Harnessing Geometric Signatures in Causal Representation Learning

Yixin Wang (University of Michigan)

Joint work with Michael Jordan (UC Berkeley), Yoshua Bengio (Mila), Kartik Ahuja (Meta AI), Divyat Mahajan (Mila), Amin Mansouri (EPFL)

How can I merge lanes in NYC heavy traffic?

How can I merge lanes in NYC heavy traffic?

How can I merge lanes in NYC heavy traffic?

How can I merge lanes in NYC heavy traffic?

How can I merge lanes in NYC heavy traffic?

How can I merge lanes in NYC heavy traffic?

How can I merge lanes in NYC heavy traffic?

How can I merge lanes in NYC heavy traffic?

How can I merge lanes in NYC heavy traffic?

Causal Inference

Structured data

	Car make	Car color	Gesture	Bumper Sticker	Dog	Dist. to Car	Succesful Merge
1	Toyota	Red	1	1	1	3	1
2	Ford	Blue	0	1	3	1.5	0
3	Honda	Yellow	1	0	2	2.3	0
4	Tesla	Red	0	1	5	4.6	1

Causal Inference

Unstructured data

Causal Inference with Unstructured Data

- First step: Causal representation learning

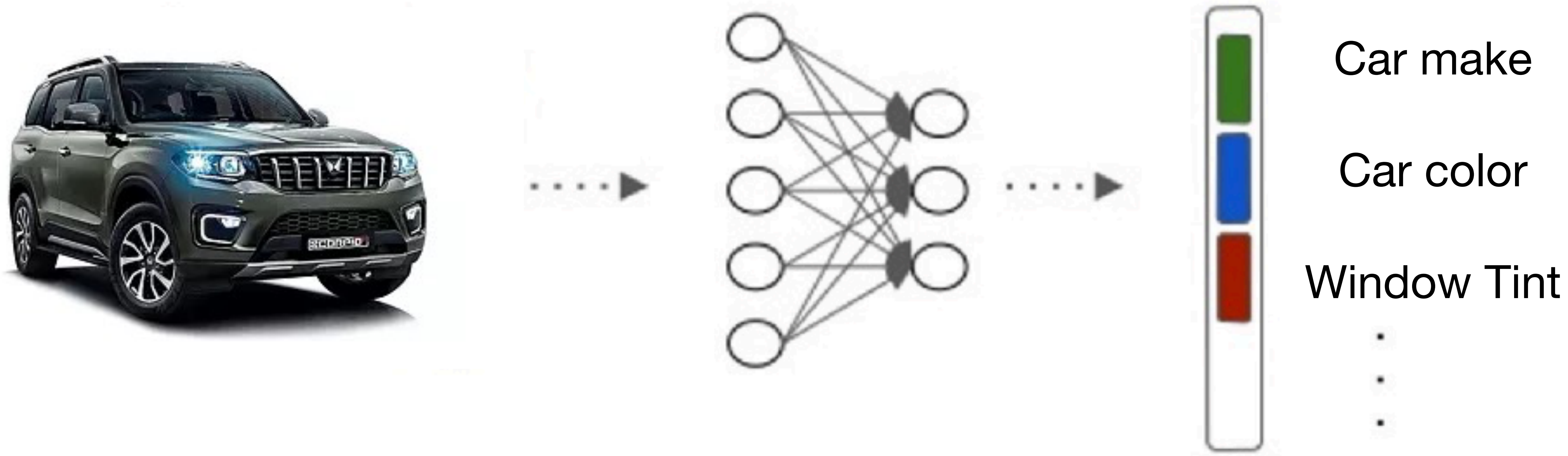
157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206

	Car make	Car color	Gesture	Bumper Sticker	Dog	Dist. to Car	Merge
1	Toyota	Red	1	1	1	3	1
2	Ford	Blue	0	1	3	1.5	0
3	Honda	Yellow	1	0	2	2.3	0
4	Tesla	Red	0	1	5	4.6	1

Unsupervised Causal Representation Learning

Identify independently controllable causal factors

Unlabelled car images



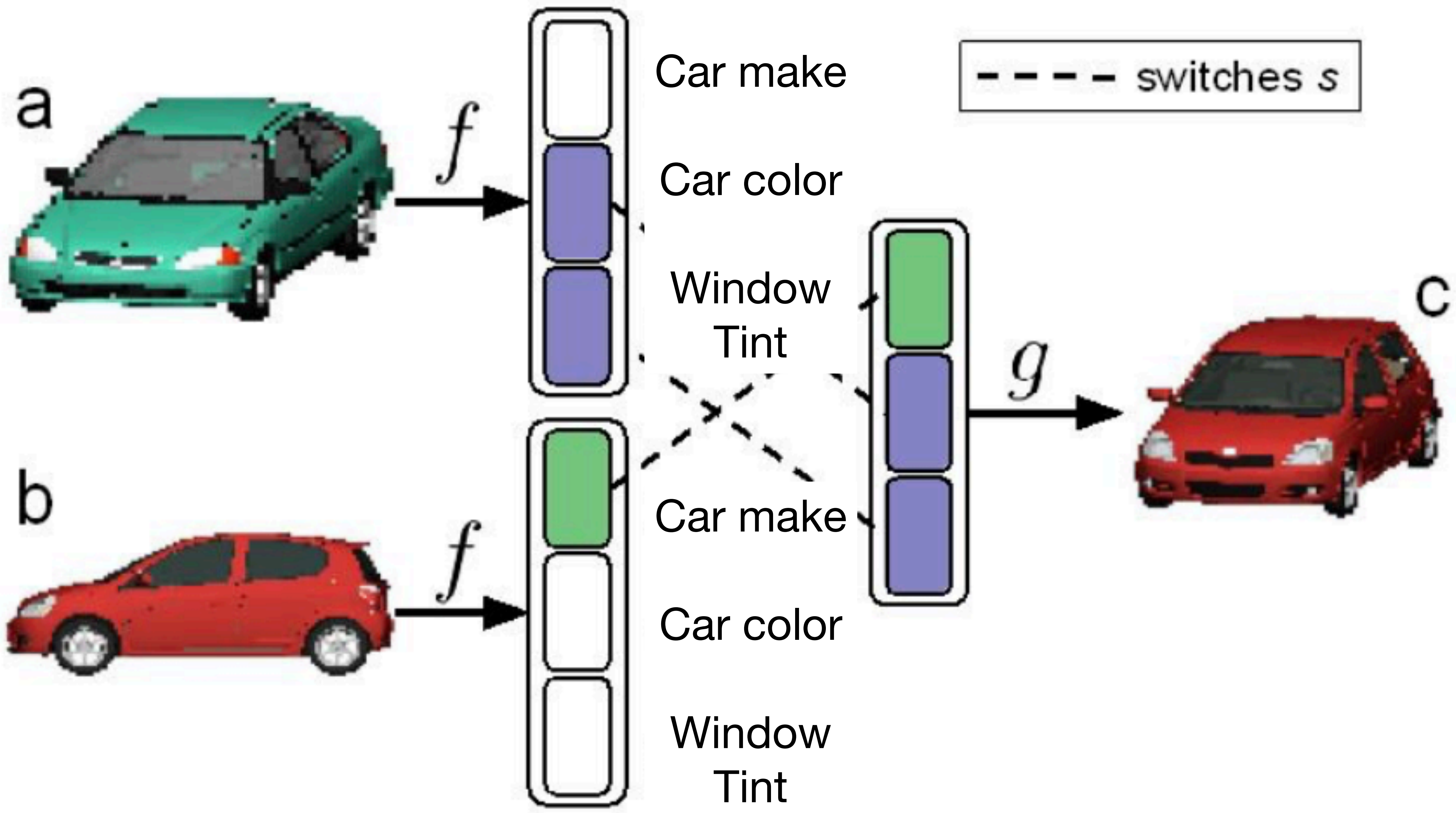
(Independently controllable) causal factors

✓ (Car make, Car color)

✗ (Car make + Car color, Car make - Car color)

Why Unsupervised Causal Representation Learning?

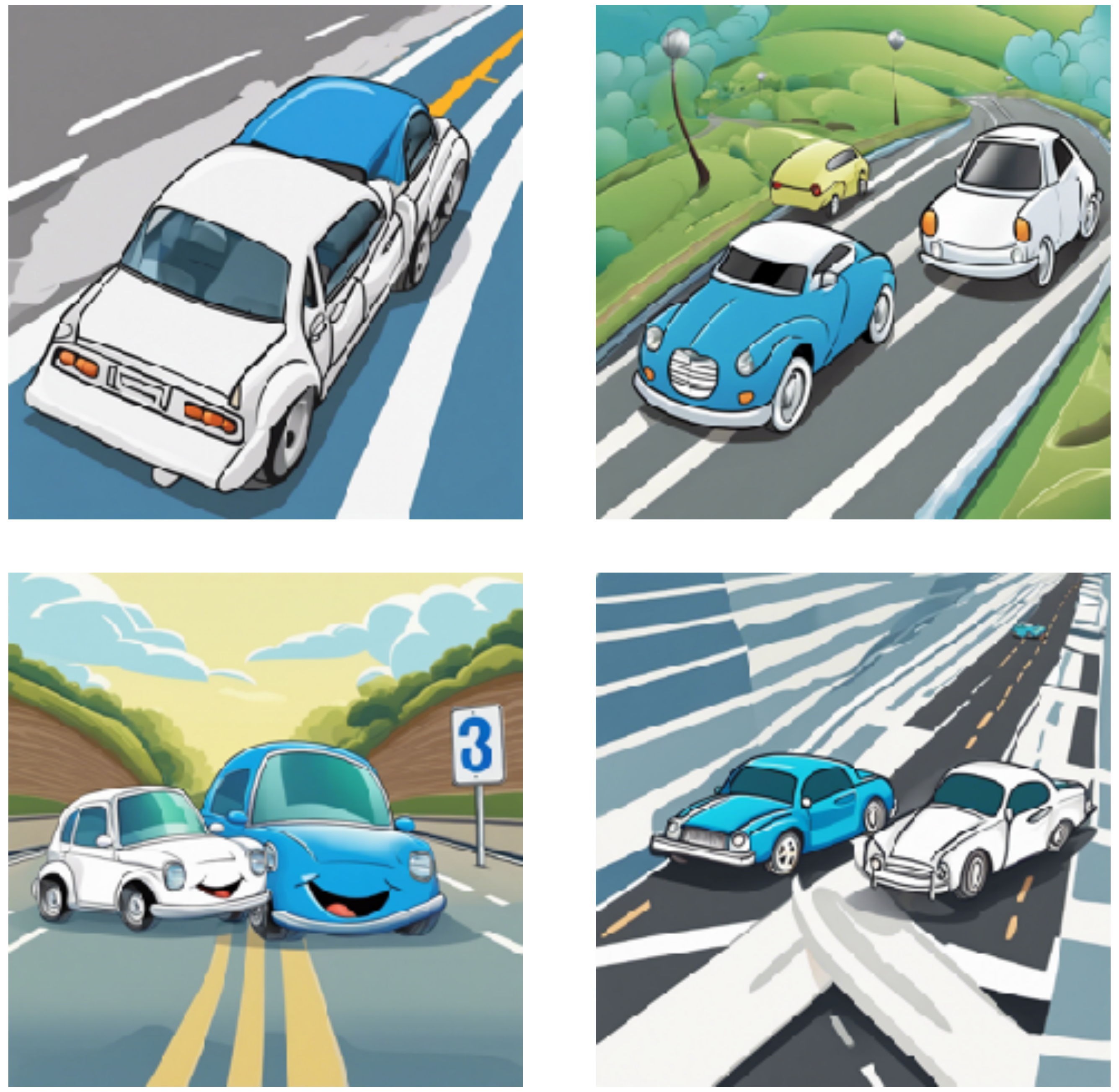
Compositional Generalization



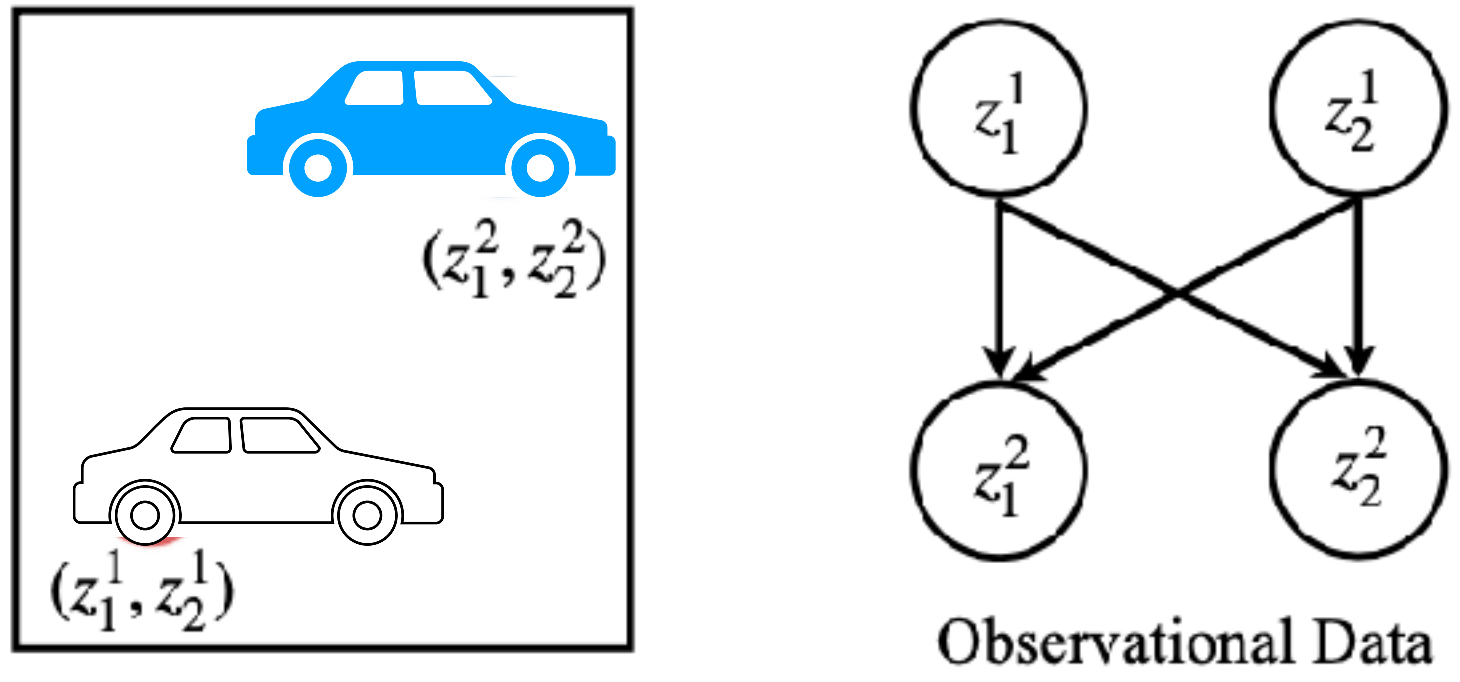
[DeepAI 2021]

Unsupervised Causal Representation Learning

Identify latent causal factors and their causal graphs



Unlabelled car images:
white car/blue car

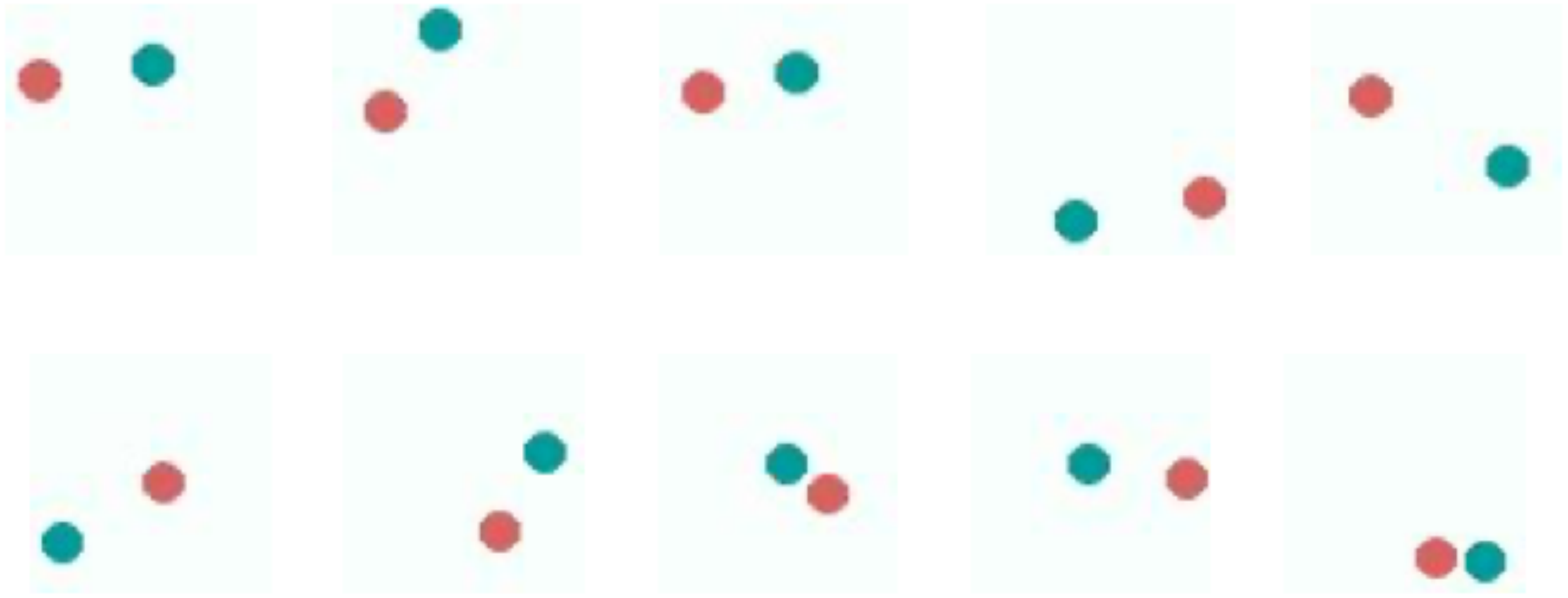


Latent causal factors

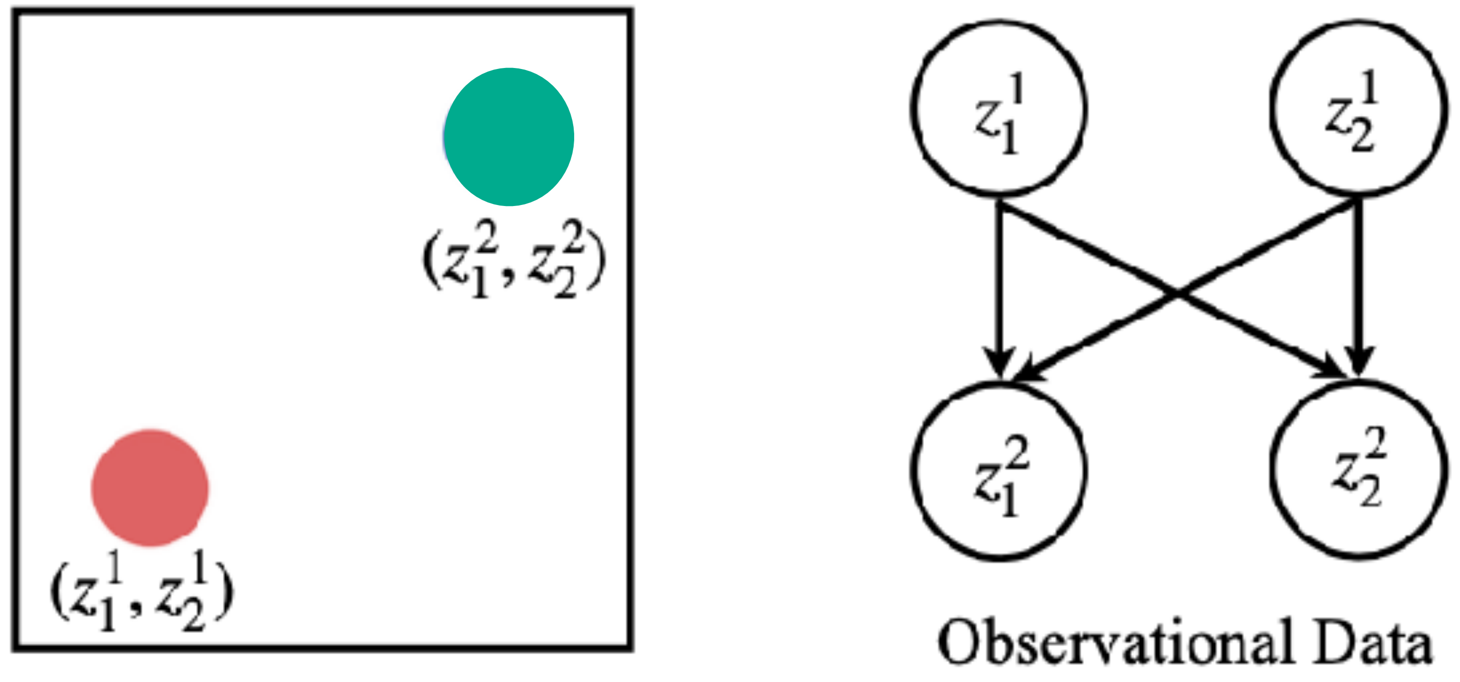
[Ahuja+ 2022]

