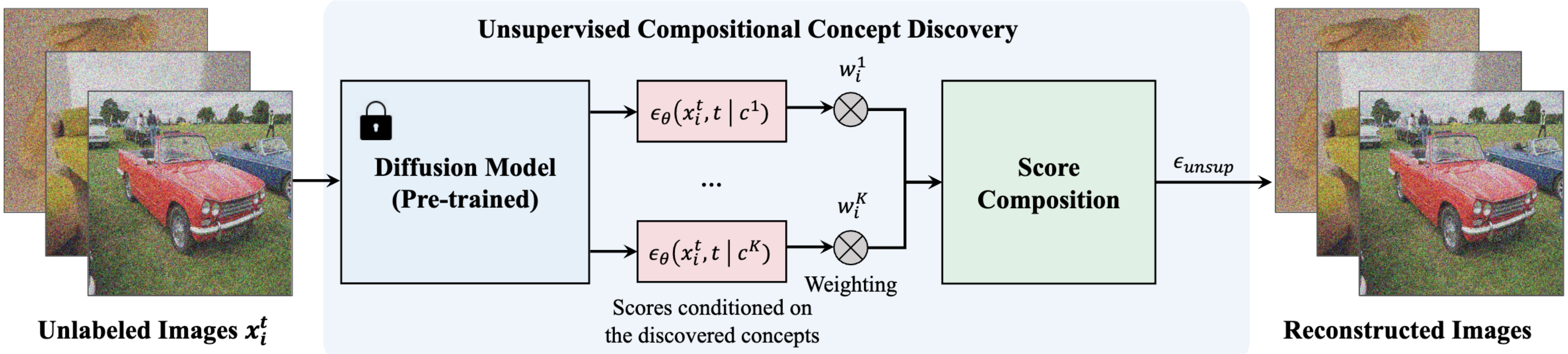
- Similar to disentanglement with weak supervision but our goal is to disentangle only the quantization of spurious features!

Future Work

Disentanglement with Additive Energy Models?

- Perhaps assumptions similar to additive decoders can help
  - $\nabla_x \log \hat{p}(x \mid \hat{z}) = \nabla_x \log p(x \mid z) \implies - < 1, \hat{\mathbf{E}}(x, \hat{z}) > = - < 1, \mathbf{E}(x, z) >$

- Empirical evidence for disentanglement with methods similar to additive energy models in recent work by Liu et al.



Thank You!

Backup Slides

Linear Identifiability of ERM

$$
\begin{aligned}
R(f) &= \mathbb{E}\left[\|Y - f(X)\|^2\right] \\
&= \mathbb{E}\left[\|\Gamma Z + N - f \circ g(Z)\|^2\right] \\
&= \mathbb{E}\Big[\|\Gamma Z - f \circ g(Z)\|^2\Big] + \mathbb{E}\left[\|N\|^2\right] - 2 * \mathbb{E}[(\Gamma Z - f \circ g(Z))^\intercal N] \\
&= \mathbb{E}\Big[\|\Gamma Z - f \circ g(Z)\|^2\Big] + \mathbb{E}\left[\|N\|^2\right] \qquad \text{(since } Z \perp N \text{ and } \mathbb{E}[N] = 0\text{)}
\end{aligned}
$$

Linear Identifiability of ERM

$$
\begin{aligned}
R(f) &= \mathbb{E}\left[\|Y - f(X)\|^2\right] \\
&= \mathbb{E}\left[\|\Gamma Z + N - f \circ g(Z)\|^2\right] \\
&= \mathbb{E}\left[\|\Gamma Z - f \circ g(Z)\|^2\right] + \mathbb{E}\left[\|N\|^2\right] - 2 * \mathbb{E}[(\Gamma Z - f \circ g(Z))^\mathsf{T} N] \\
&= \mathbb{E}\left[\|\Gamma Z - f \circ g(Z)\|^2\right] + \mathbb{E}\left[\|N\|^2\right] \qquad \text{(since } Z \perp N \text{ and } \mathbb{E}[N] = 0\text{)}
\end{aligned}
$$