Unsupervised Causal Representation Learning

Identify latent causal factors and their causal graphs



Unlabelled ball images

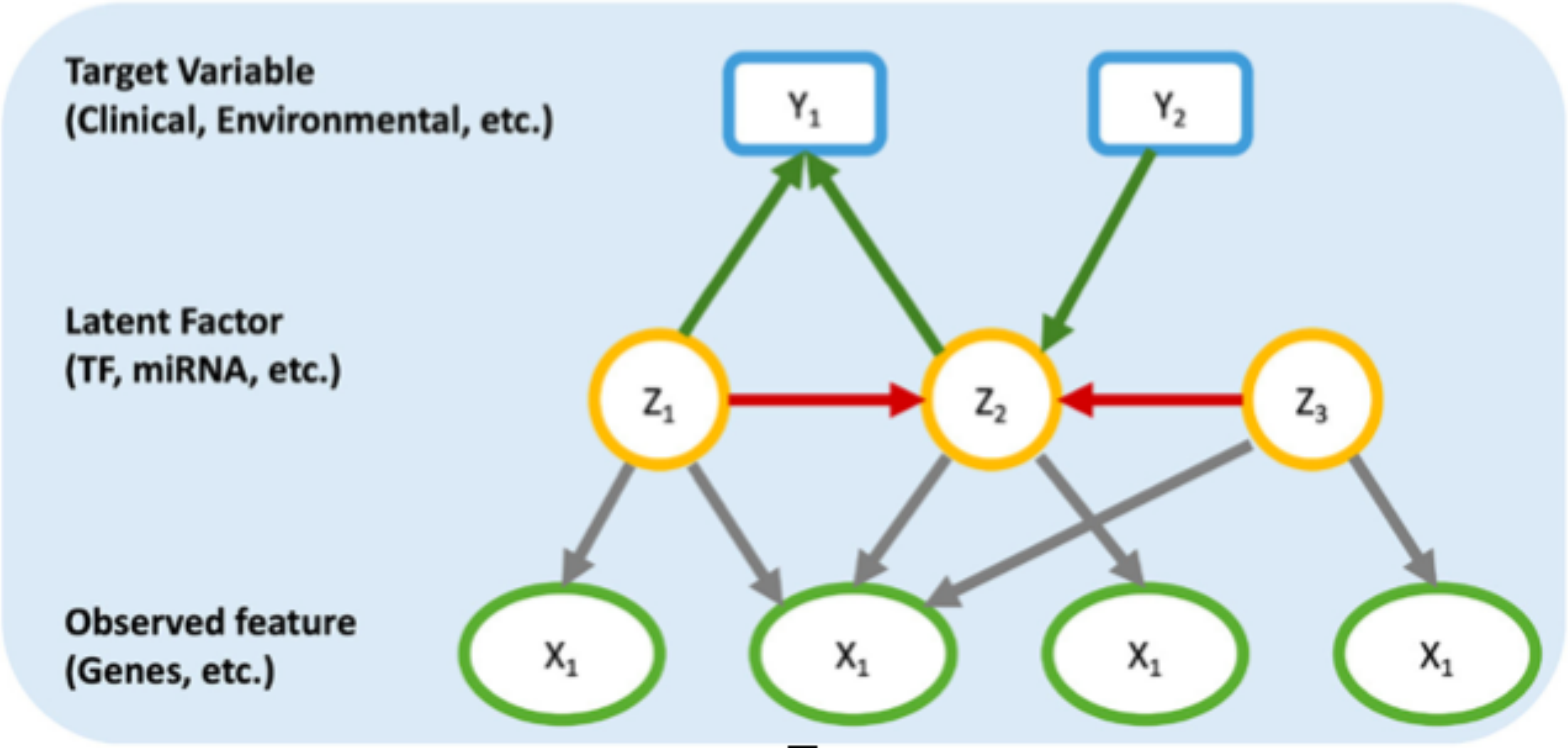


Latent causal factors

[Ahuja+ 2022]

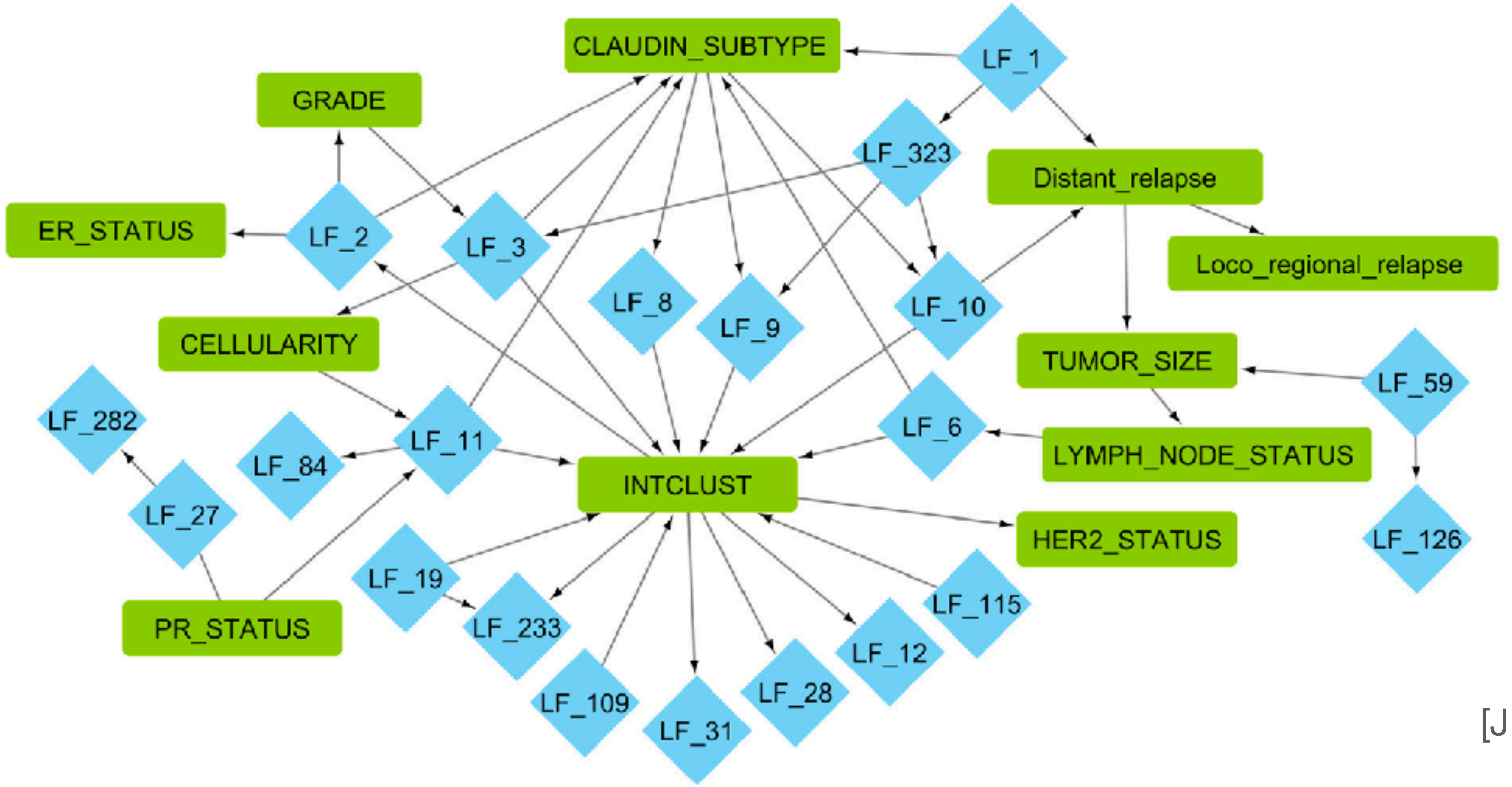
Why Unsupervised Causal Representation Learning?

Understand latent drivers and mechanisms in science



[Jia+ 2022]

Causal representation learning

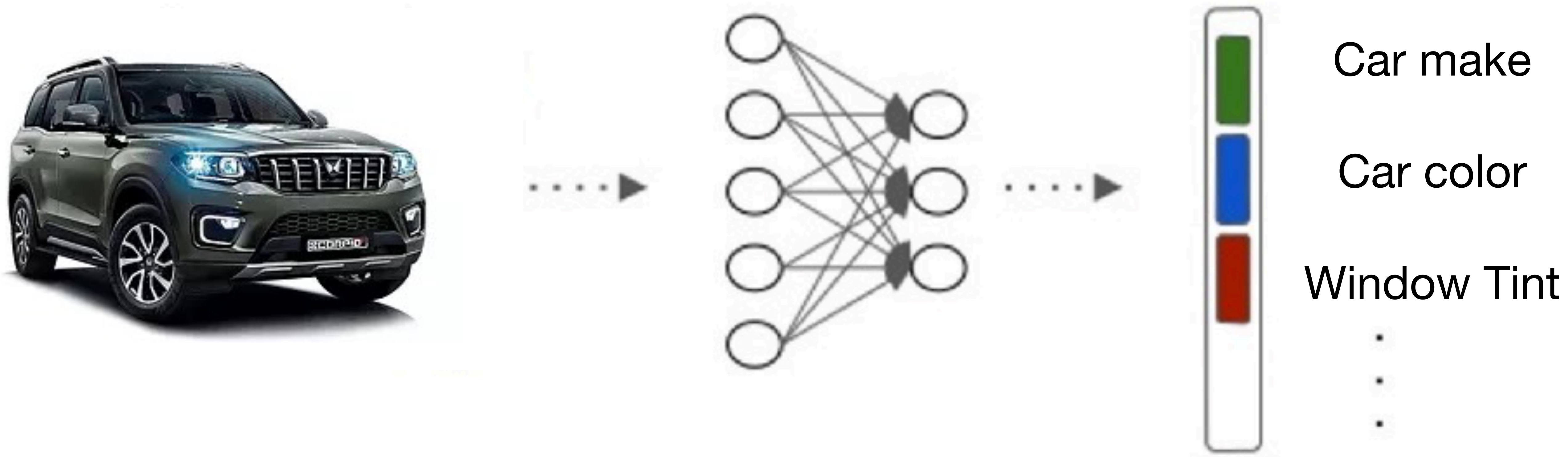


[Jia+ 2022]

Learn causal representations → Interpret causal factors → Design effective interventions

Why can’t we just fit a latent variable model?

Why can't we just fit a latent variable model?

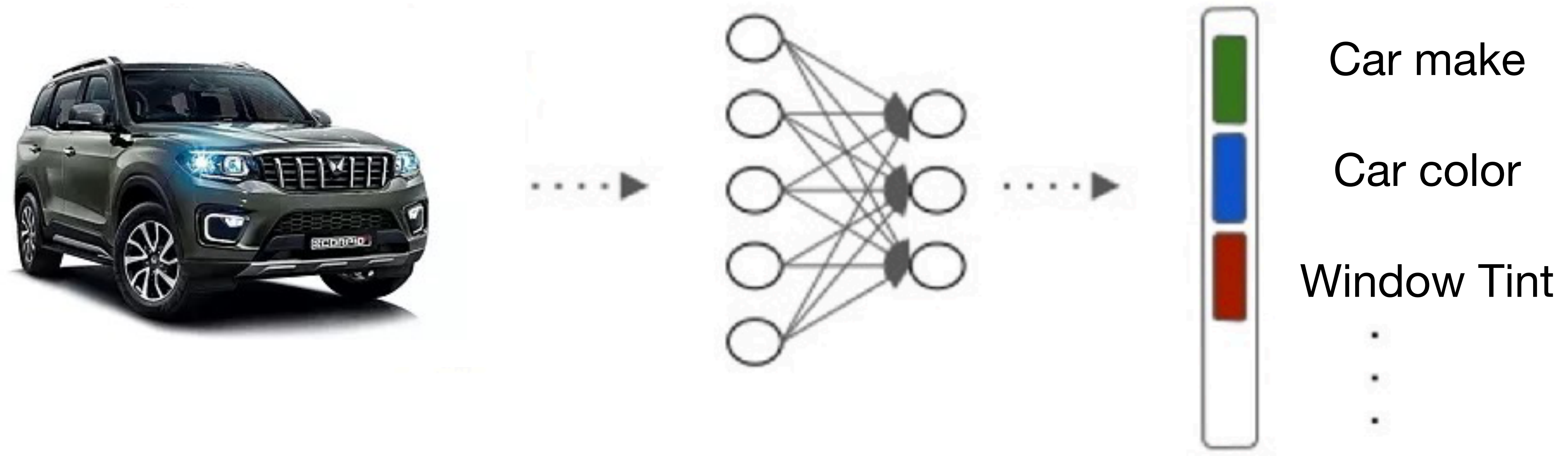


$$x_i \sim N(f(z_i), \sigma^2)$$
$$z_i \sim N(0, I)$$

- Fit a nonlinear factor model (variational autoencoder) $x_i \sim N(f(z_i), \sigma^2)$ to the data $x_1, \ldots, x_n$
- Obtain a representation function $\hat{z}_i = g(x_i)$ for all i.

Why can't we just fit a flexible latent variable model?

The non-identifiability of representations from flexible latent variable models

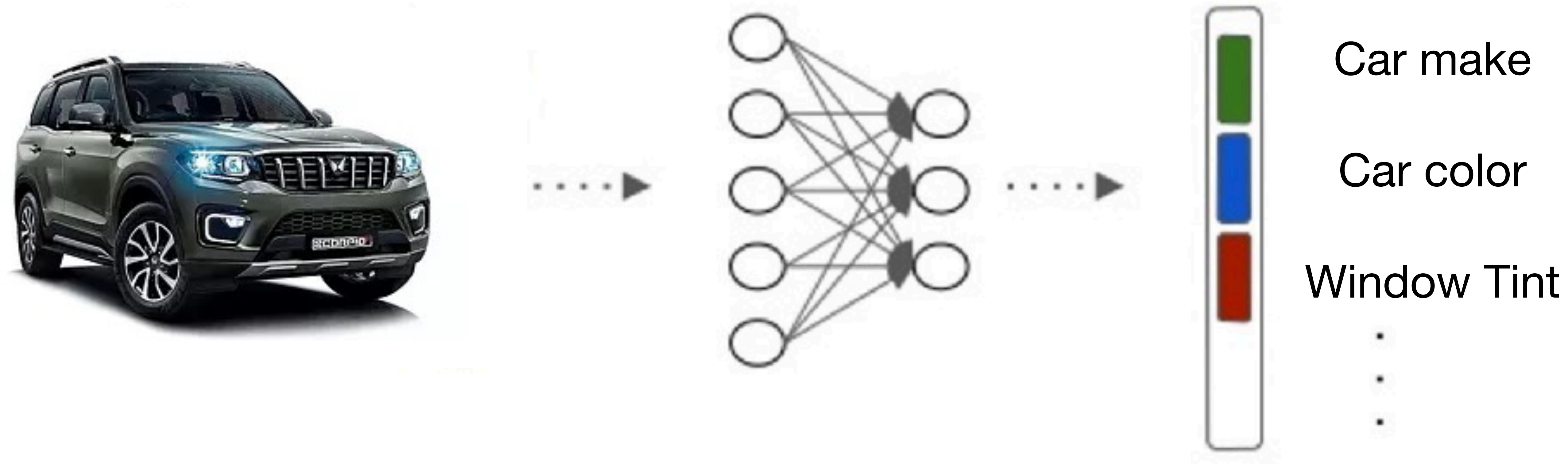


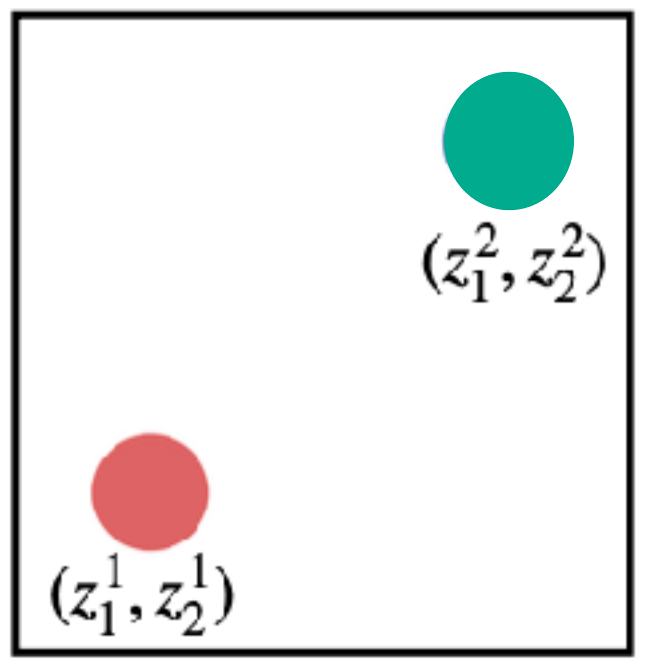
$$x_i \sim N(f(z_i), \sigma^2)$$
$$z_i \sim N(0, I)$$

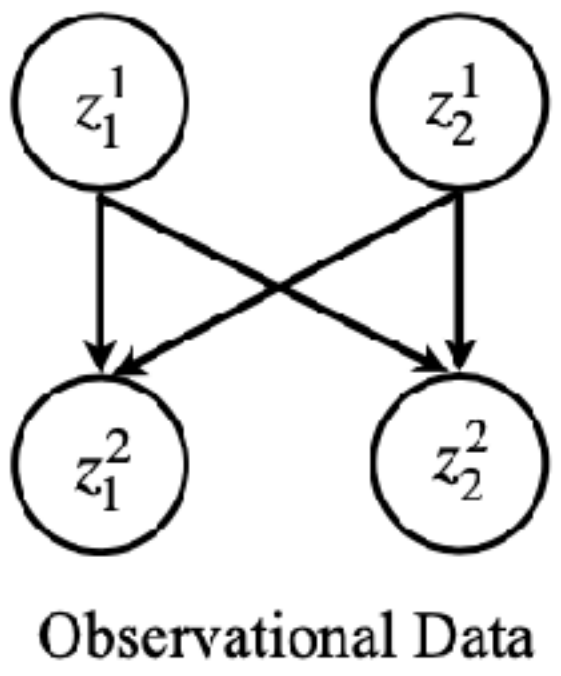
- But the latent variable model can return **multiple** representation functions that are **equally valid**
  - Given the same dataset, $x_1, \ldots, x_n$, fit the same model twice
  - One gives $\hat{z}_i = g_1(x_i)$ for all i, and the other gives $\hat{z}_i = g_2(x_i)$ for all i.

Why can't we just fit a flexible latent variable model?

When the causal factors are not identifiable (aka underdetermined, non-unique)

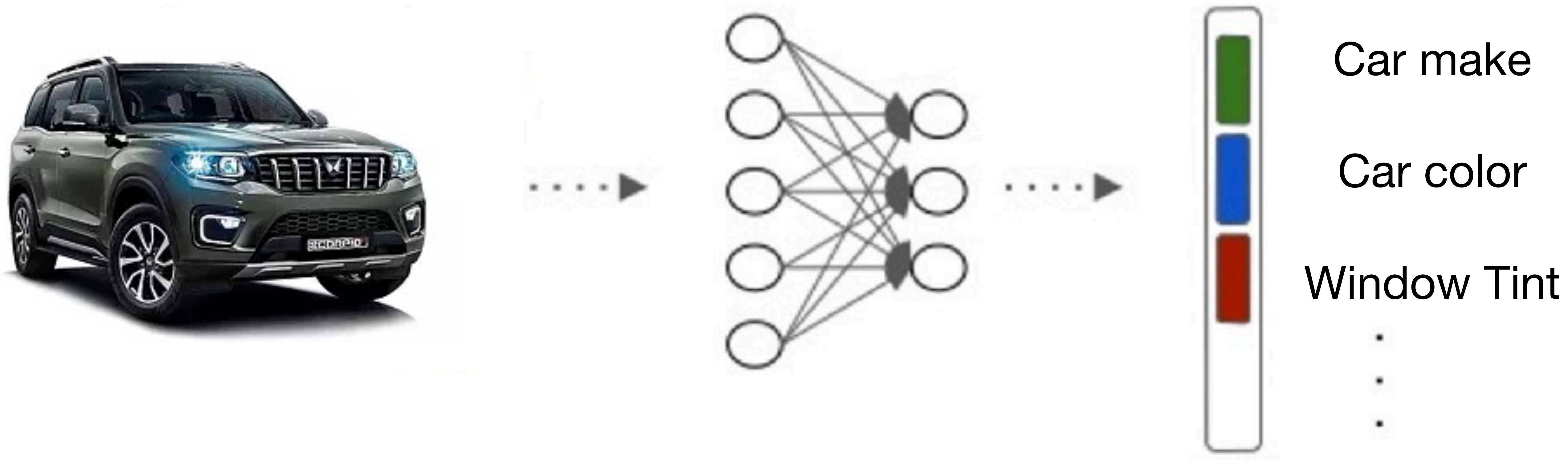






- Challenge the **interpretation**: e.g. $\hat{z}_i = (x_{i3}, x_{i2})$ vs $\hat{z}_i = (x_{i3} + x_{i2}, x_{i3} - x_{i2})$
- Learning the **causal graph** among non-identified causal factors no longer makes sense
- Prevent the downstream **design of targeted interventions** for latent causal factors

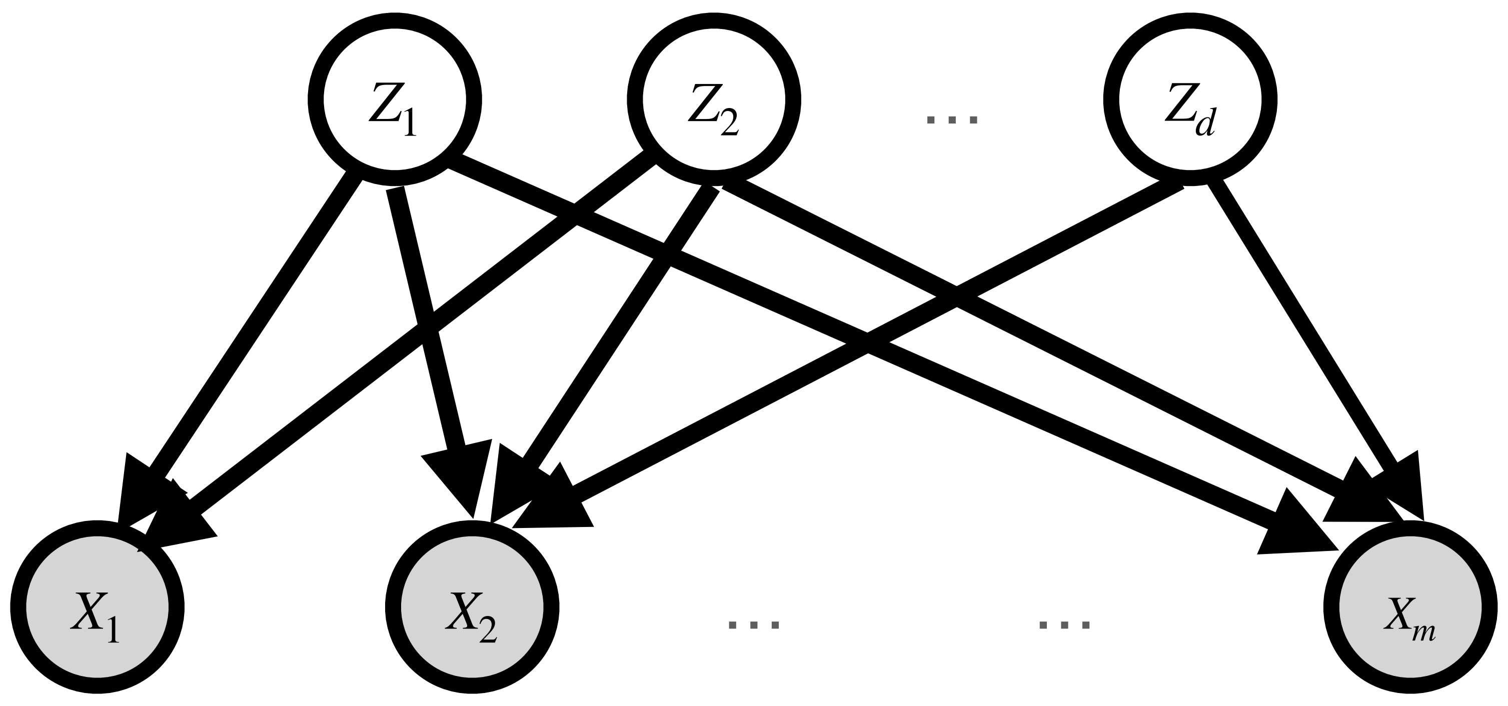
Identification of latent causal factors



- **Identify** latent causal factors
  - Suppose the data $x_{1:n}$ is generated by some **true** latent causal factors $x_i = g(z_i)$ for all i
  - Provide an algorithm that takes in $x_{1:n}$ and output $\hat{g}, \hat{z}_i$ such that $\hat{g} = g, z_i = \hat{z}_i$ **for all i**