Minimum risk ($R(f) = \mathbb{E}[||N||^2]$) $\implies f = \Gamma Z$

$\implies \hat{\Gamma}\hat{Z} = \Gamma Z$

Linear Identifiability of ERM

$$
\begin{aligned}
R(f) &= \mathbb{E}\big[\|Y - f(X)\|^2\big] \\
&= \mathbb{E}\big[\|\Gamma Z + N - f \circ g(Z)\|^2\big] \\
&= \mathbb{E}\Big[\|\Gamma Z - f \circ g(Z)\|^2\Big] + \mathbb{E}\big[\|N\|^2\big] - 2 * \mathbb{E}[(\Gamma Z - f \circ g(Z))^\intercal N] \\
&= \mathbb{E}\Big[\|\Gamma Z - f \circ g(Z)\|^2\Big] + \mathbb{E}\big[\|N\|^2\big] \qquad (\textbf{since } Z \perp N \textbf{ and } \mathbb{E}[N] = 0)
\end{aligned}
$$

Minimum risk ($R(f) = \mathbb{E}[\,||N||^2\,]$) $\implies f = \Gamma Z$

$\implies \hat{\Gamma}\hat{Z} = \Gamma Z$

Since $\hat{\Gamma} \in R^{k,d}$ is full column rank (k>d), we can find a set S of d rows such that $\hat{W} = [\hat{\Gamma}]_{S,:}$ is invertible

Linear Identifiability of ERM

$$
\begin{aligned}
R(f) &= \mathbb{E}\left[\|Y - f(X)\|^2\right] \\
&= \mathbb{E}\left[\|\Gamma Z + N - f \circ g(Z)\|^2\right] \\
&= \mathbb{E}\left[\|\Gamma Z - f \circ g(Z)\|^2\right] + \mathbb{E}\left[\|N\|^2\right] - 2 * \mathbb{E}[(\Gamma Z - f \circ g(Z))^\intercal N] \\
&= \mathbb{E}\left[\|\Gamma Z - f \circ g(Z)\|^2\right] + \mathbb{E}\left[\|N\|^2\right] \qquad \text{(since } Z \perp N \text{ and } \mathbb{E}[N] = 0\text{)}
\end{aligned}
$$

Minimum risk ($R(f) = \mathbb{E}[\,||N||^2]$)

$$
\begin{aligned}
&\implies f = \Gamma Z \\
&\implies \hat{\Gamma}\hat{Z} = \Gamma Z \\
&\implies \hat{W}\hat{Z} = WZ \\
&\implies \hat{Z} = (\hat{W})^{-1} WZ \\
&\implies \hat{Z} = AZ
\end{aligned}
$$

Linear Identifiability of IC-ERM

$$
\begin{aligned}
R(f) &= \mathbb{E}\left[\|Y - f(X)\|^2\right] \\
&= \mathbb{E}\left[\|\Gamma Z + N - f \circ g(Z)\|^2\right] \\
&= \mathbb{E}\left[\|\Gamma Z - f \circ g(Z)\|^2\right] + \mathbb{E}\left[\|N\|^2\right] - 2 * \mathbb{E}[(\Gamma Z - f \circ g(Z))^\top N] \\
&= \mathbb{E}\left[\|\Gamma Z - f \circ g(Z)\|^2\right] + \mathbb{E}\left[\|N\|^2\right] \qquad \text{(since } Z \perp N \text{ and } \mathbb{E}[N] = 0\text{)}
\end{aligned}
$$