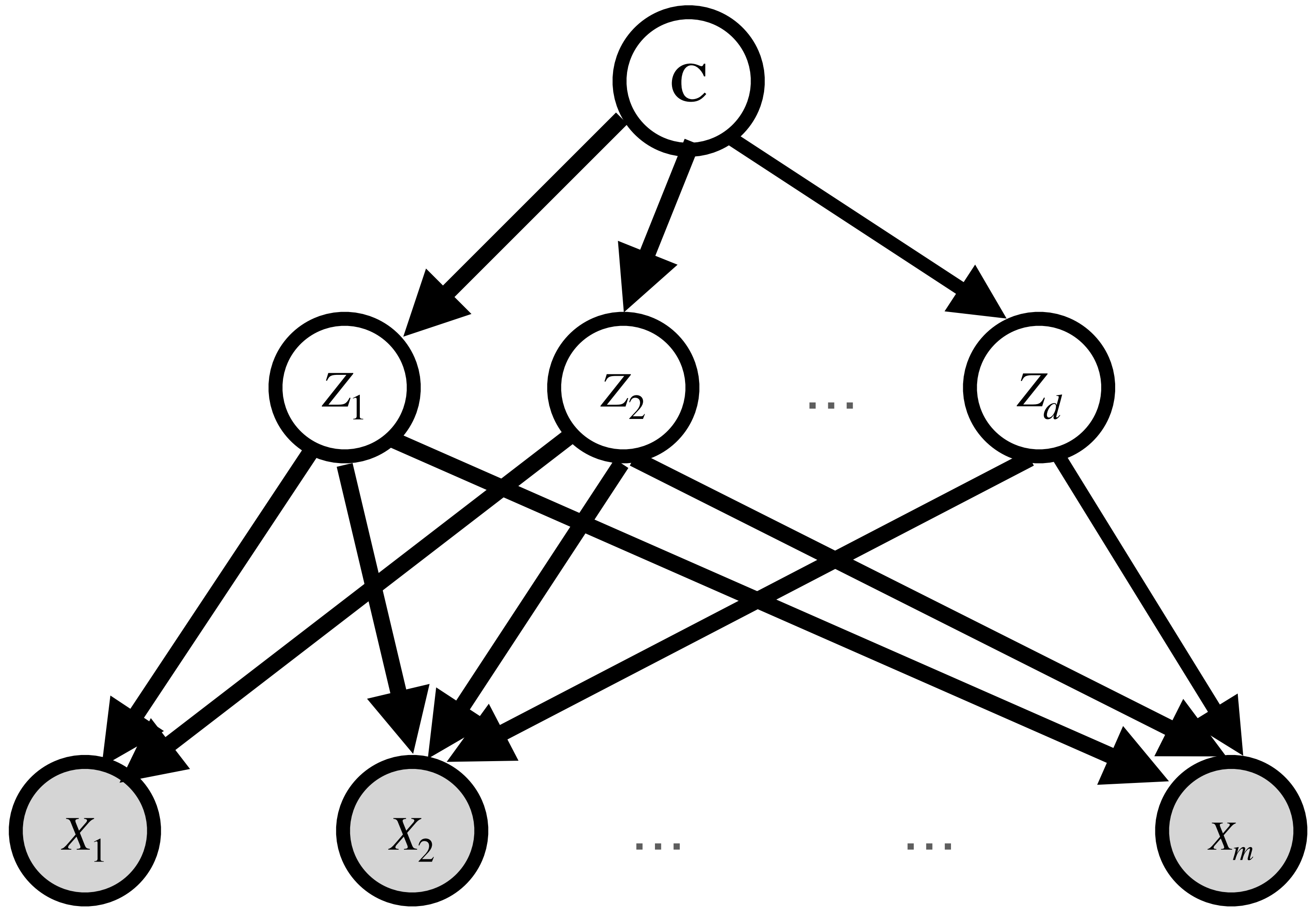
How can we identify latent causal factors?

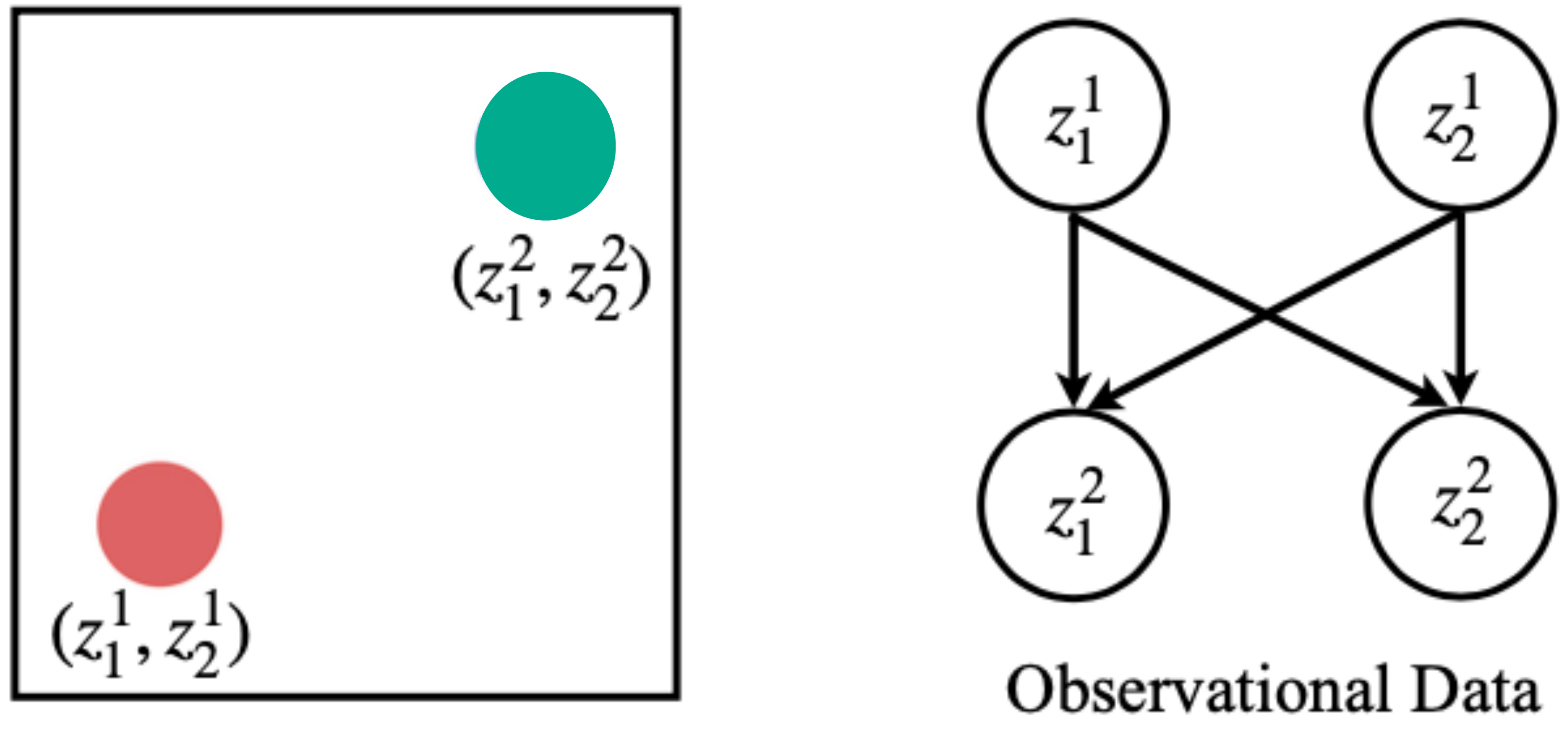
Predominant: Establish identifiability for flexible latent variable models



- Key assumption: **Independent** latent factors
  - **Independent component analysis**
    - Independent latents + non-Gaussianity
  - **Variational autoencoder**
    - Independent latents + Auxiliary variable
    - Independent latents + Gaussian mixture prior (w/o auxiliary)

But latent causal factors are **rarely** statistically independent…

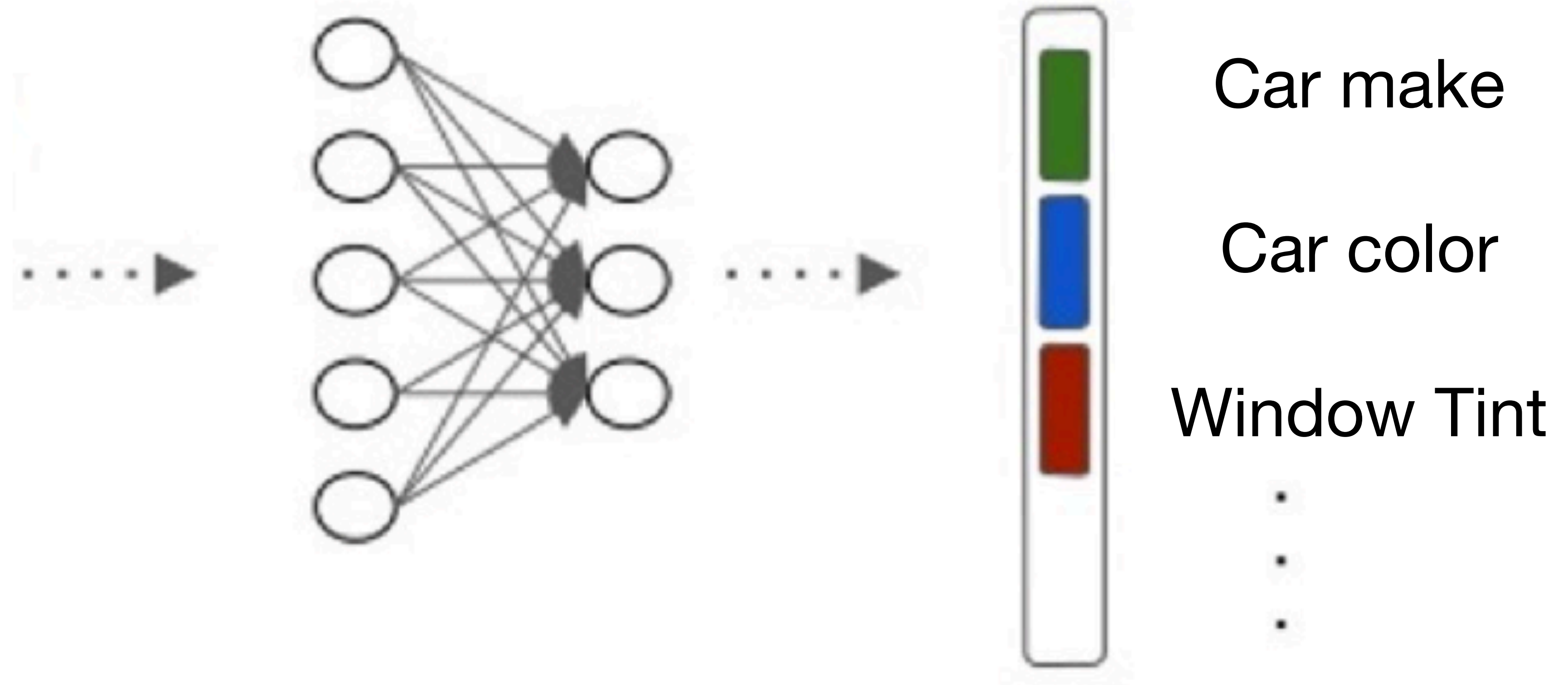


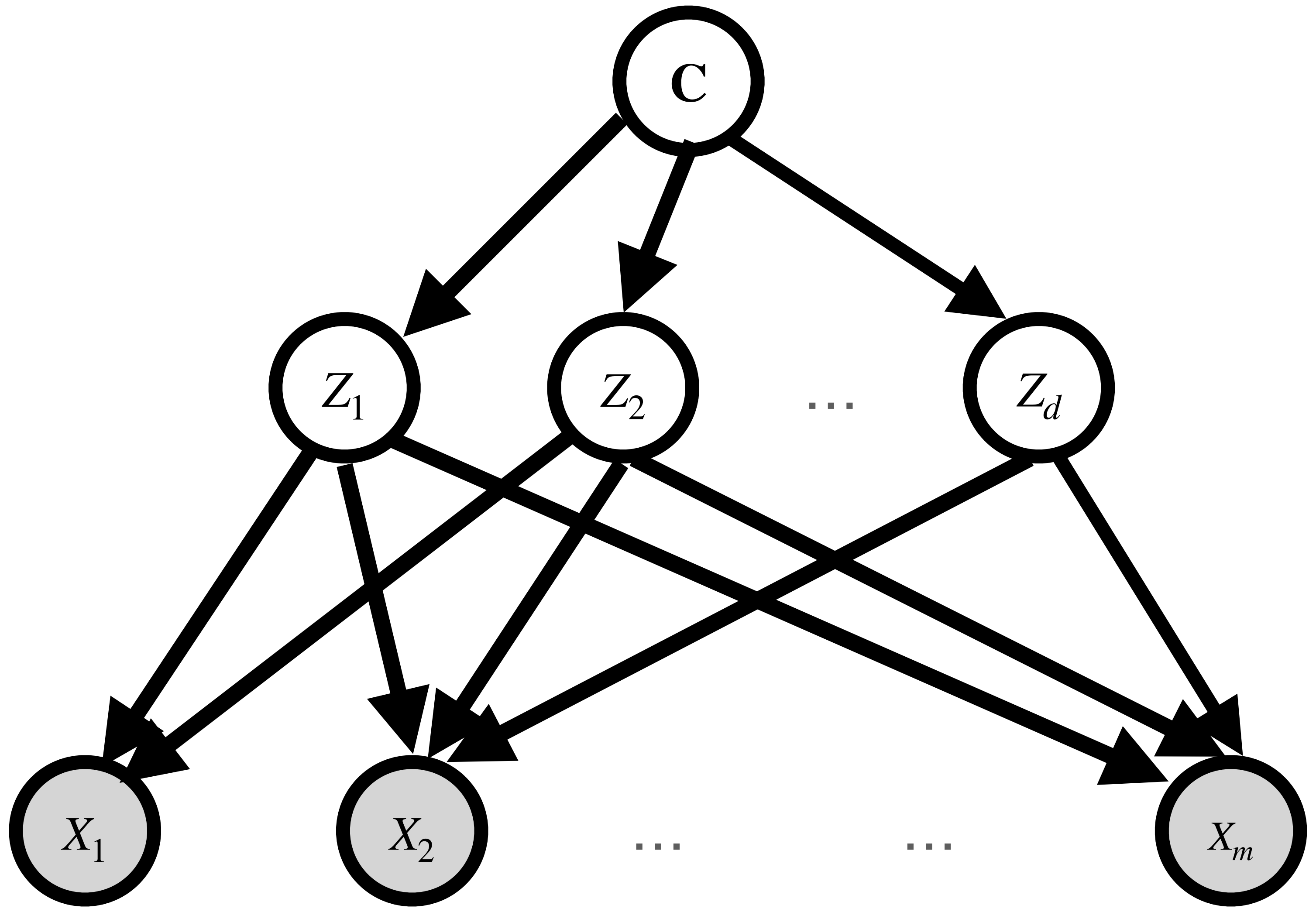


- They are **correlated**, or even **causally connected**.
- What assumptions can help identify correlated latent causal factors?

Geometric signatures: Independence of support

Simplest case: Correlated latent causal factors



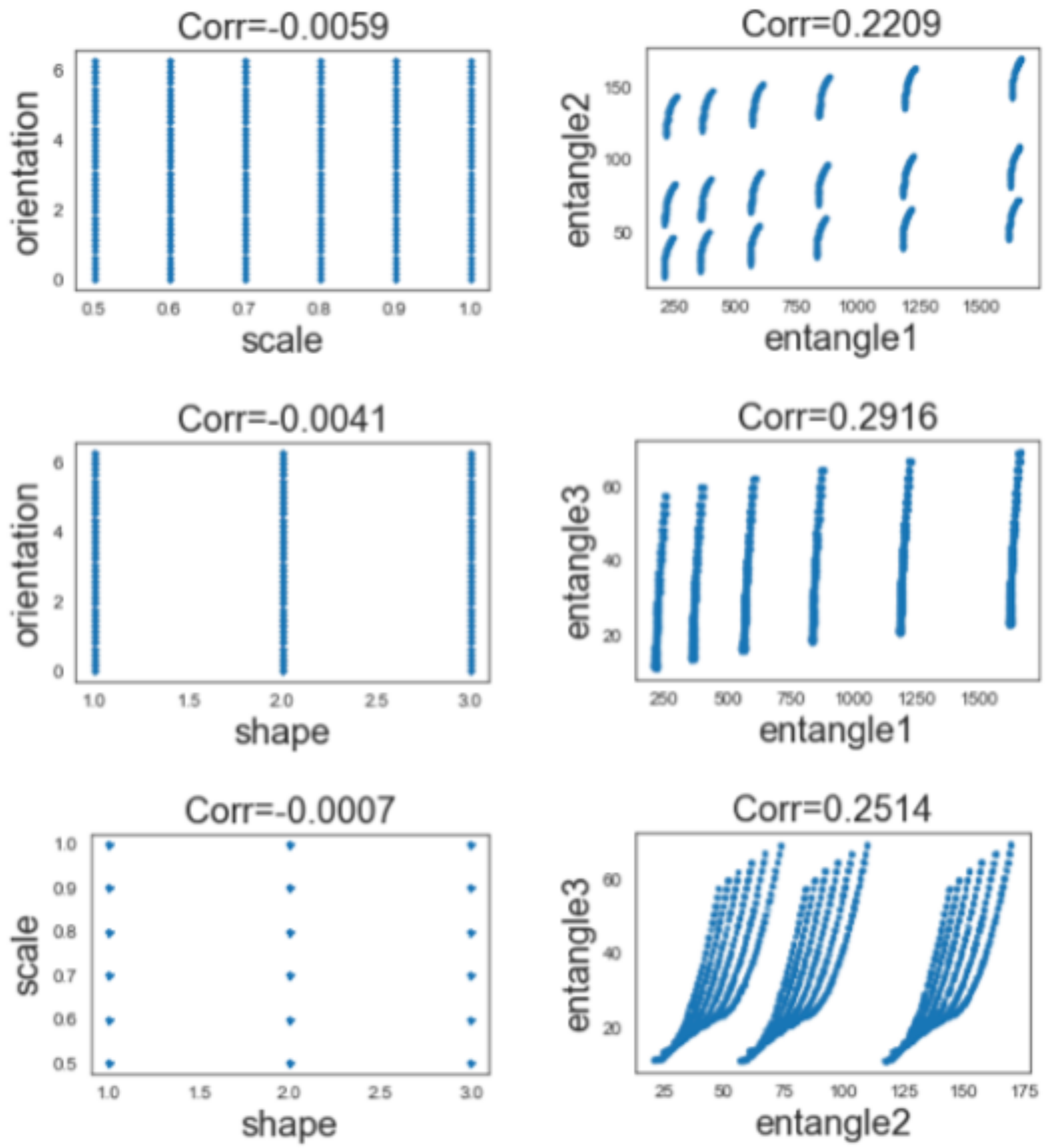


- Goal: identify the correlated latent factors $Z_1, \ldots, Z_d$
- The latent factors are **correlated** but **not causally connected**.

Correlated latent causal factors

Independent **support** condition

- Key observation: Latent causal factors often have **independent support** $\mathrm{supp}(Z_1, \ldots, Z_d) = \mathrm{supp}(Z_1) \times \cdots \times \mathrm{supp}(Z_d)$

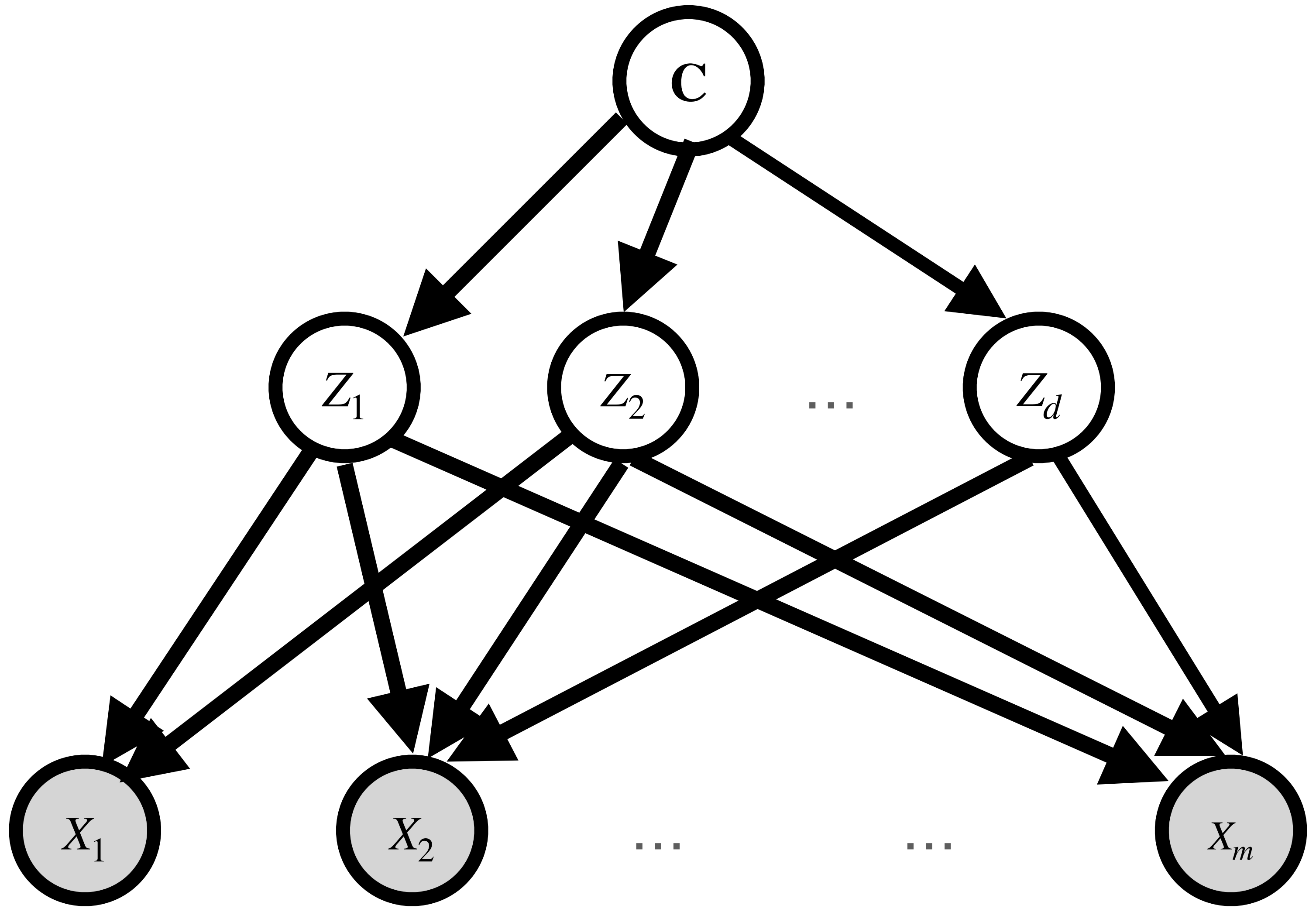


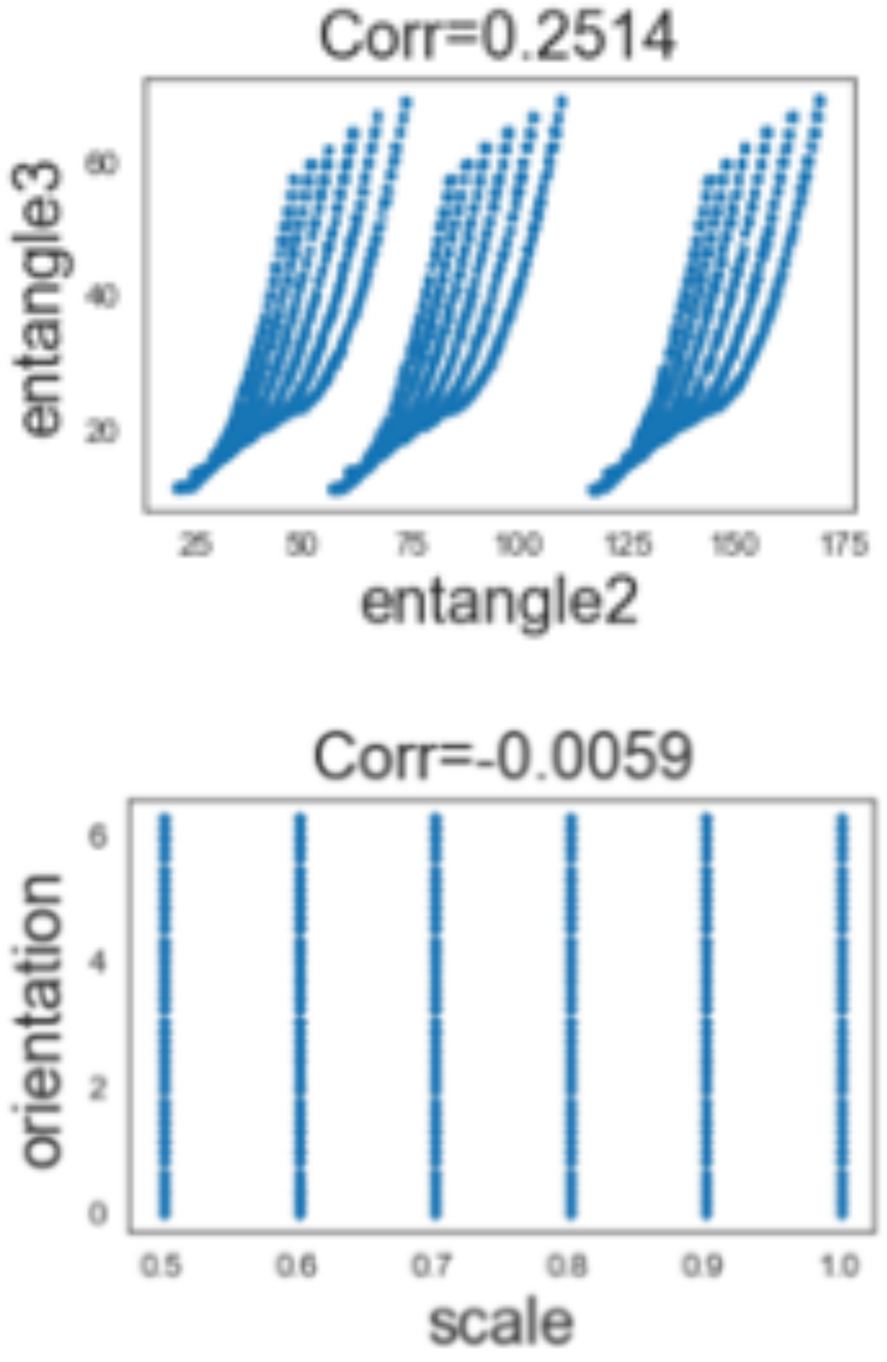
Independent support

Non-Independent support

Correlated latent causal factors

Independent **support** condition





- **Theorem** (W. & Jordan, 2021) Under a **positivity** condition, **no causal connections** among the latent causal factors implies that they must have **independent support**.

Correlated latent causal factors

Measure the Independence of support

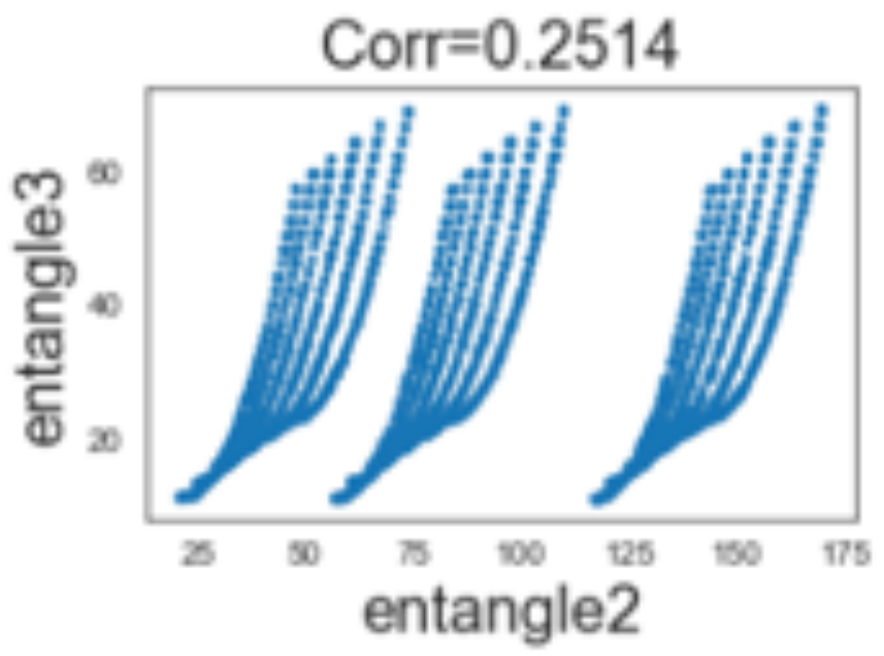
- **Independence-of-support-score (IOSS): A disentanglement metric**

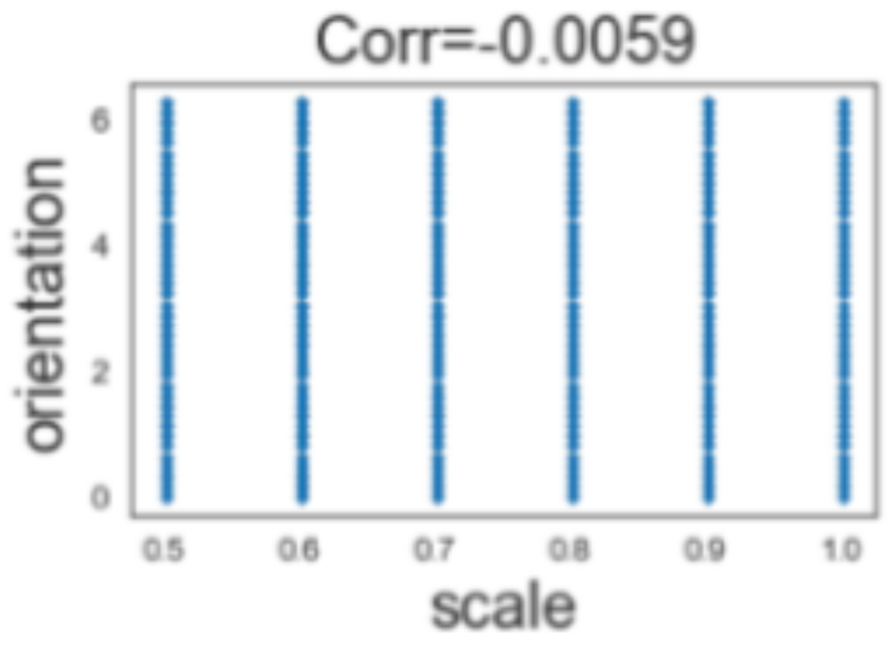
$$\mathrm{IOSS} \triangleq d_H(\mathrm{supp}(\bar{G}_1, \ldots, \bar{G}_d), \mathrm{supp}(\bar{G}_1) \times \cdots \times \mathrm{supp}(\bar{G}_d))$$

where $\bar{G}_j = (G_j - \inf G_j)/(\sup G_j - \inf G_j)$ is the standardized G_j and

$$d_\mathrm{H}(X, Y) \triangleq \max \left\{ \sup_{x \in X} \inf_{y \in Y} d(x, y), \sup_{y \in Y} \inf_{x \in X} d(x, y) \right\}$$

is the Hausdorff distance.





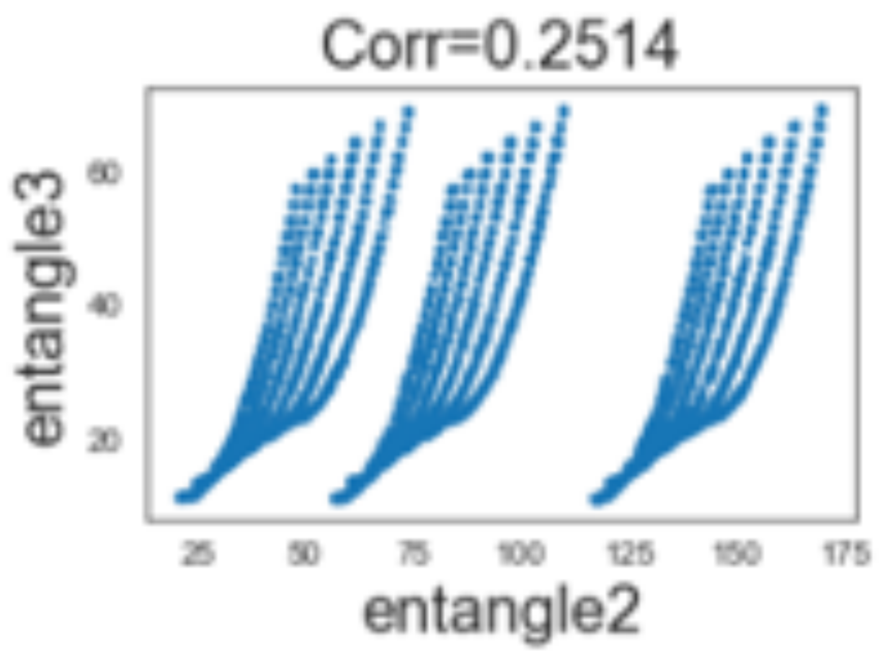
Correlated latent causal factors

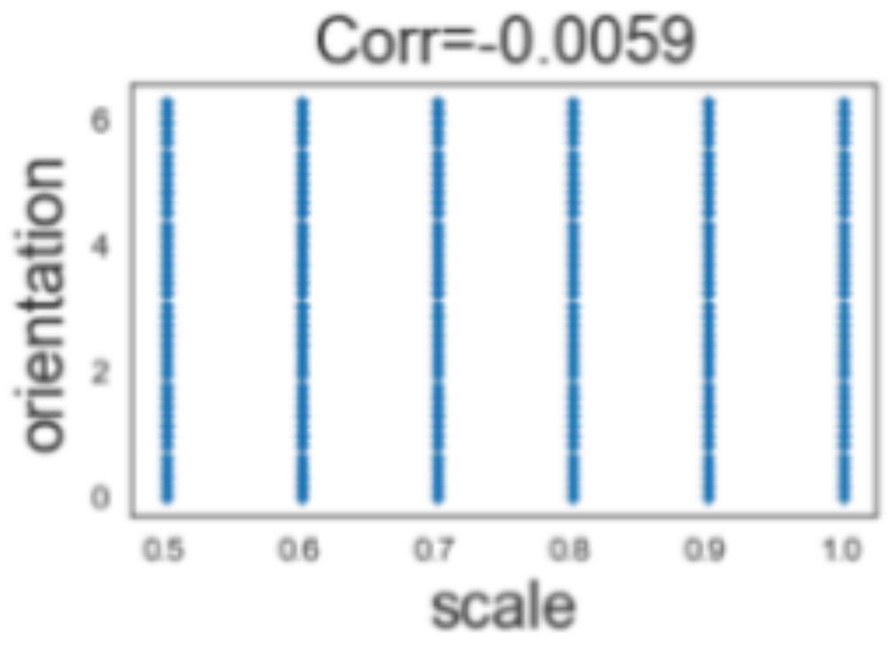
How to enforce independent support?

- **Algorithm :**
  - Fit latent variable model with **IOSS penalty**,

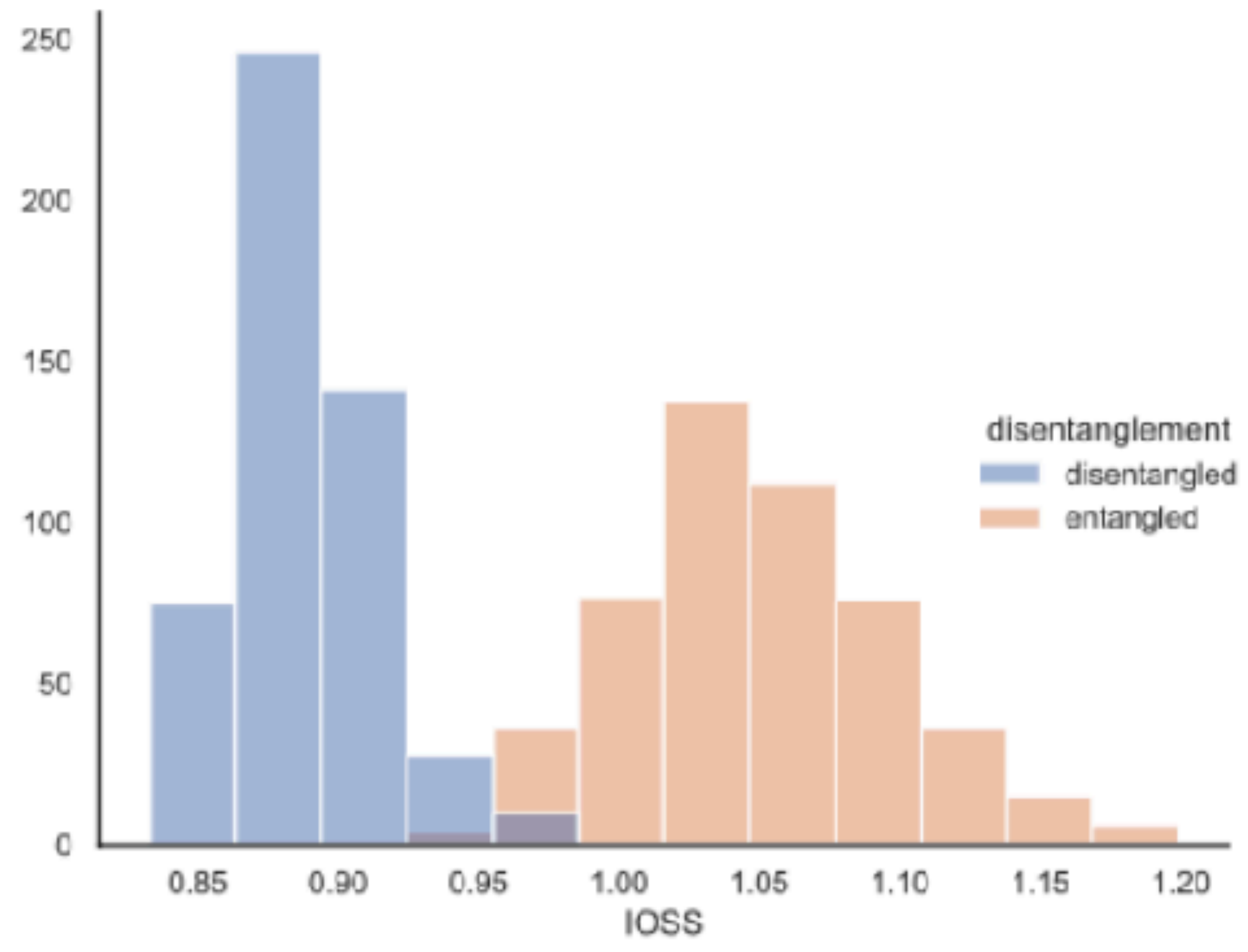
    $L + \lambda \cdot \text{IOSS}$

  - $\text{IOSS} \triangleq d_H(\text{supp}(\bar{Z}_1, \ldots, \bar{Z}_d), \text{supp}(\bar{Z}_1) \times \cdots \times \text{supp}(\bar{Z}_d))$