Minimum risk ($R(f) = \mathbb{E}[\,||N||^2]$) $\implies f = \Gamma Z$

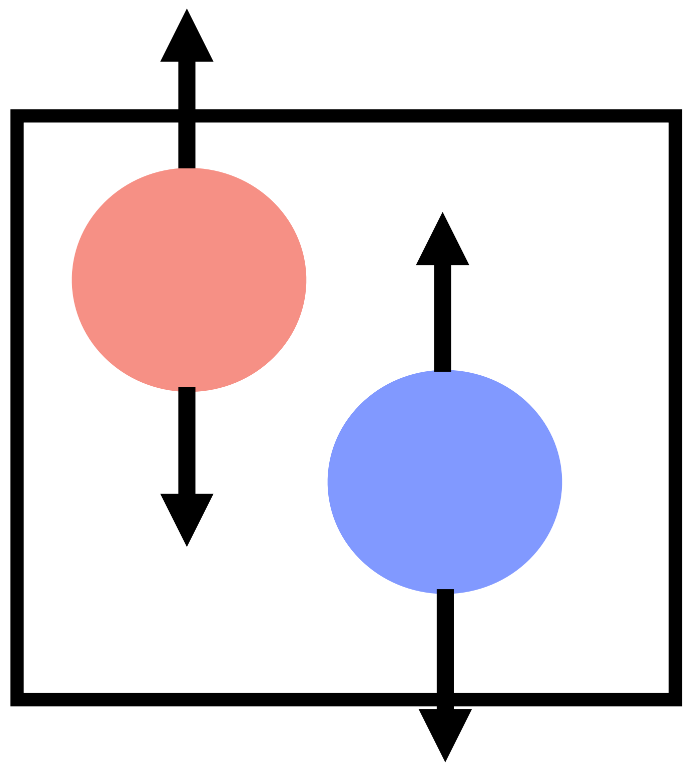
$\implies \hat{Z} = AZ$

$\hat{Z}$ has mutually independent components; similar to the **Linear ICA** problem!
Hence, A must be **permutation & scaling** matrix.

Datasets

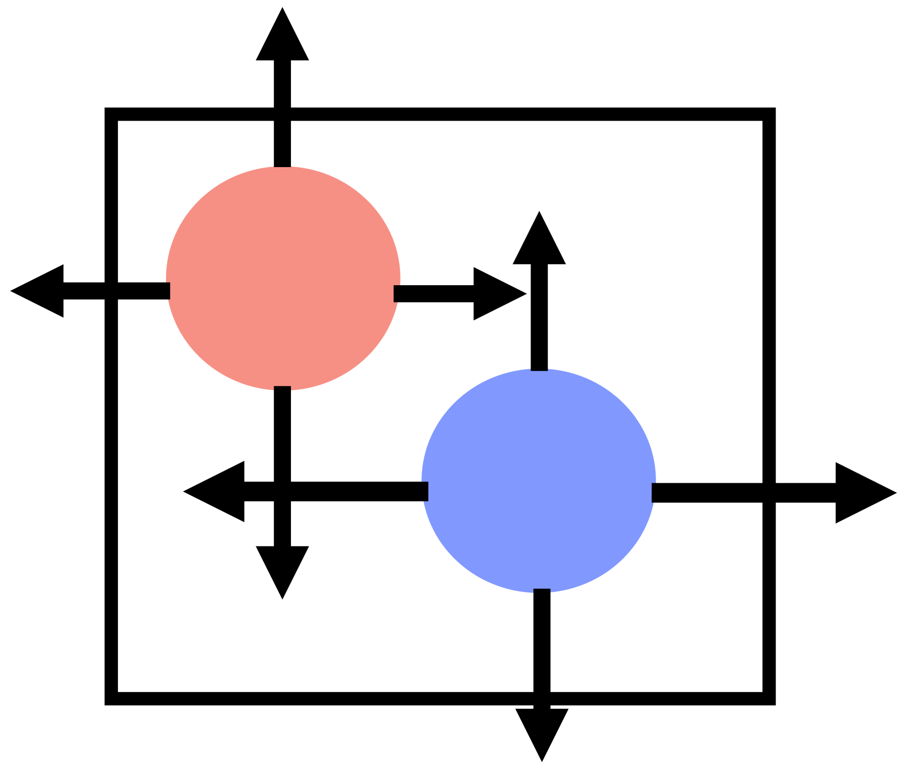
Scalar Latent Dataset: Balls move only along y-axis