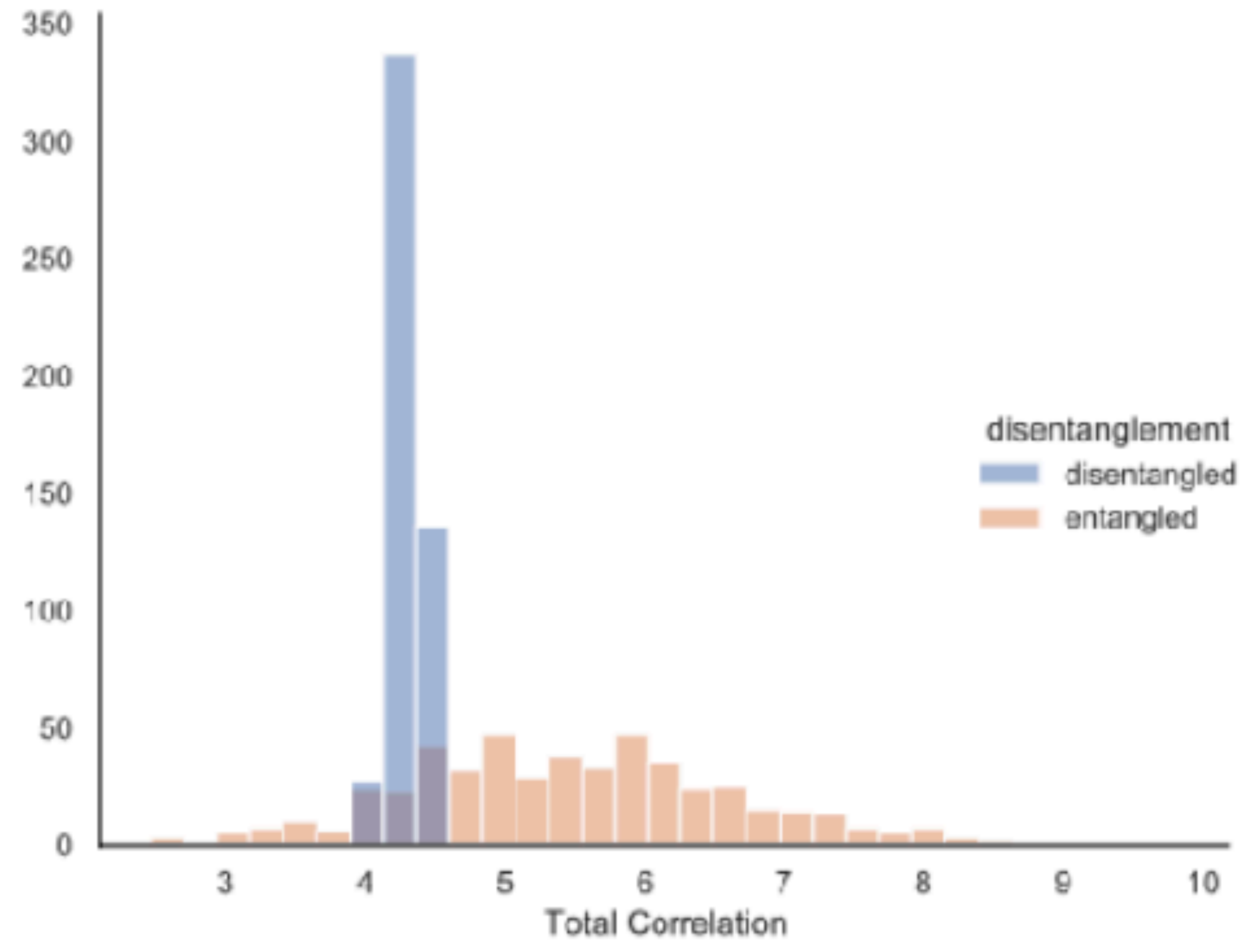




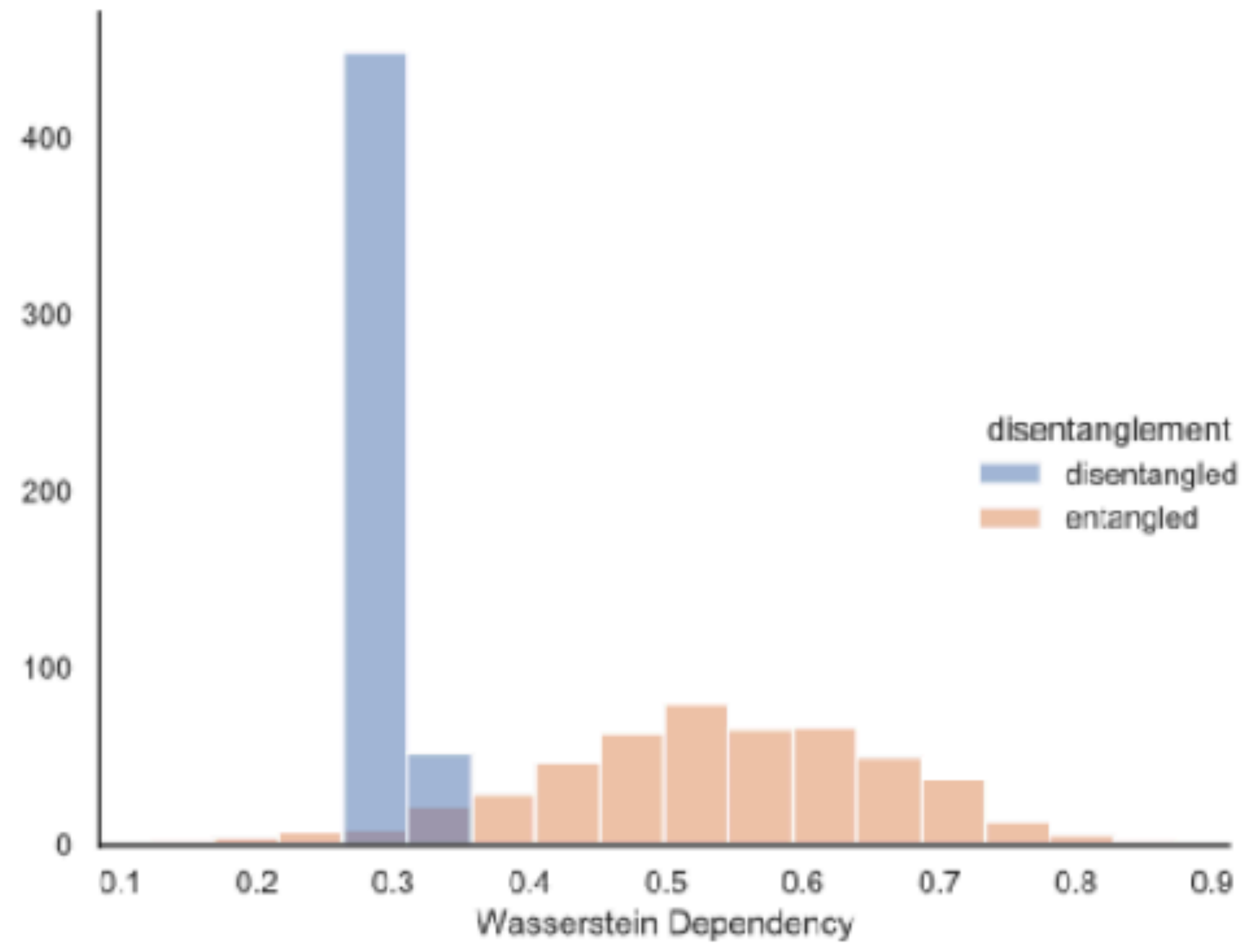
Predict faithfulness to true causal factors



(a) IOSS

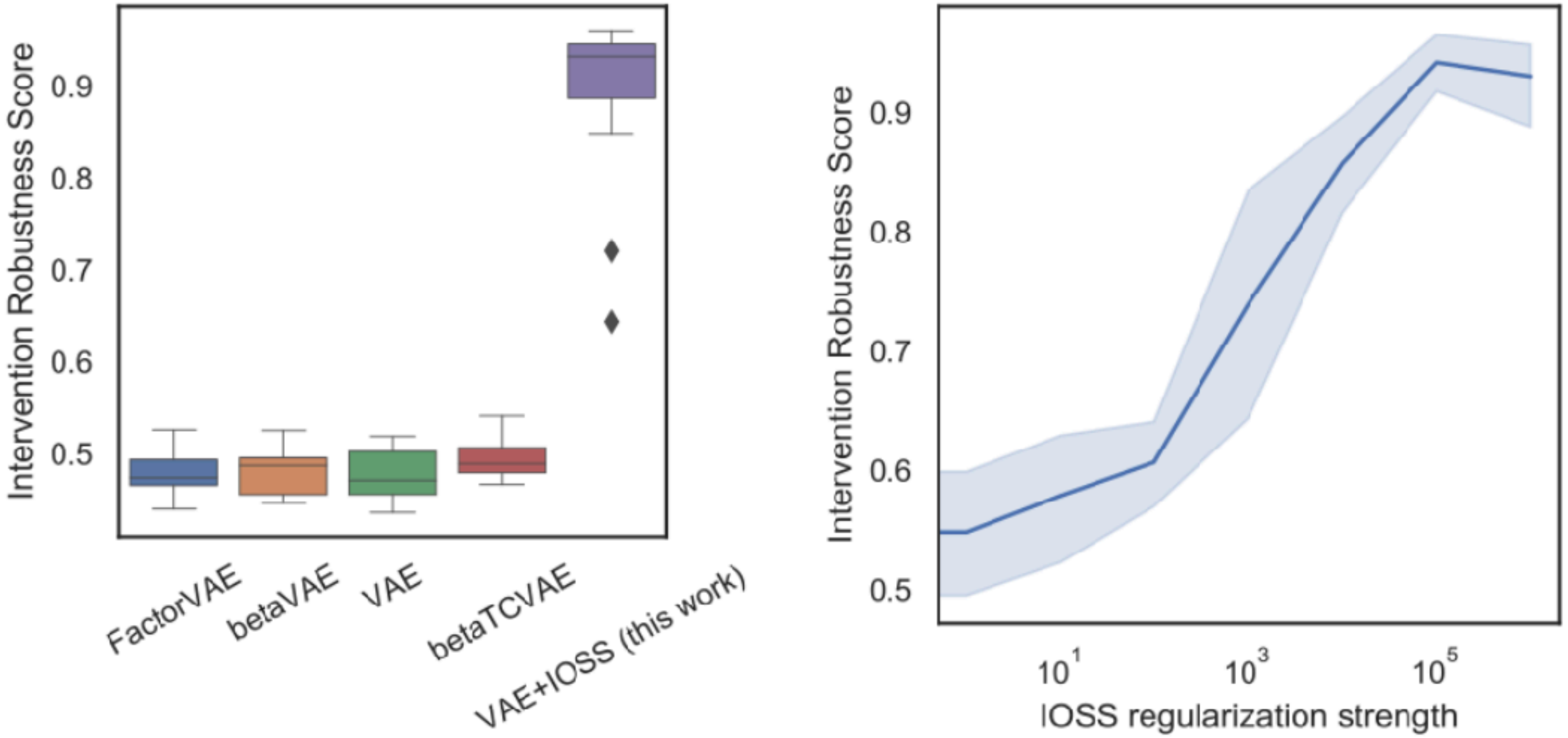


(b) Total Correlation



(c) Wasserstein Dependency

Learning latent causal factors with IOSS

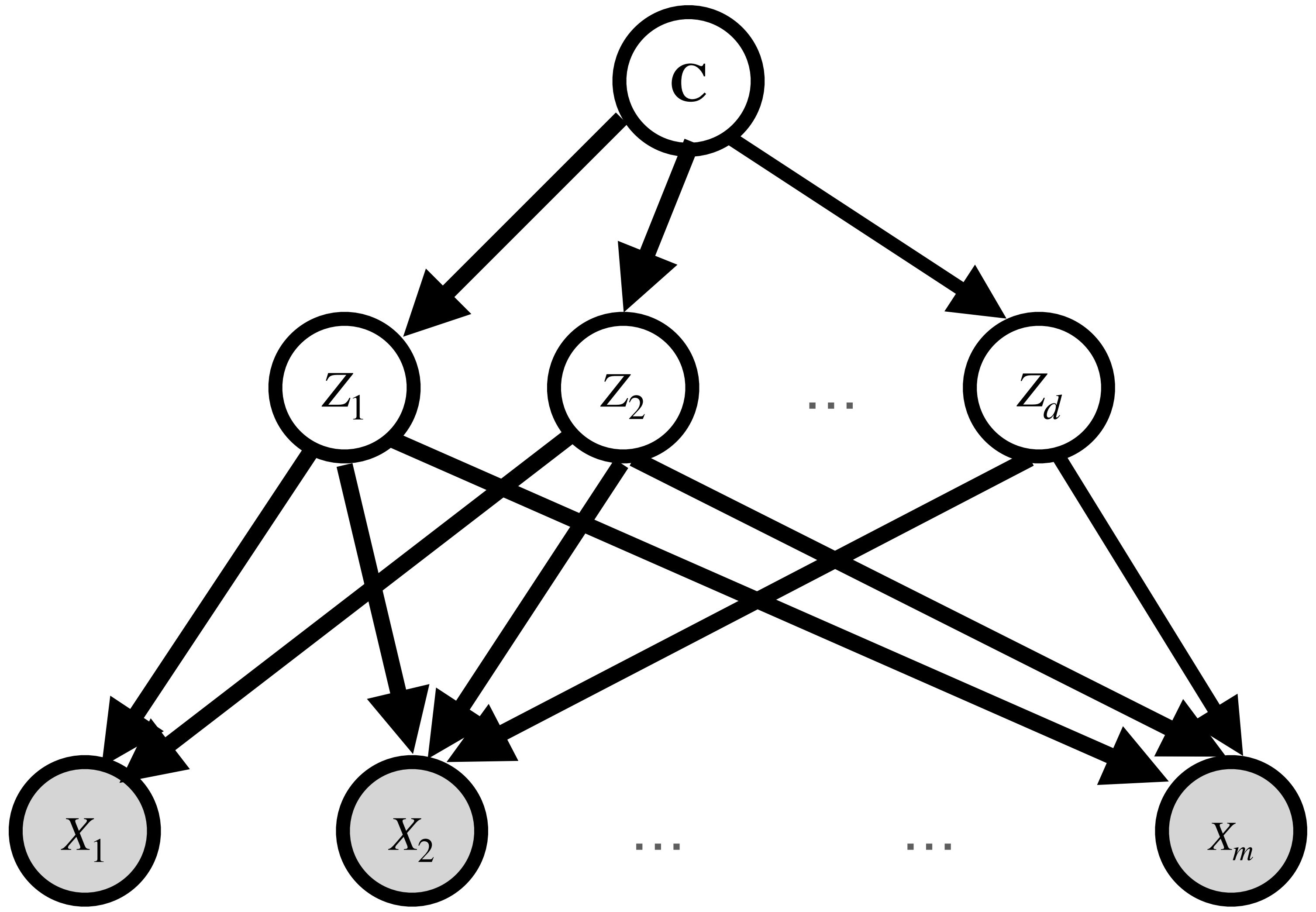


Independent support seems to help,
but can it **identify** correlated latents?

Rest of the talk

Data generating process

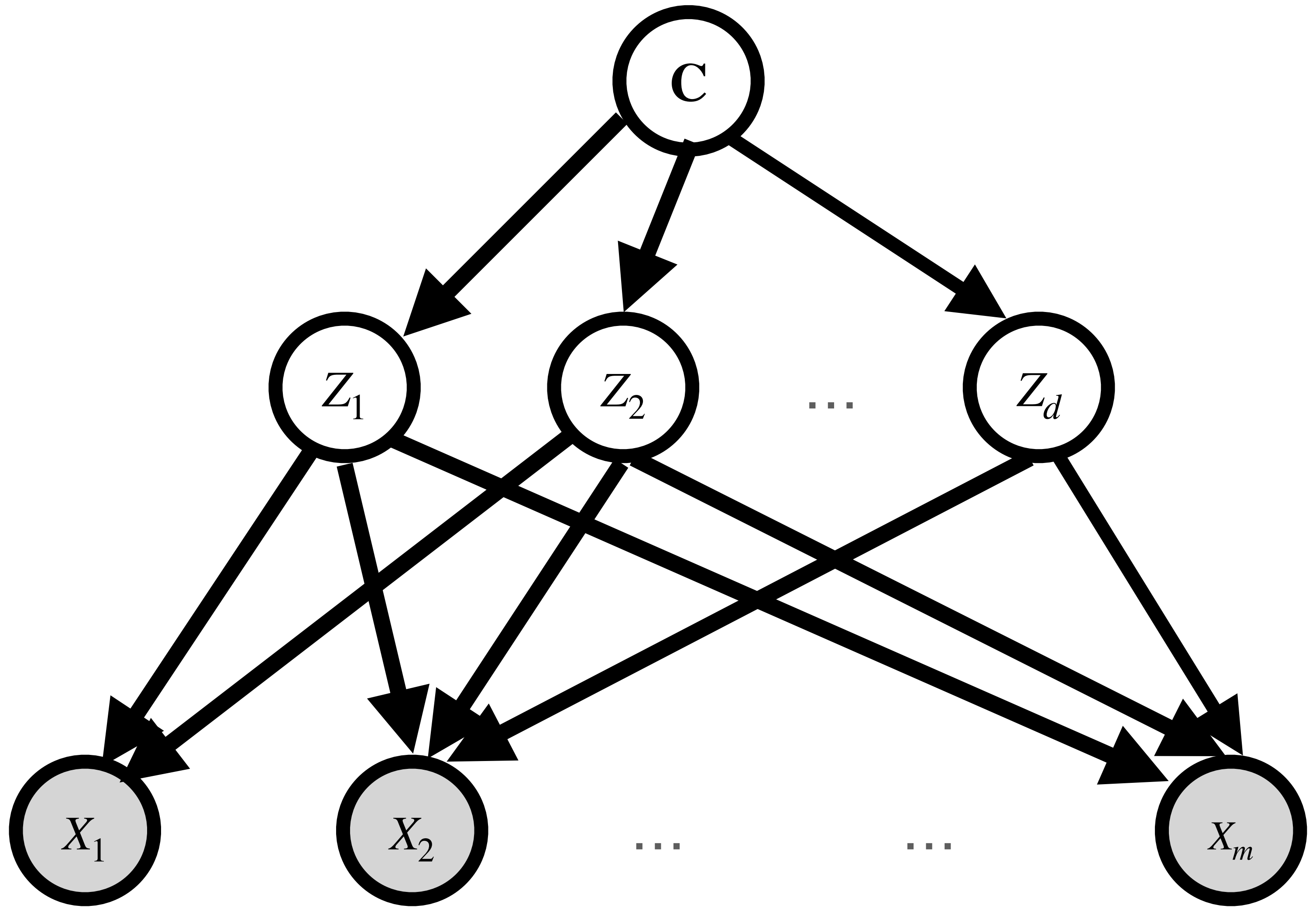
$$\forall j \in \{1, \cdots, K\}, \;\; z \sim \mathbb{P}_Z^{(j)} \quad x \leftarrow g(z)$$

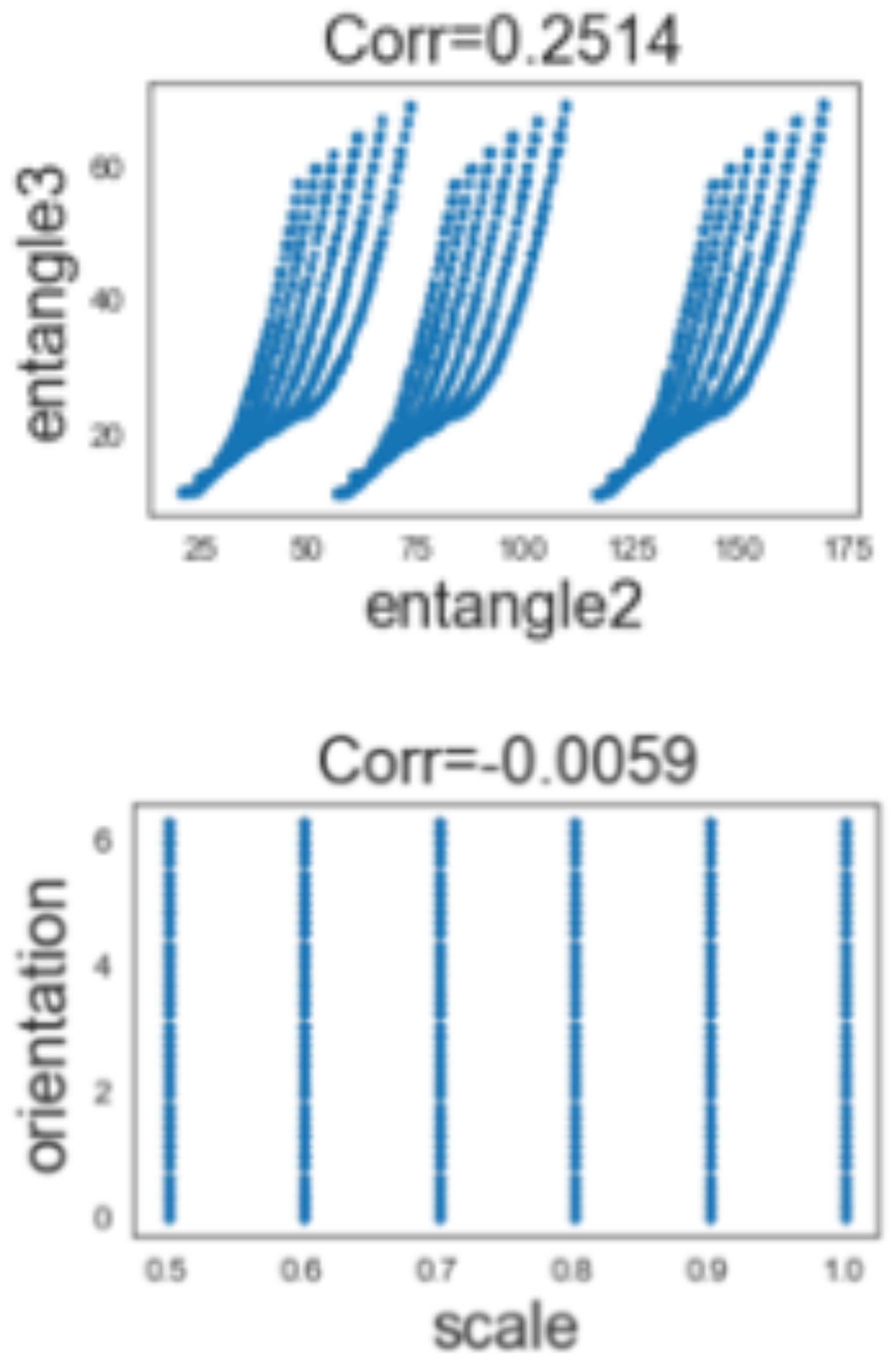


- Explore the **properties** of **distributions** (potentially across domains) and **mixing functions** g that permit representation **identification** of latent causal factors through **simple regularizers**

Correlated latent causal factors

Identification via independent support



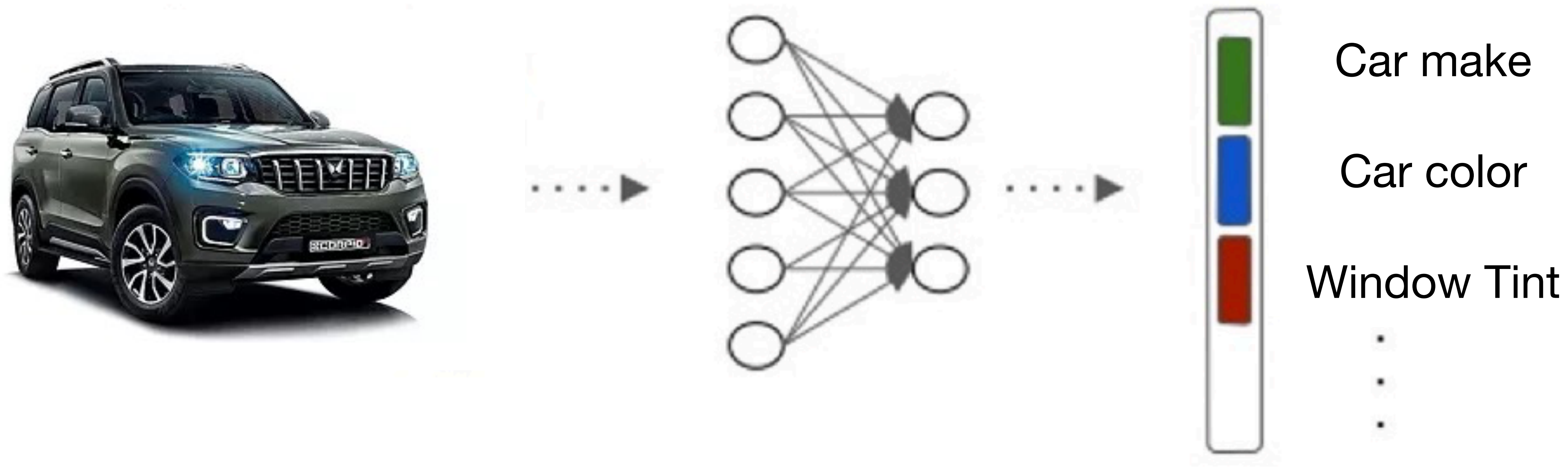


- **Informal Theorem** ([Ahuja, Mahajan, W., & Bengio, 2022]) Under **polynomial** decoder and **bounded** true factors, the **pairwise independent support** condition can **identify** latent causal factors up to **permutation, shift, and scaling**.