$$\dim(Z) = 2$$
$$\mathscr{B} = \{\{1\}, \{2\}\}$$



Block Latent Dataset: Balls move along both x and y-axis

$$\dim(Z) = 4$$
$$\mathscr{B} = \{\{1,2\}, \{3,4\}\}$$



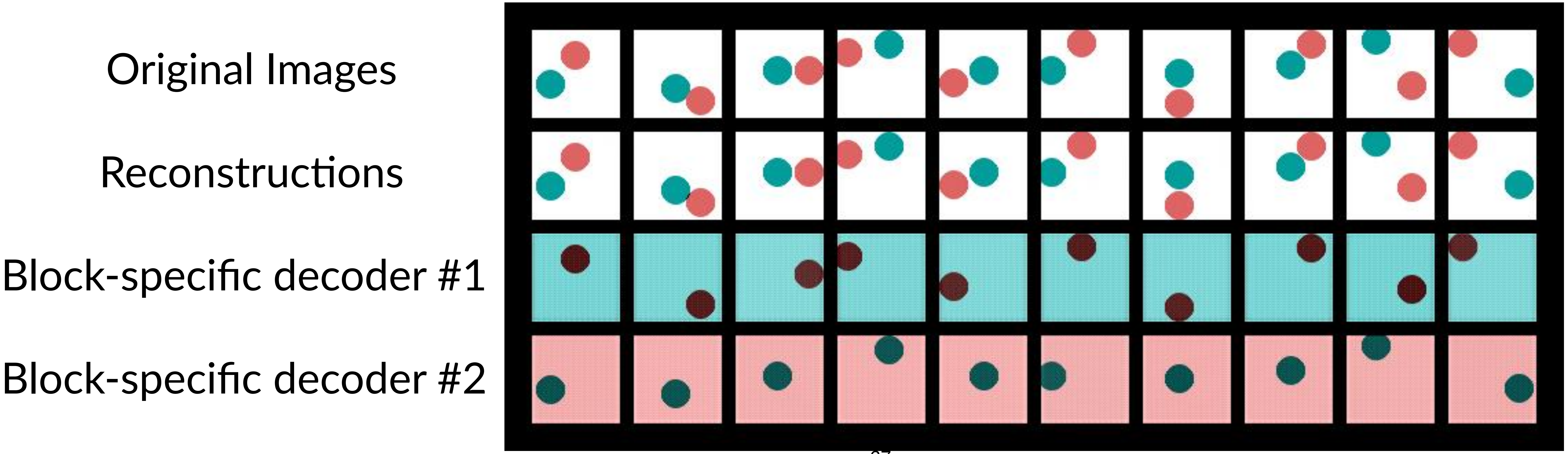
Independent Case: $\boldsymbol{z}_{B_1} \perp \boldsymbol{z}_{B_2}$

Dependent Case: $\boldsymbol{z}_{B_1} \not\perp \boldsymbol{z}_{B_2}$

Disentanglement

	Scalar Latent	Block Latent Independent	Block Latent Dependent
Non-Additive Decoder	70.6 (5.2)	53.9 (7.6)	78.1 (2.9)
Additive Decoder	**91.5 (3.6)**	**92.2 (4.9)**	**99.9 (0.0)**

Modified MCC score (Higher implies more disentangled)



Proof Sketch

Samples from the CPE still have the support for marginal distributions!

$$Supp(Z_B^{tr}) = Supp(Z_B^{te})$$

Precisely, for all $B \in \mathscr{B}$ we have

$$\hat{f}^{(B)}(z_B) = f^{(\pi(B))}(v_{\pi(B)}(z_B)) + c^{(B)}$$

Block decoders equality holds true for all blocks on samples from CPE

Additivity enables you to get final image via addition of block decoders!

Global Disentanglement is necessary for compositionality!

We need the learned block decoders to correspond in unique manner to true block decoders!

Proof Sketch

$$\mathbb{E}^{\text{train}}||\boldsymbol{x} - \hat{\boldsymbol{f}}(\hat{\boldsymbol{g}}(\boldsymbol{x}))||^2 = \mathbb{E}^{\text{train}}||\boldsymbol{f}(\boldsymbol{z}) - \hat{\boldsymbol{f}}(\hat{\boldsymbol{g}}(\boldsymbol{f}(\boldsymbol{z})))||^2 = 0\,,$$

$\hat{f}$: Diffeomorpshim

$\hat{g}$: continuous

$$\boldsymbol{f} \circ \boldsymbol{v}(\boldsymbol{z}) = \hat{\boldsymbol{f}}(\boldsymbol{z})\ \forall \boldsymbol{z} \in \hat{\mathcal{Z}}^{\text{train}}$$

$$\sum_{B \in \mathcal{B}} \boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z})) = \sum_{B \in \mathcal{B}} \hat{\boldsymbol{f}}^{(B)}(\boldsymbol{z}_B)\ \forall \boldsymbol{z} \in \hat{\mathcal{Z}}^{\text{train}}\,.$$

$\boldsymbol{v} := \boldsymbol{f}^{-1} \circ \hat{\boldsymbol{f}}$ is a C^2-diffeomorphism

Derivative w.r.t $z_j \in J$ for some $J \in \mathscr{B}$

$$\sum_{B \in \mathcal{B}} \sum_{i \in B} D_i \boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z})) D_j \boldsymbol{v}_i(\boldsymbol{z}) = D_j \hat{\boldsymbol{f}}^{(J)}(\boldsymbol{z}_J)\,.$$

Proof Sketch

$$\boldsymbol{f} \circ \boldsymbol{v}(\boldsymbol{z}) = \hat{\boldsymbol{f}}(\boldsymbol{z}) \; \forall \boldsymbol{z} \in \hat{\mathcal{Z}}^{\text{train}}$$

$$\sum_{B \in \mathcal{B}} \boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z})) = \sum_{B \in \mathcal{B}} \hat{\boldsymbol{f}}^{(B)}(\boldsymbol{z}_B) \; \forall \boldsymbol{z} \in \hat{\mathcal{Z}}^{\text{train}} .$$

$\boldsymbol{v} := \boldsymbol{f}^{-1} \circ \hat{\boldsymbol{f}}$ is a C^2-diffeomorphism

Derivative w.r.t $z_j \in J$ for some $J \in \mathscr{B}$

$$\sum_{B \in \mathcal{B}} \sum_{i \in B} D_i \boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z})) D_j \boldsymbol{v}_i(\boldsymbol{z}) = D_j \hat{\boldsymbol{f}}^{(J)}(\boldsymbol{z}_J) .$$

Derivative w.r.t $z_{j'} \in J'$ for some $J' \in \mathscr{B} / \{J\}$

$$\sum_{B \in \mathcal{B}} \sum_{i \in B} \left[D_i \boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z})) D^2_{j,j'} \boldsymbol{v}_i(\boldsymbol{z}) + \sum_{i' \in B} D^2_{i,i'} \boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z})) D_{j'} \boldsymbol{v}_{i'}(\boldsymbol{z}) D_j \boldsymbol{v}_i(\boldsymbol{z}) \right] = 0$$

Proof Sketch

$$\sum_{B\in\mathcal{B}}\sum_{i\in B} D_i \boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z}))D_j\boldsymbol{v}_i(\boldsymbol{z}) = D_j\hat{\boldsymbol{f}}^{(J)}(\boldsymbol{z}_J)\,.$$