Identification via independent support

Setup

- **Data generating process:**
  - $x \leftarrow g(z)$,
    - $z \sim P_z$ are true latent factors,
    - $g : \mathbb{R}^d \to \mathbb{R}^n$ is an injective mixing function.
- **Goal:** Learn an encoder $f : \mathbb{R}^n \to \mathbb{R}^d$; for each x estimate the true latent z
- **Representation:** $\hat{z} = f(x)$.



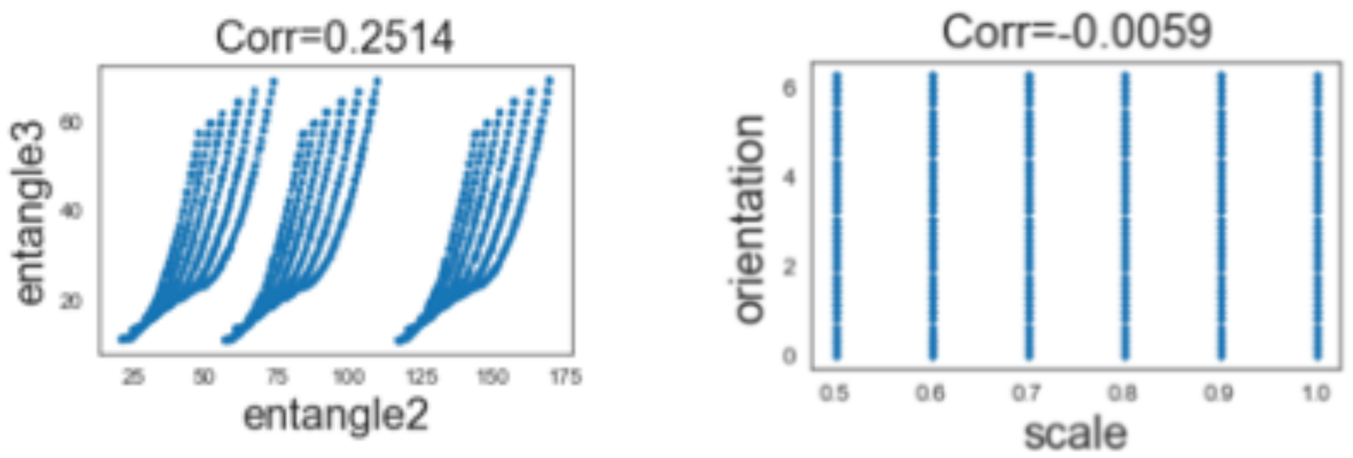
Identification via independent support

Identifiability

- **Algorithm:**
  - $h \circ f(x) = x, \qquad \forall x \in \mathcal{X}$ **reconstruction identity**
  - s.t. $\hat{\mathcal{Z}}_{k,m} = \hat{\mathcal{Z}}_k \times \hat{\mathcal{Z}}_m, \qquad \forall k, m$ **independent support** constraint
    - $\hat{\mathcal{Z}}_j$ is support of the jth dim of the representation $\hat{z} = f(x)$.

Identification via independent support

Identifiability



- **Identifiability:**
  - Under suitable conditions, it **identify** latent causal factors up to **permutation, shift, and scaling:**
    - the learned representation satisfies $\hat{z} = \Pi \Lambda z + c,$
  - Π is permutation matrix and Λ is diagonal matrix.

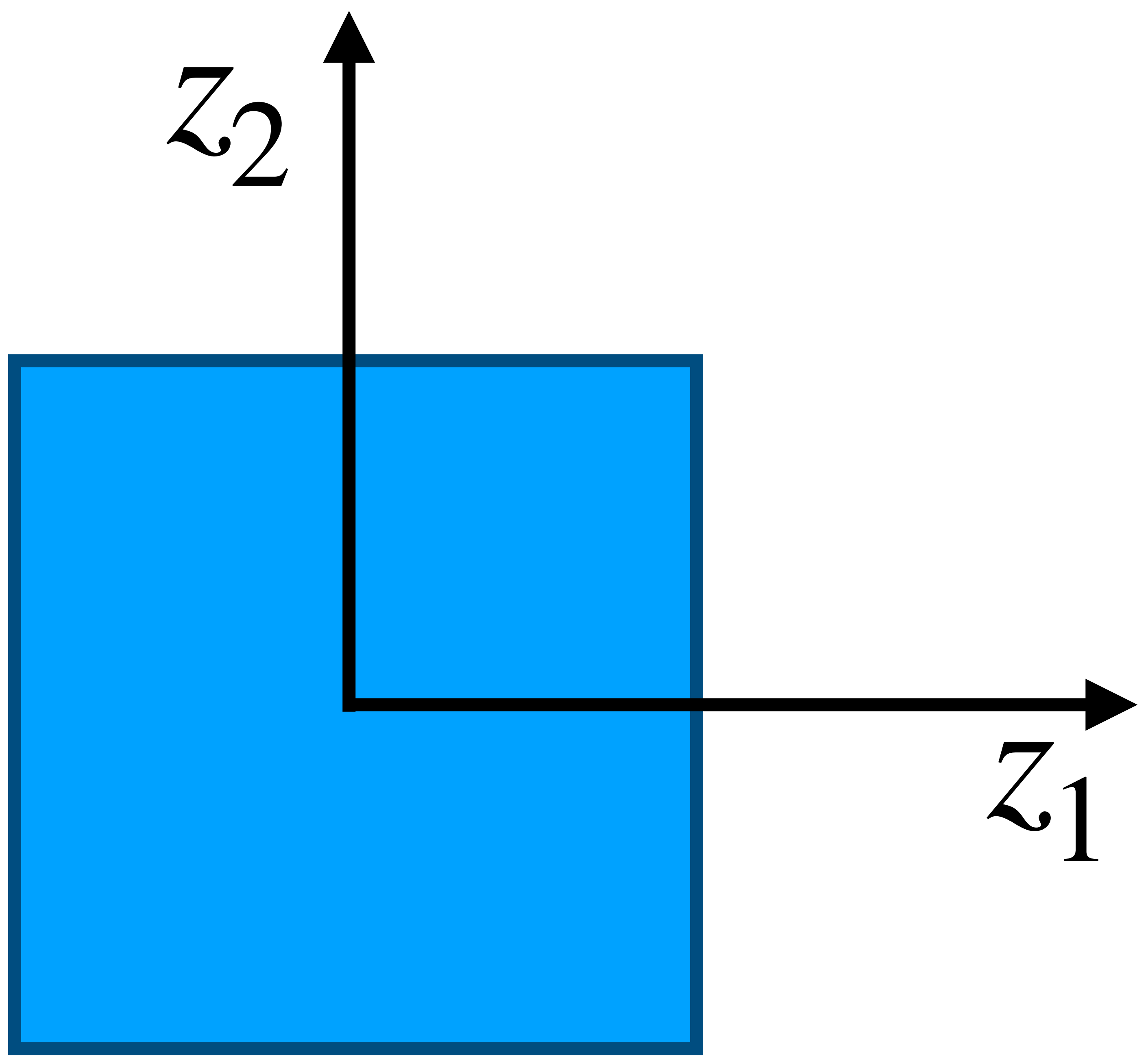
Generalizes linear ICA to polynomial mixing and correlated latents with independent support

Identification via independent support

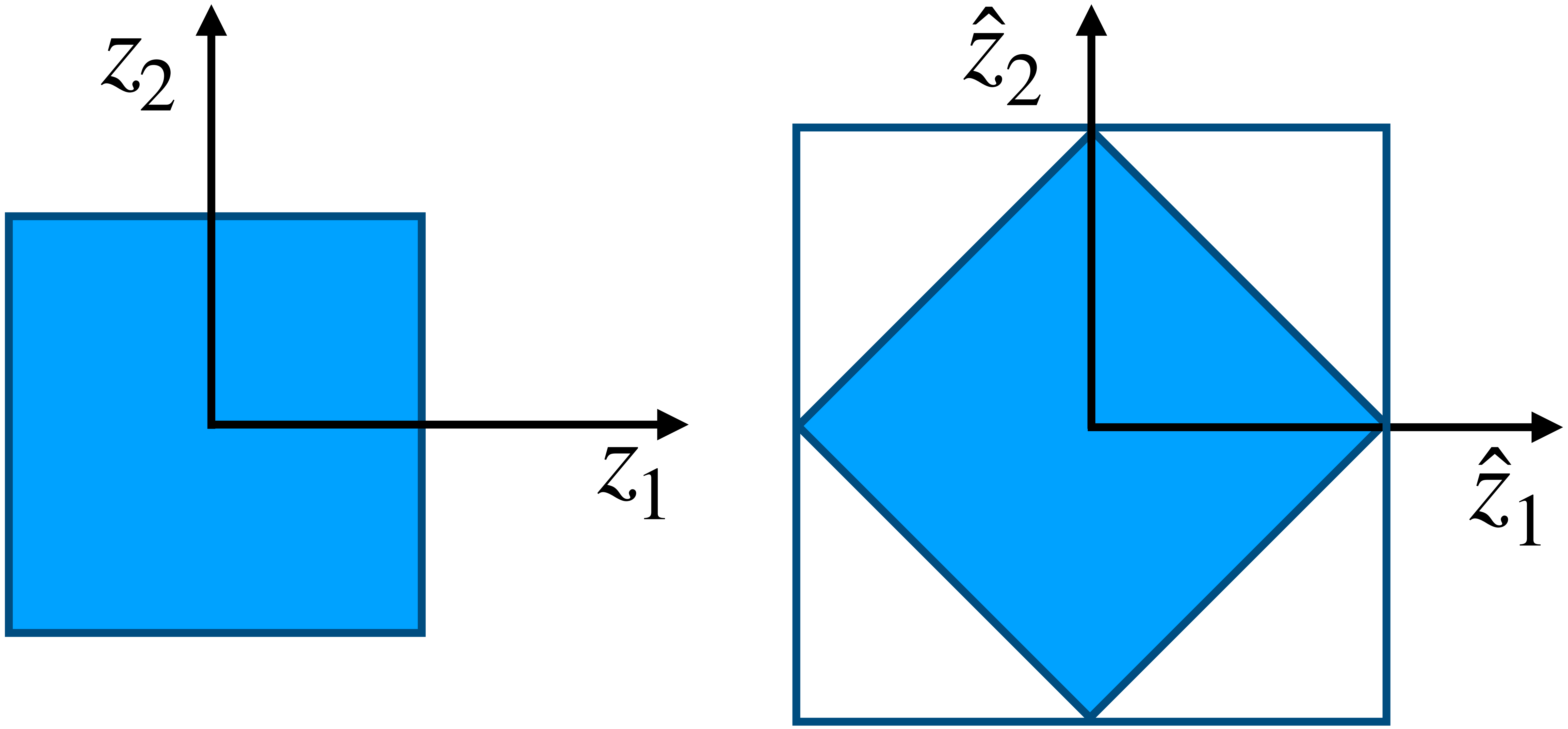
How can we achieve identification?

- **Two steps:**
  - **Polynomial decoder** gives affine (a.k.a. linear) identification $\hat{z} = Az + c$
  - **Independent support** gives further coordinate-wise identification, $\hat{z} = \Pi\Lambda z + c$, Π is permutation matrix and Λ is diagonal matrix.

Geometric Intuition



Geometric Intuition

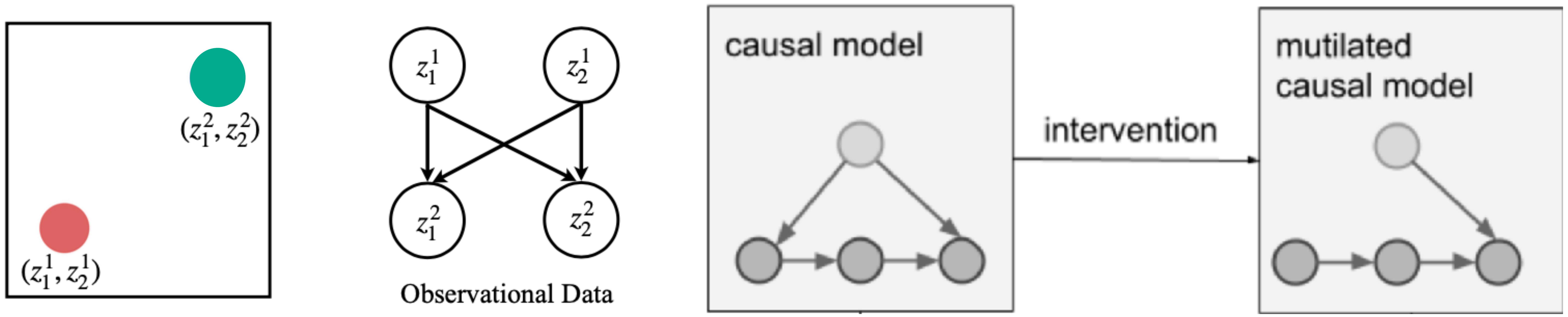


$$\hat{z} = Uz$$

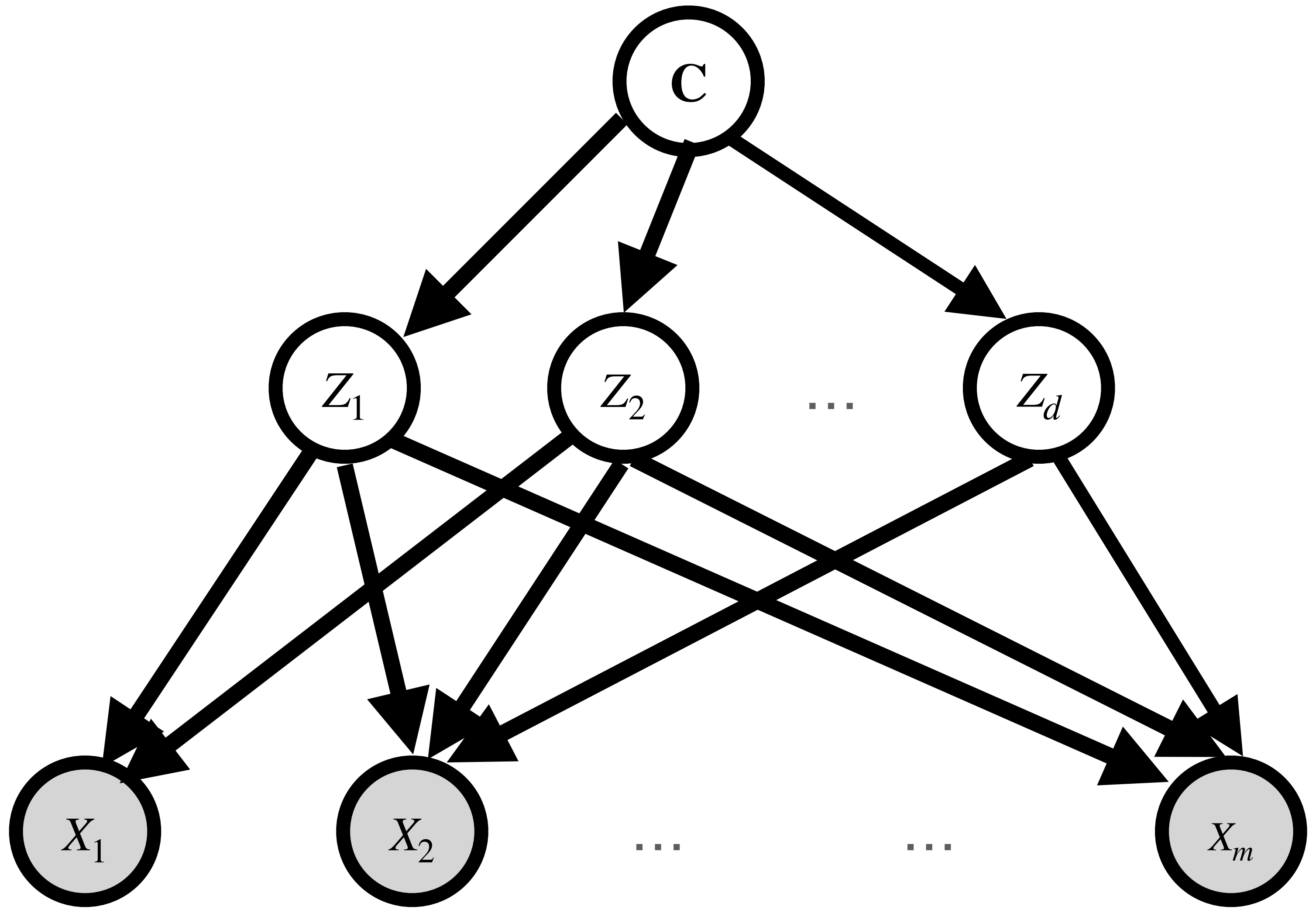
What if the latent causal factors are causally connected?

Causally connected latent causal factors

Interventions (+IOSS) are here to help!



Causally connected latent factors

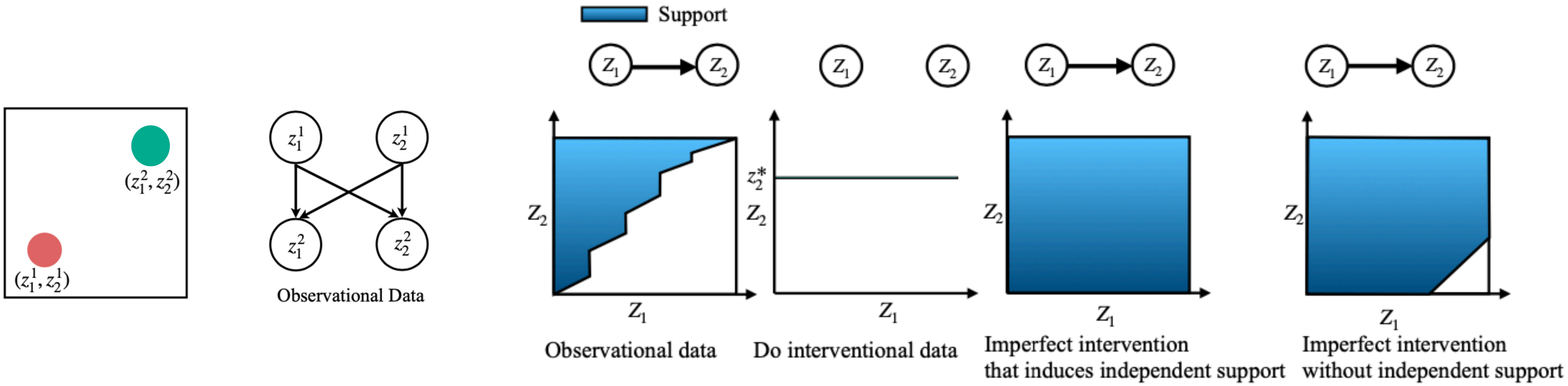


Non-causally-connected latent factors

- **Interventions**—by definition—mutilates the arrow between the intervened variables and its parents.
- Can handle both **perfect** interventions and (some) **imperfect** interventions
- Identify latent **causal factors** with sufficiently many interventions; then identify their **causal graph**

Interventional Causal Representation Learning

Geometric Signatures from multi-domain interventional data



- **Do not know** which latent causal factors were intervened on.
- Only know some factors were intervened.
- **Geometric signatures** reveal the latent causal factors.

Causal Representation Learning using Geometric signals

Correlated or causally connected latents; distribution-free identification

Input data	**Assm. on Z**	**Assm. on g**	**Identification**
Obs	$Z_r \perp Z_s \| U, U$ aux info.	Diffeomorphic	Perm & scale (Khemakhem, 2020)
Obs	Non-empty interior	Injective poly	Affine (Theorem 1)
Obs	Non-empty interior	$\approx$ Injective poly	$\approx$ Affine (Theorem 6)
Obs	Independent support	Injective poly	Perm, shift, & scale (Theorem 4)
Obs + *do* intervn	Non-empty interior	Injective poly	Perm, shift, & scale (Theorem 2)
Obs + *do* intervn	Non-empty interior	Diffeomorphic	$\approx$ Perm & comp-wise (Theorem 7)
Obs + Perfect intervn	Non-empty interior	Injective poly	Block affine (Theorem 3)
Obs + Imperfect intervn	Partially indep. support	Injective poly	Block affine (Theorem 3)
Counterfactual	Bijection w.r.t. noise	Diffeomorphic	Perm & comp-wise (Brehmer, 2022)

[Ahuja, Mahajan, W., & Bengio, 2022]

Causally connected latent causal factors

Interventions (+IOSS) are here to help!