Derivative w.r.t $z_{j'} \in J'$ for some $J' \in \mathscr{B}/\{J\}$

$$\sum_{B\in\mathcal{B}}\sum_{i\in B}\left[D_i\boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z}))D^2_{j,j'}\boldsymbol{v}_i(\boldsymbol{z}) + \sum_{i'\in B} D^2_{i,i'}\boldsymbol{f}^{(B)}(\boldsymbol{v}_B(\boldsymbol{z}))D_{j'}\boldsymbol{v}_{i'}(\boldsymbol{z})D_j\boldsymbol{v}_i(\boldsymbol{z})\right] = 0$$

Simplifying the expression

$$\boldsymbol{M}(\boldsymbol{z})\boldsymbol{w}(\boldsymbol{v}(\boldsymbol{z}),k) = 0\,.$$

$$\boldsymbol{w}(\boldsymbol{z},k) := ((D_i\boldsymbol{f}_k^{(B)}(\boldsymbol{z}_B))_{i\in B}, (D^2_{i,i}\boldsymbol{f}_k^{(B)}(\boldsymbol{z}_B))_{i\in B}, (D^2_{i,i'}\boldsymbol{f}_k^{(B)}(\boldsymbol{z}_B))_{(i,i')\in B^2_<})_{B\in\mathcal{B}}$$

$$\boldsymbol{M}(\boldsymbol{z}) := [[\vec{a}_i(\boldsymbol{z})]_{i\in B}, [\vec{b}_i(\boldsymbol{z})]_{i\in B}, [\vec{c}_{i,i'}(\boldsymbol{z})]_{(i,i')\in B^2_<}]_{B\in\mathcal{B}}\,,$$

Proof Sketch

$$\boldsymbol{M}(\boldsymbol{z})\boldsymbol{w}(\boldsymbol{v}(\boldsymbol{z}), k) = 0\,.$$

$$\boldsymbol{w}(\boldsymbol{z}, k) := ((D_i \boldsymbol{f}_k^{(B)}(\boldsymbol{z}_B))_{i\in B}, (D^2_{i,i}\boldsymbol{f}_k^{(B)}(\boldsymbol{z}_B))_{i\in B}, (D^2_{i,i'}\boldsymbol{f}_k^{(B)}(\boldsymbol{z}_B))_{(i,i')\in B^2_<})_{B\in\mathcal{B}}$$

$$\boldsymbol{M}(\boldsymbol{z}) := [[\vec{a}_i(\boldsymbol{z})]_{i\in B}, [\vec{b}_i(\boldsymbol{z})]_{i\in B}, [\vec{c}_{i,i'}(\boldsymbol{z})]_{(i,i')\in B^2_<}]_{B\in\mathcal{B}}\,,$$

$$\Big\downarrow \qquad W(v(z))^T = [w(v(z),1), \cdots, w(v(z), d_x)]$$

$$\boldsymbol{W}(\boldsymbol{v}(\boldsymbol{z}))\boldsymbol{M}(\boldsymbol{z})^\top = 0$$

Assumption of sufficient non-linearity on f implies $W(v(z))$ has full-column rank

$$\Big\downarrow$$

$$\boldsymbol{M}(\boldsymbol{z})^\top = 0$$

Proof Sketch

$$\boldsymbol{W}(\boldsymbol{v}(\boldsymbol{z}))\boldsymbol{M}(\boldsymbol{z})^\top = 0$$

Assumption of sufficient non-linearity on f implies $W(v(z))$ has full-column rank

$$\boldsymbol{M}(\boldsymbol{z}) := [[\vec{a}_i(\boldsymbol{z})]_{i\in B}, [\vec{b}_i(\boldsymbol{z})]_{i\in B}, [\vec{c}_{i,i'}(\boldsymbol{z})]_{(i,i')\in B^2_<}]_{B\in\mathcal{B}}$$

$$\boldsymbol{M}(\boldsymbol{z})^\top = 0$$

$$\forall i \in \{1,\dots,d_z\}, \vec{b}_i(\boldsymbol{z}) = 0,$$

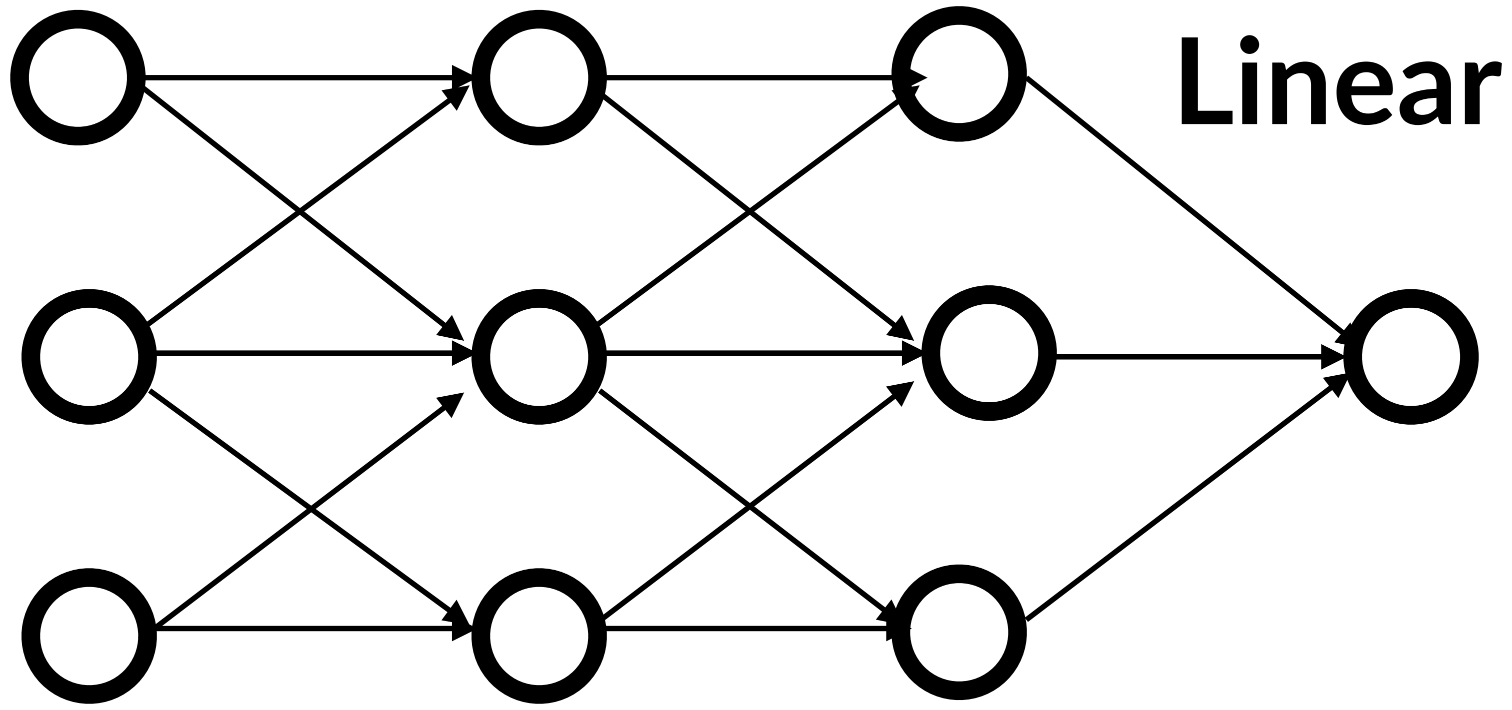
$$\forall i \in \{1,\dots,d_z\}, \forall (j,j') \in S^c, D_j\boldsymbol{v}_i(\boldsymbol{z})D_{j'}\boldsymbol{v}_i(\boldsymbol{z}) = 0$$

The final equality can be implied further to show $Dv(z)$ is $\mathscr{B}$-block permutation matrix