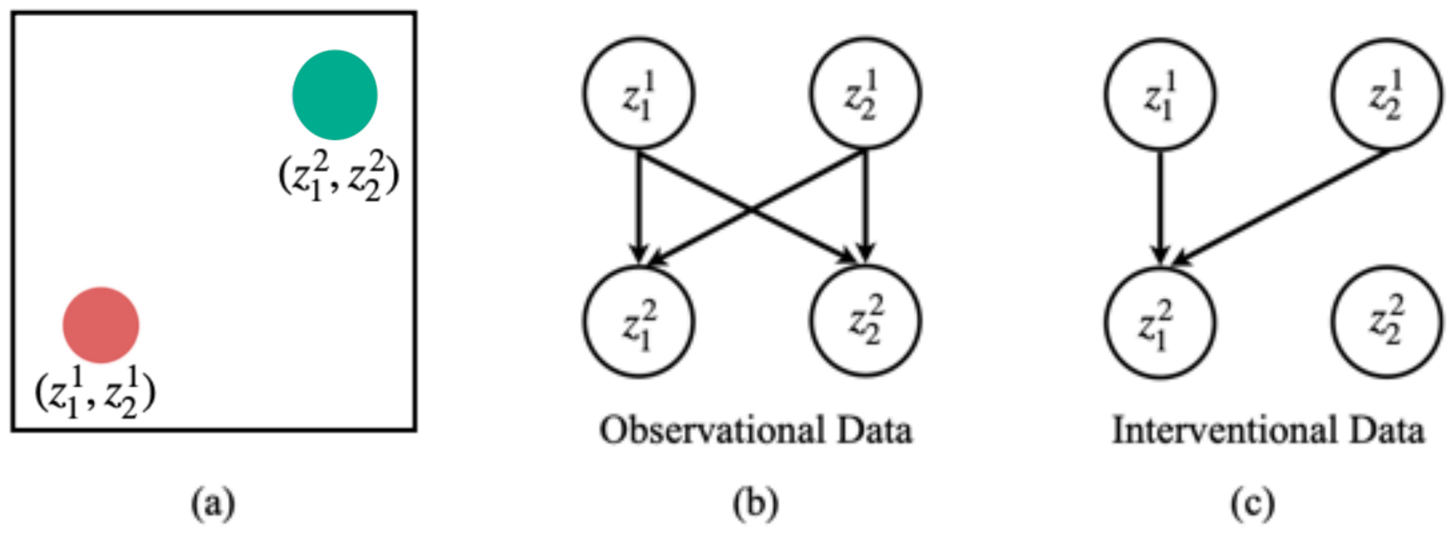
Autoencoder with do intervention penalty

$$\mathbb{E}\left[\|h \circ f(x) - x\|^2\right] + \lambda(f_k(x) - z^\dagger)^2$$

Autoencoder with IOSS penalty

$$\mathbb{E}\left[\|h \circ f(x) - x\|^2\right] + \lambda \sum_{k \neq j} \mathrm{IOSS}_{k,j}$$

Interventional Causal Representation Learning



#interv dist.	Uniform	SCM linear	SCM non-linear
1	33.2 ± 7.09	42.7 ± 1.43	34.9 ± 2.29
3	72.2 ± 4.04	73.9 ± 2.77	65.2 ± 3.71
5	88.3 ± 1.02	83.6 ± 0.94	77.2 ± 1.79
7	88.1 ± 1.10	85.5 ± 0.82	81.9 ± 2.37
9	87.5 ± 1.33	84.8 ± 1.49	81.1 ± 2.53

- Mean correlated coefficient (MCC) with the true causal factors.
- **Interventional causal representation learning with IOSS** can **identify** true latent causal factors, without compromising reconstruction quality.

What just happened?

- Single-node perfect and some imperfect interventions
- One fixed causal graph for entire observational data
- **These assumptions do not apply to complex multi-domain datasets**

The fixed causal graph assumption

General Multi-domain Causal Representation Learning

An invariance principle for causal representations

Domain 1

Domain 2

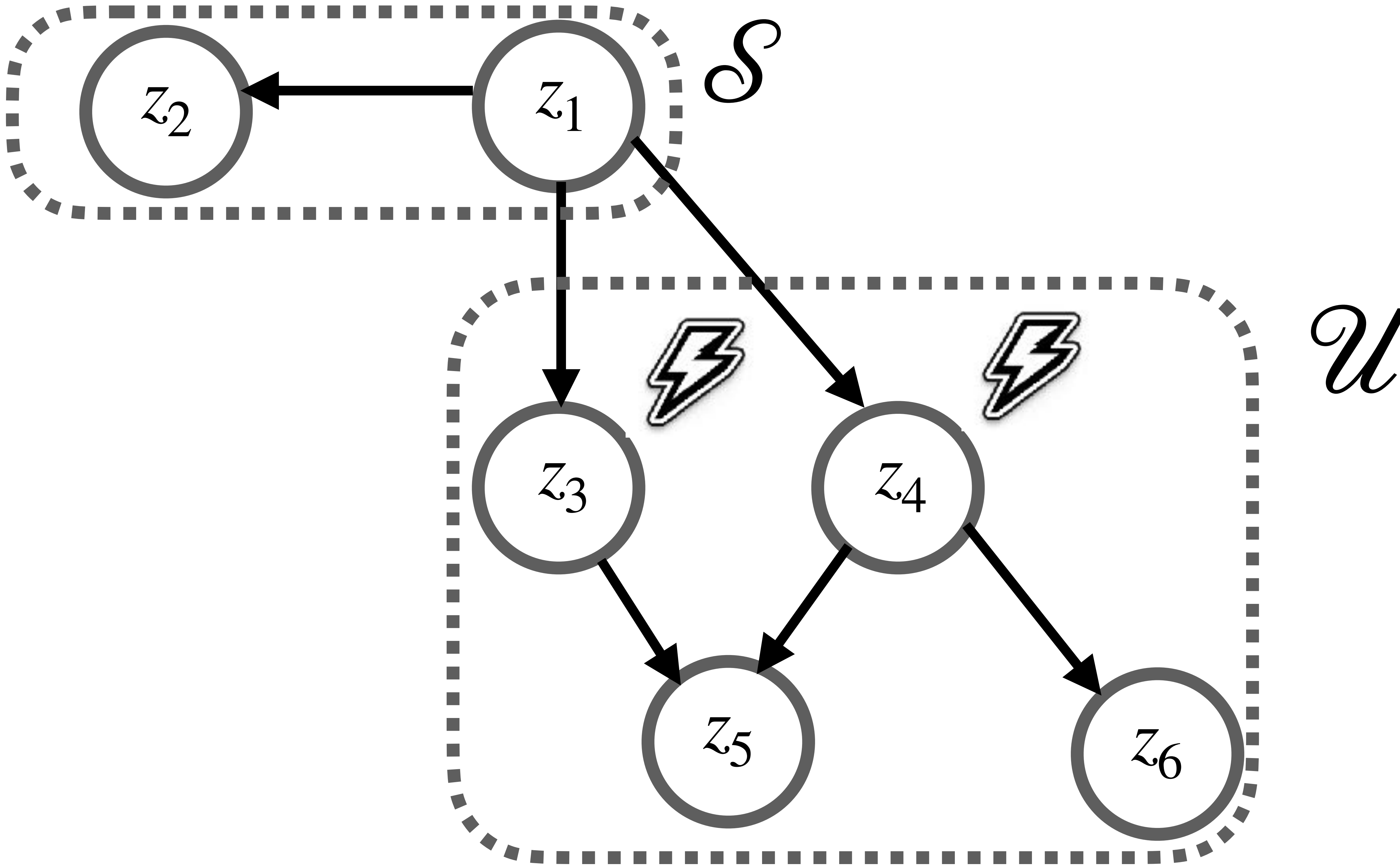
$$F[p_{Z_{\mathcal{S}}}^{(1)}] = F[p_{Z_{\mathcal{S}}}^{(2)}]$$

Distributional properties of a subset of latents is same between two domains

General Multi-domain Causal Representation Learning

An invariance principle for causal representations

- **Multi-node imperfect interventions**
- Distributional properties (e.g. **support**) of **intervened** nodes and **downstream** nodes ($\mathcal{U}$) change
- Rest of the nodes ($\mathcal{S}$) are not impacted



General Multi-domain Causal Representation Learning

Input data	Assm. on p_Z	Assm. on g	Identification
Observational	$z_i \perp z_j \mid u$, 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	Diffeomorhic	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.)

[Ahuja, Mansouri, & W., 2023]

General Multi-domain Causal Representation Learning

Autoencoder with invariance penalty

- **Algorithm (Autoencoder with invariance penalty)**

- $$\mathbb{E}\left[\|h \circ f(x) - x\|^2\right] + \lambda \sum_{j \neq k} D(p^j_{\hat{z}_{\mathcal{S}'}}, p^k_{\hat{z}_{\mathcal{S}'}})$$

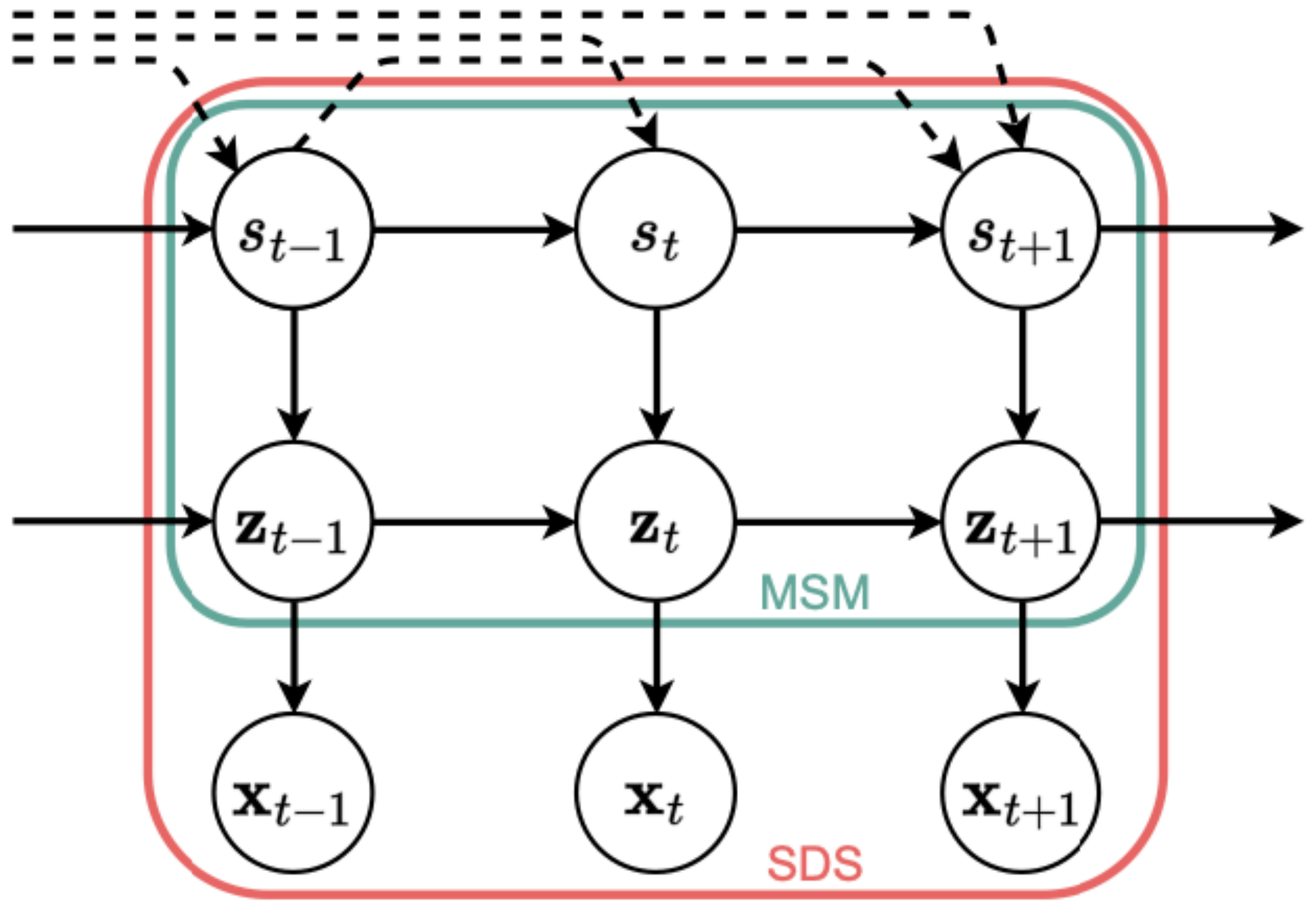
Empirical Studies

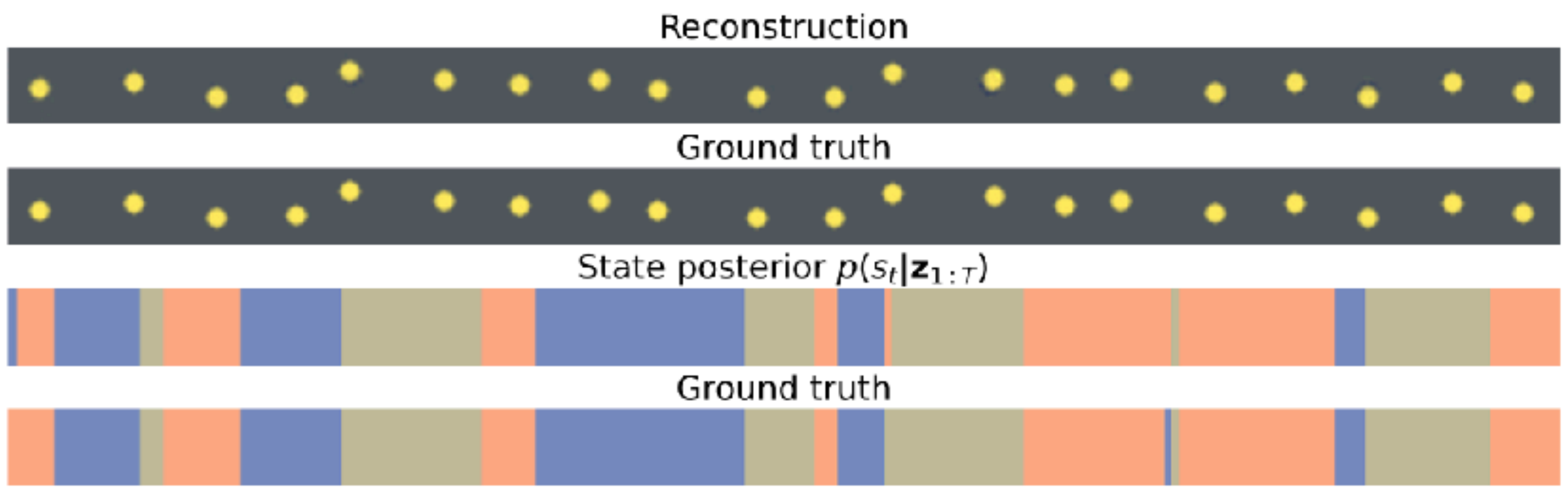
g	Domains	$(R^2_{\mathcal{S}}, R^2_{\mathcal{U}})$
Linear	2	$(0.33 \pm 0.01, 0.46 \pm 0.03)$
Linear	16	$(0.97 \pm 0.00, 0.04 \pm 0.00)$
Polynomial	2	$(0.58 \pm 0.02, 0.07 \pm 0.01)$
Polynomial	16	$(0.95 \pm 0.00, 0.01 \pm 0.00)$
Ball-images	2	$(0.73 \pm 0.01, 0.35 \pm 0.02)$
Ball-images	16	$(0.82 \pm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

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

Causal Inference with Unstructured Data

Switching Dynamical Systems





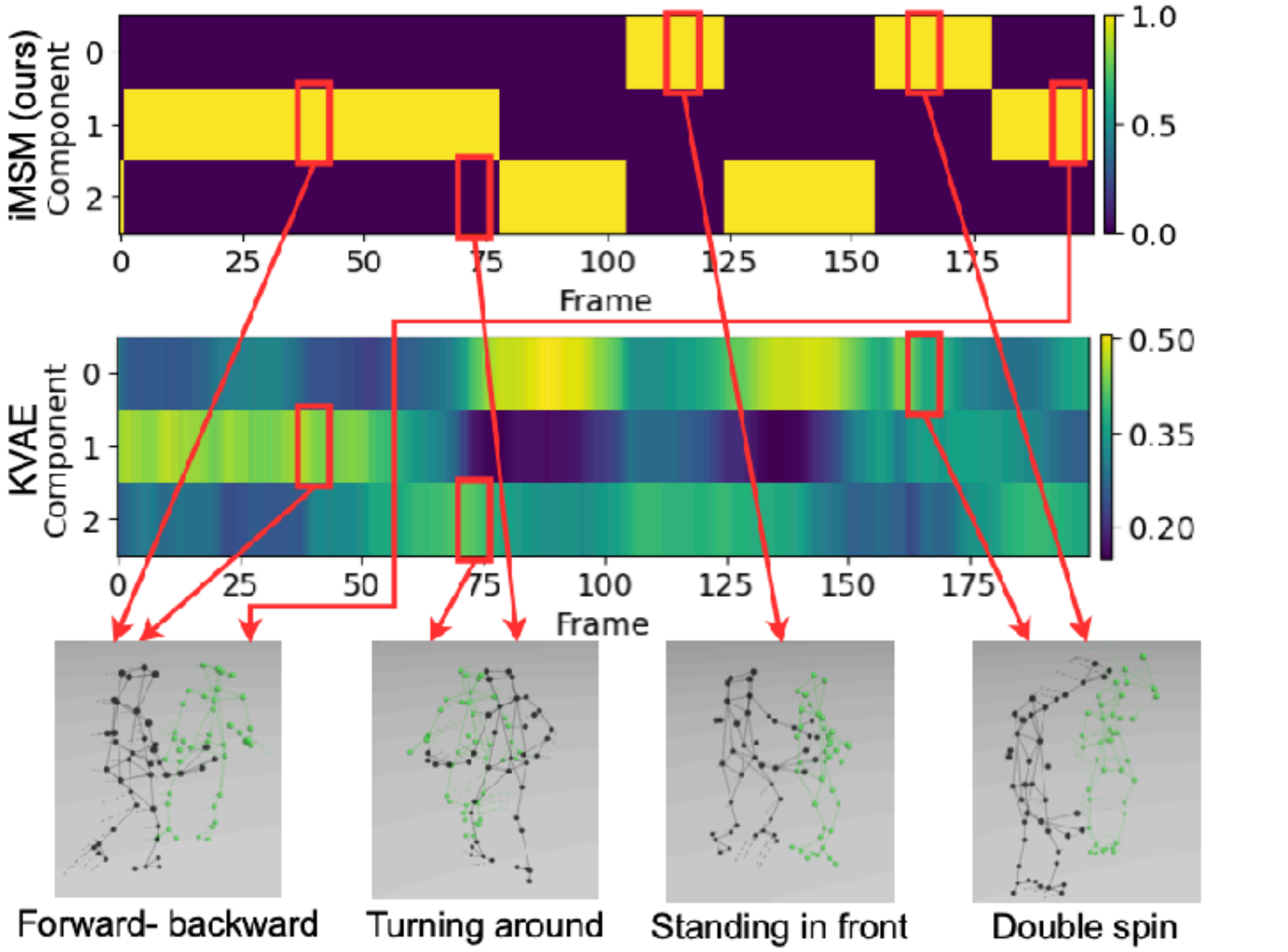


Figure 9: Posterior probability of a *salsa dancing* sequence of iMSM and KVAE (Fraccaro et al., 2017) along with several patterns distinguished in the example.

Takeaways

- Causal inference with **unstructured data** requires identifying latent causal factors first, a task known as **causal representation learning**
- The goal is to identify latent causal factors from **unlabelled** observational, interventional, or multi-domain data.
- Causal factors are often **correlated or causally connected**. How to identify?
- Consider **geometric** signatures e.g. independence-of-support
- Identify latent causal factors from observational, interventional, and general multi-domain data with the **independent or invariant support** constraint.

Thank you!

- Y. Wang and M.I. Jordan
  Desiderata for Representation Learning: A Causal Perspective
  Journal of Machine Learning Research, 2024+
  https://github.com/yixinwang/representation-causal-public

- K. Ahuja, D. Mahajan, Y. Wang, and Y. Bengio
  Interventional Causal Representation Learning
  ICML 2023 (Oral)
  https://github.com/facebookresearch/CausalRepID

- K. Ahuja, A. Mansouri, and Y. Wang
  Multi-Domain Causal Representation Learning via Weak Distributional Invariances
  AISTATS 2024
  https://github.com/facebookresearch/MD-CRL

Extra slides

Affine Identification

Reconstruction identity $h \circ f(x) = x, \forall x \in \mathcal{X}$

(Theorem, Ahuja et al.)

g is an injective polynomial & $\mathcal{Z}$ has a non-empty interior.

Solve the reconstruction identity with the h as a polynomial

$\hat{z} = Az + c, \; \forall z \in \mathcal{Z}$

Affine Identification

$$h \circ f(x) = x$$

$$h(\hat{z}) = g(z)$$

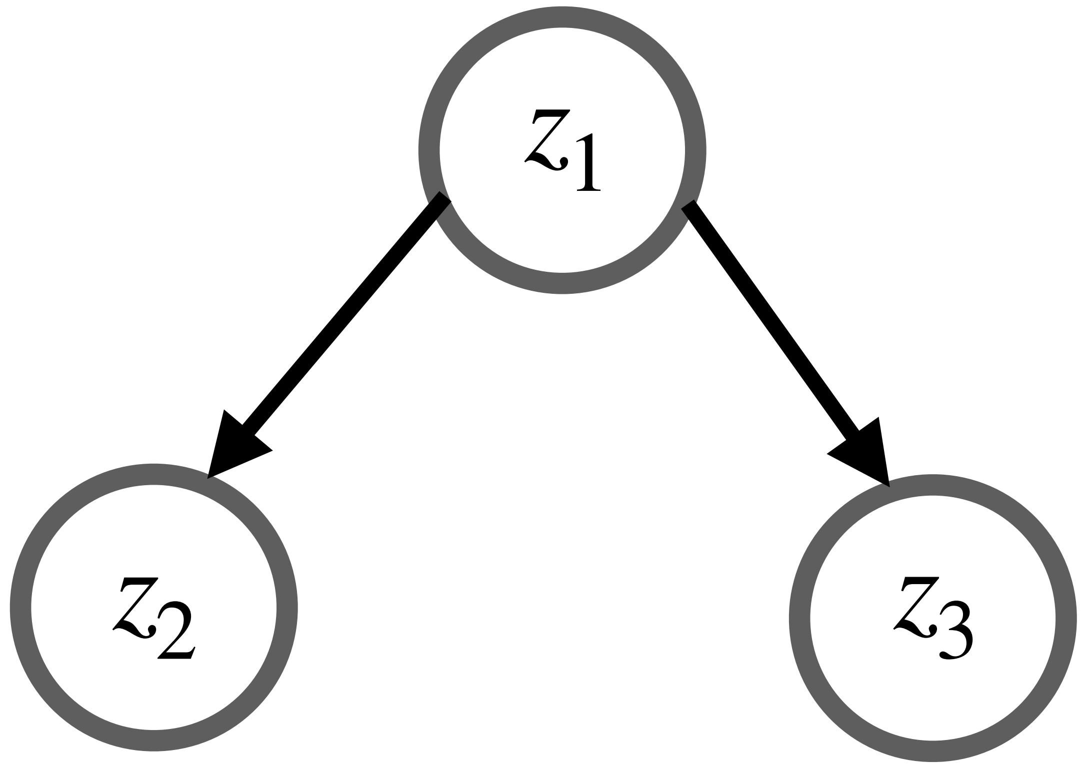
$$H \begin{bmatrix} 1 \\ \hat{z} \\ \hat{z} \otimes \hat{z} \\ \vdots \\ \hat{z} \underbrace{\otimes \cdots \otimes}_{\textbf{p times}} \hat{z} \end{bmatrix} = G \begin{bmatrix} 1 \\ z \\ z \otimes z \\ \vdots \\ z \underbrace{\otimes \cdots \otimes}_{\textbf{p times}} z \end{bmatrix}$$

Affine Identification

$$z = A \begin{bmatrix} 1 \\ \hat{z} \\ \hat{z} \otimes \hat{z} \\ \vdots \\ \hat{z} \underbrace{\otimes \cdots \otimes}_{\textbf{p times}} \hat{z} \end{bmatrix}$$

$$z = A_1 \hat{z} + A_2 \hat{z} \otimes \hat{z} + \cdots + A_p \hat{z} \underbrace{\otimes \cdots \otimes}_{\textbf{p times}} \hat{z}$$

Why invariance works?