Independence Constrained ERM

ERM

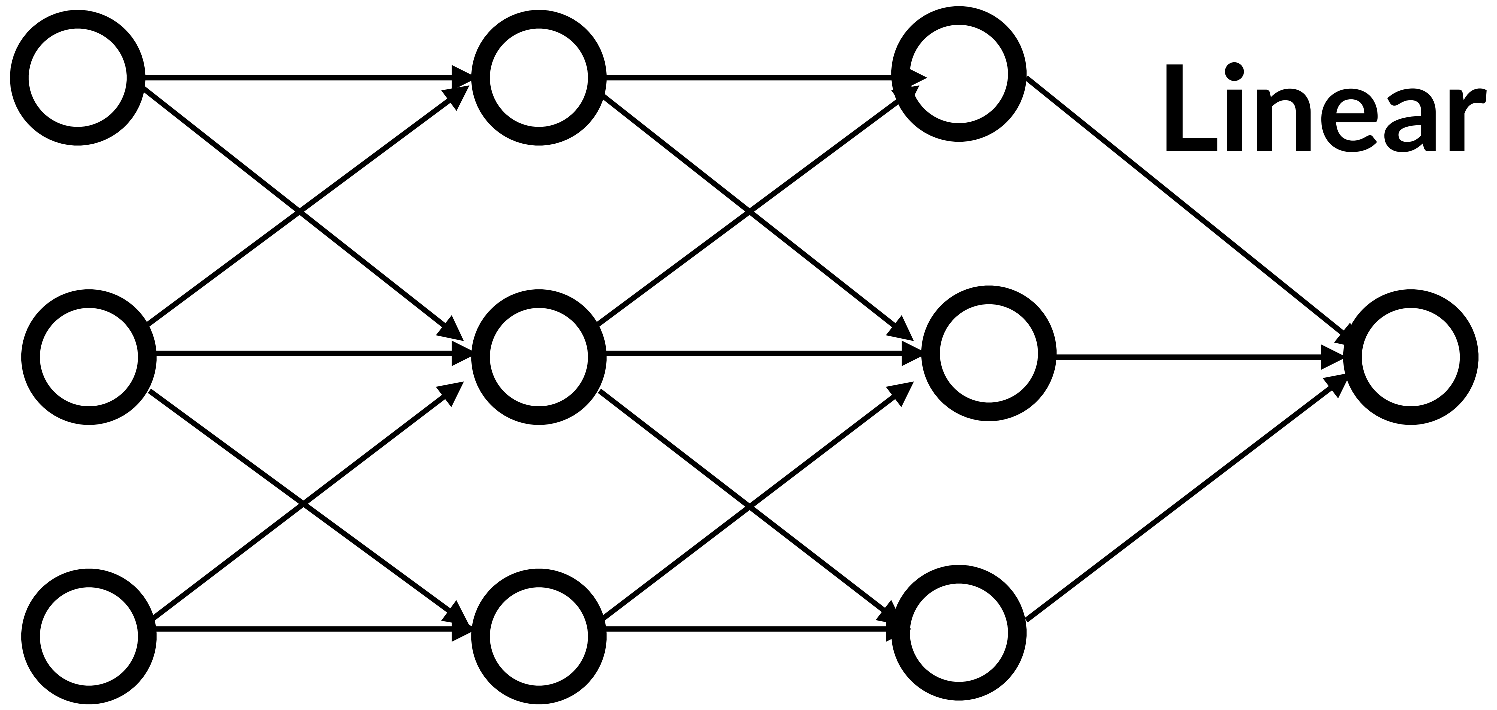
Representation Network



Linear Layer

$$\min_{W \in \mathbb{R}^{d \times k}, \Phi \in \mathcal{H}_\Phi} \sum_{i=1}^{N} \ell\Big(W \circ \Phi(x_i), y_i\Big)$$

IC-ERM



Linear Layer

$$\min_{W \in \mathbb{R}^{d \times k}, \Phi} \sum_{i=1}^{N} \ell\Big(W \circ \Phi(x_i), y_i\Big)$$

$\Phi(X)$ **is i.i.d.**

Representation Network