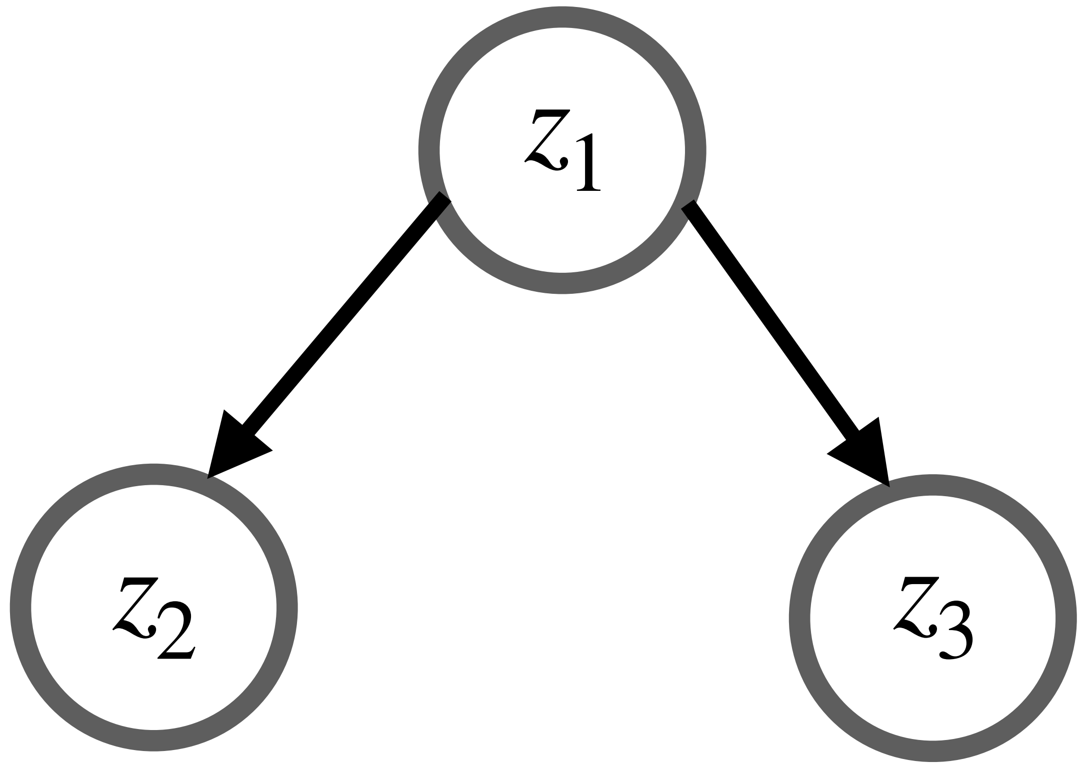


$$\hat{z}_1^{(1)} \overset{d}{=} \hat{z}_1^{(2)}$$

$$\alpha z_1^{(1)} + \beta z_2^{(1)} + \gamma z_3^{(1)} \overset{d}{=} \alpha z_1^{(2)} + \beta z_2^{(2)} + \gamma z_3^{(2)}$$

$$\theta(z_1^{(1)}) + \beta \rho_2^{(1)} + \gamma \rho_3^{(1)} \overset{d}{=} \theta(z_1^{(2)}) + \beta \rho_2^{(2)} + \gamma \rho_3^{(2)}$$

Why invariance works?



$$\underbrace{\theta(z_1^{(1)})}_{u} + \underbrace{\beta\rho_2^{(1)} + \gamma\rho_3^{(1)}}_{v} \overset{d}{=} \underbrace{\theta(z_1^{(2)})}_{\tilde{u}} + \underbrace{\beta\rho_2^{(2)} + \gamma\rho_3^{(2)}}_{\tilde{v}}$$

$$M_u(t)M_v(t) = M_{\tilde{u}}(t)M_{\tilde{v}}(t)$$

$$v \overset{d}{=} \tilde{v}$$

Why support invariance works?

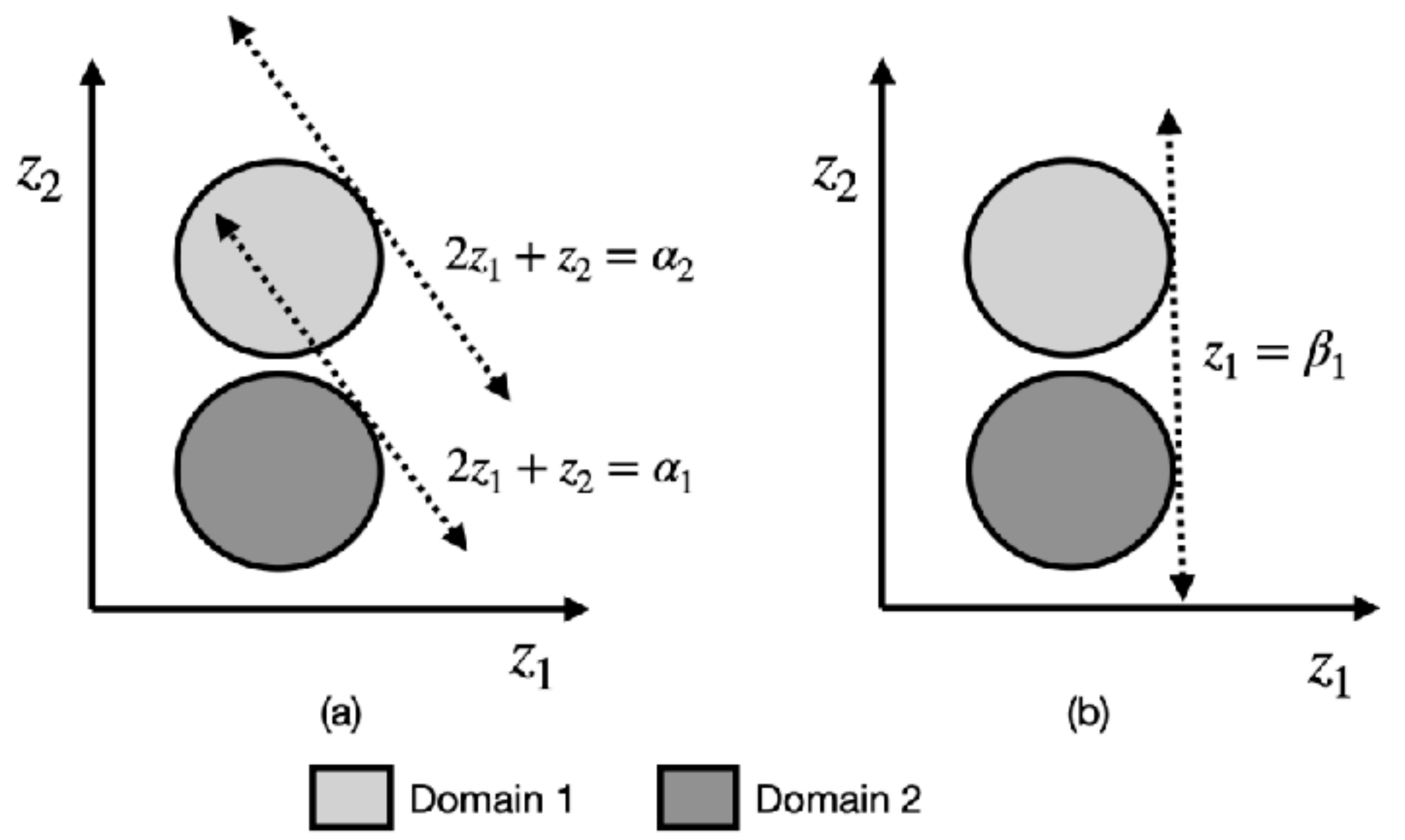


Image Experiments Setup

- **Uniform:** Each coordinate of Ball 1 (x_1, y_1) and Ball 2 (x_2, y_2) are sampled from $\text{Uniform}(0.1, 0.9)$.
- **SCM (linear):** The coordinates of Ball 1 (x_1, y_1) are sampled from $\text{Uniform}(0.1, 0.9)$, which are used to sample the coordinates of Ball 2 as follows:

$$x_2 \sim \begin{cases} \text{Uniform}(0.1, 0.5) & \text{if } x_1 + y_1 \geq 1.0 \\ \text{Uniform}(0.5, 0.9) & \text{if } x_1 + y_1 < 1.0 \end{cases}$$

$$y_2 \sim \begin{cases} \text{Uniform}(0.5, 0.9) & \text{if } x_1 + y_1 \geq 1.0 \\ \text{Uniform}(0.1, 0.5) & \text{if } x_1 + y_1 < 1.0 \end{cases}$$

- **SCM (non-linear):** The coordinates of Ball 1 (x_1, y_1) are sampled from $\text{Uniform}(0.1, 0.9)$, which are used to sample the coordinates of Ball 2 as follows:

$$x_2 \sim \begin{cases} \text{Uniform}(0.1, 0.5) & \text{if } 1.25 \times (x_1^2 + y_1^2) \geq 1.0 \\ \text{Uniform}(0.5, 0.9) & \text{if } 1.25 \times (x_1^2 + y_1^2) < 1.0 \end{cases}$$

$$y_2 \sim \begin{cases} \text{Uniform}(0.5, 0.9) & \text{if } 1.25 \times (x_1^2 + y_1^2) \geq 1.0 \\ \text{Uniform}(0.1, 0.5) & \text{if } 1.25 \times (x_1^2 + y_1^2) < 1.0 \end{cases}$$

Identification via independent support

Why does independent support help?

- **Independent support** gives further coordinate-wise identification, $\hat{z} = \Pi \Lambda z + c$, Π is permutation matrix and Λ is diagonal matrix.

- **Why?** Suppose we have two sets of representations (z_1, z_2) and $(\hat{z}_1, \hat{z}_2)$
    - Polynomial decoder makes them linearly identifiable. $\hat{z}_1 = a_{11} z_1 + a_{12} z_2$, $\hat{z}_2 = a_{21} z_1 + a_{22} z_2$.
    - (z_1, z_2) and $(\hat{z}_1, \hat{z}_2)$ cannot both have independent support when $a_{11}, a_{12}, a_{21}, a_{22}$ are all nonzero.

Identification via independent support

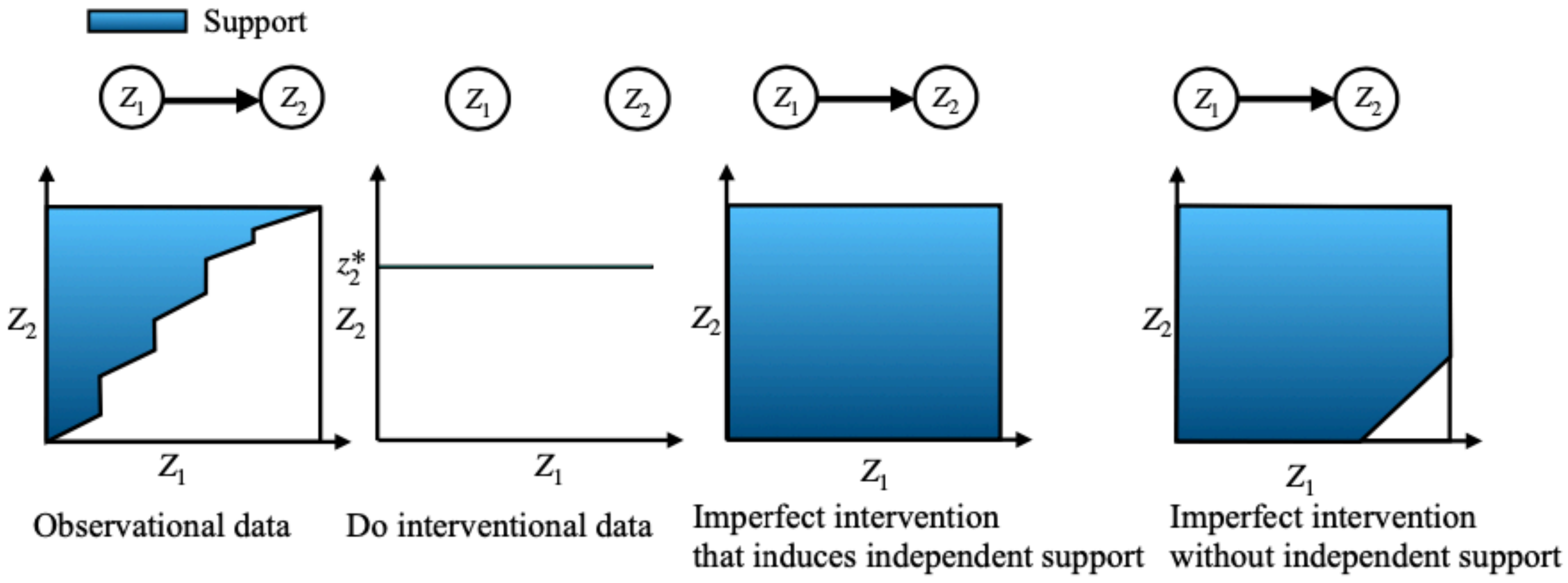
Why does independent support help?

- **Why?**
  - **Core step:** When $\hat{z}_1 = a_{11}z_1 + a_{12}z_2$, $\hat{z}_2 = a_{21}z_1 + a_{22}z_2$.
  - (z_1, z_2) and $(\hat{z}_1, \hat{z}_2)$ cannot both have independent support when $a_{11}, a_{12}, a_{21}, a_{22}$ are all nonzero.
- **Intuition (example):** $\hat{z}_1 = z_1 + z_2$**,** $\hat{z}_2 = z_1 - z_2$**,** $\mathrm{supp}(z_1, z_2) = [1,2] \times [0,2]$**,**
  - The support of $\hat{z}_2$ depends on the value of $\hat{z}_1$, violating independent support.
  - $\mathrm{supp}(\hat{z}_2 \,|\, \hat{z}_1 = 4) = \{0\}$, $\mathrm{supp}(\hat{z}_2 \,|\, \hat{z}_1 = 1) = \{1\}$

Interventional Causal Representation Learning

Geometric Signatures III: Perfect and imperfect interventions

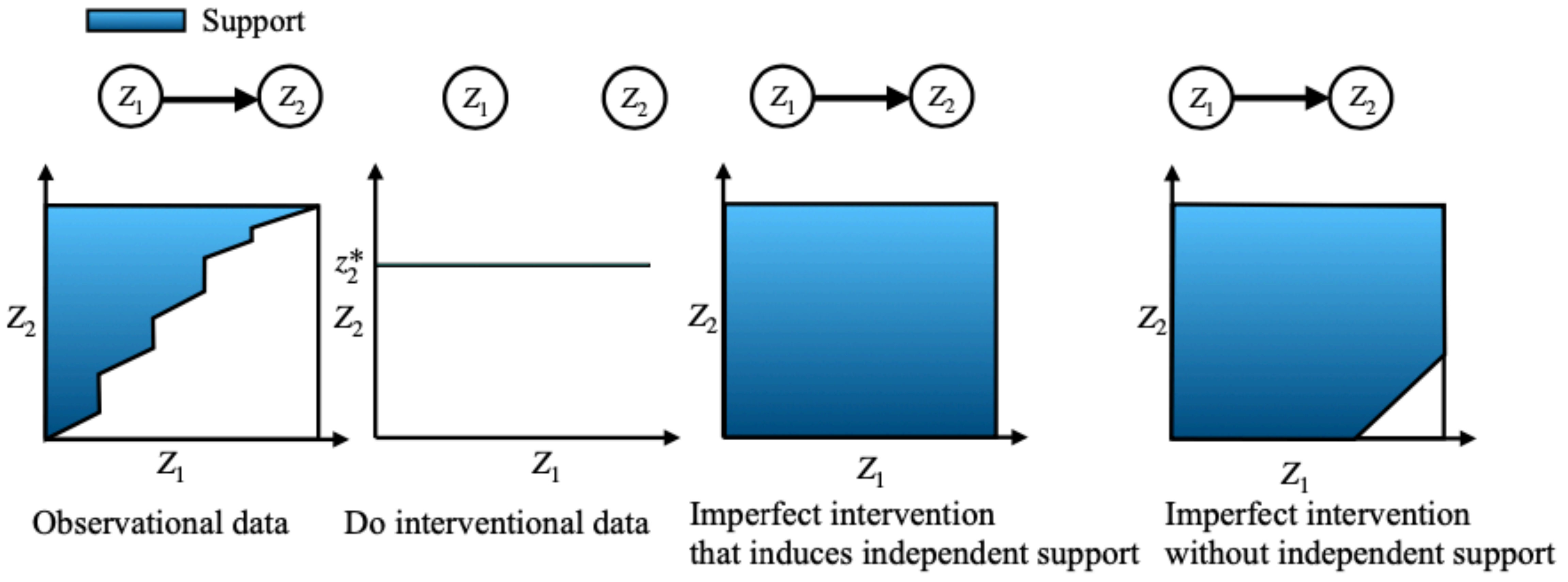
- **Data generating process:**
  - Support independence under intervention on i $\quad \mathcal{Z}_{i,j}^{(i)} = \mathcal{Z}_{i}^{(i)} \times \mathcal{Z}_{j}^{(i)} \ \ \forall j \in \mathcal{S}$



Interventional Causal Representation Learning

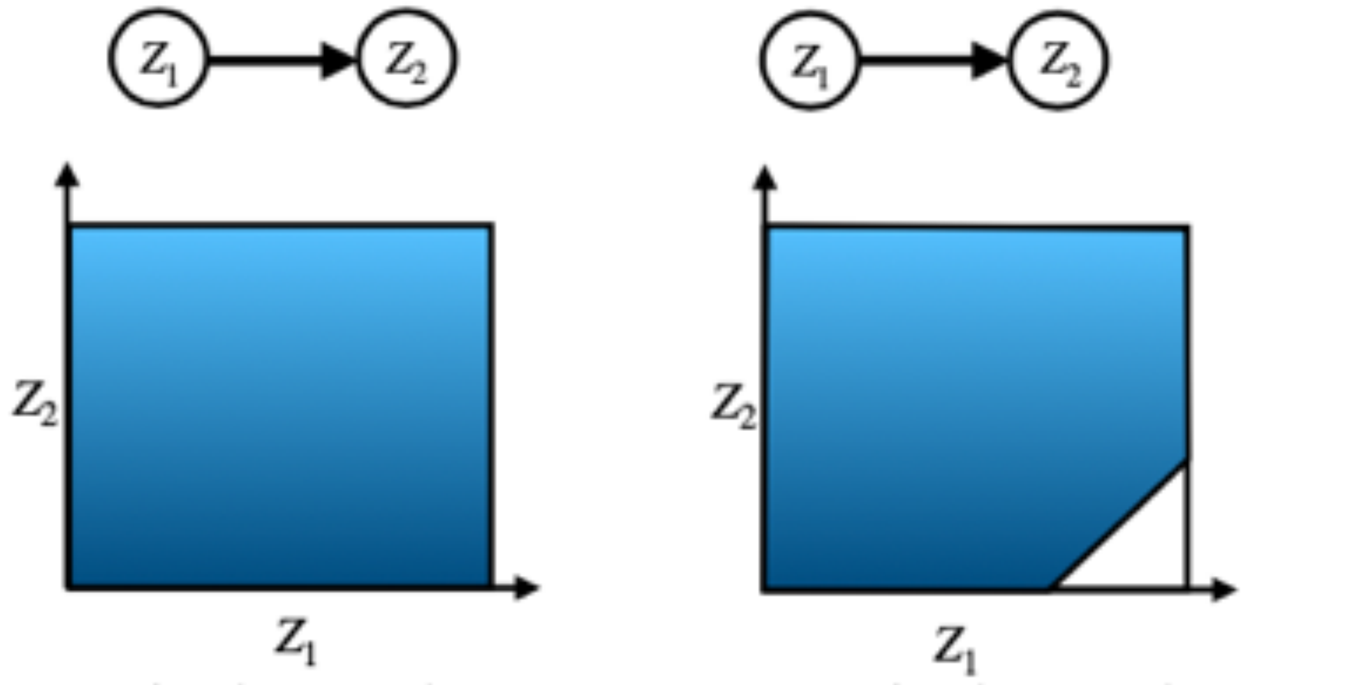
Geometric Signatures III: Perfect and imperfect interventions

- **Algorithm:**
  - $h \circ f(x) = x, \forall x \in \mathcal{X} \cup \{\cup_{j=1}^{t} \mathcal{X}^{(i,j)}\}$ **reconstruction identity**
  - $\hat{\mathcal{Z}}_{k,m}^{(i)} = \hat{\mathcal{Z}}_{k}^{(i)} \times \hat{\mathcal{Z}}_{m}^{(i)}, \forall m \in \mathcal{S}'$ **pairwise independent support** constraint
    - $\mathcal{X}^{(i,j)}$: support of x under j^{th}do intervention on z_i



Interventional Causal Representation Learning

Geometric Signatures III: Perfect and imperfect interventions



- **Theorem** ([Ahuja, Mahajan, W., & Bengio, 2022]
- Suppose
  - (1) the true mixing function g is an **injective polynomial**
  - (2) the support of latents $\mathcal{Z}$ has a **non-empty interior**
  - (3) the intervened latent's **support** is **independent** from the latents in $\mathcal{S}$
- Then the **intervened latent** can be identified up to **block-affine transformations**
  - The algorithm returns representation $\hat{z} = f(x)$ that satisfies $\hat{z}_k = a_k^\top z + c_k$, $\hat{z}_m = a_m^\top z + c_m, \forall m \in \mathcal{S}'$, where a_k and a_m do not share non-zero components.

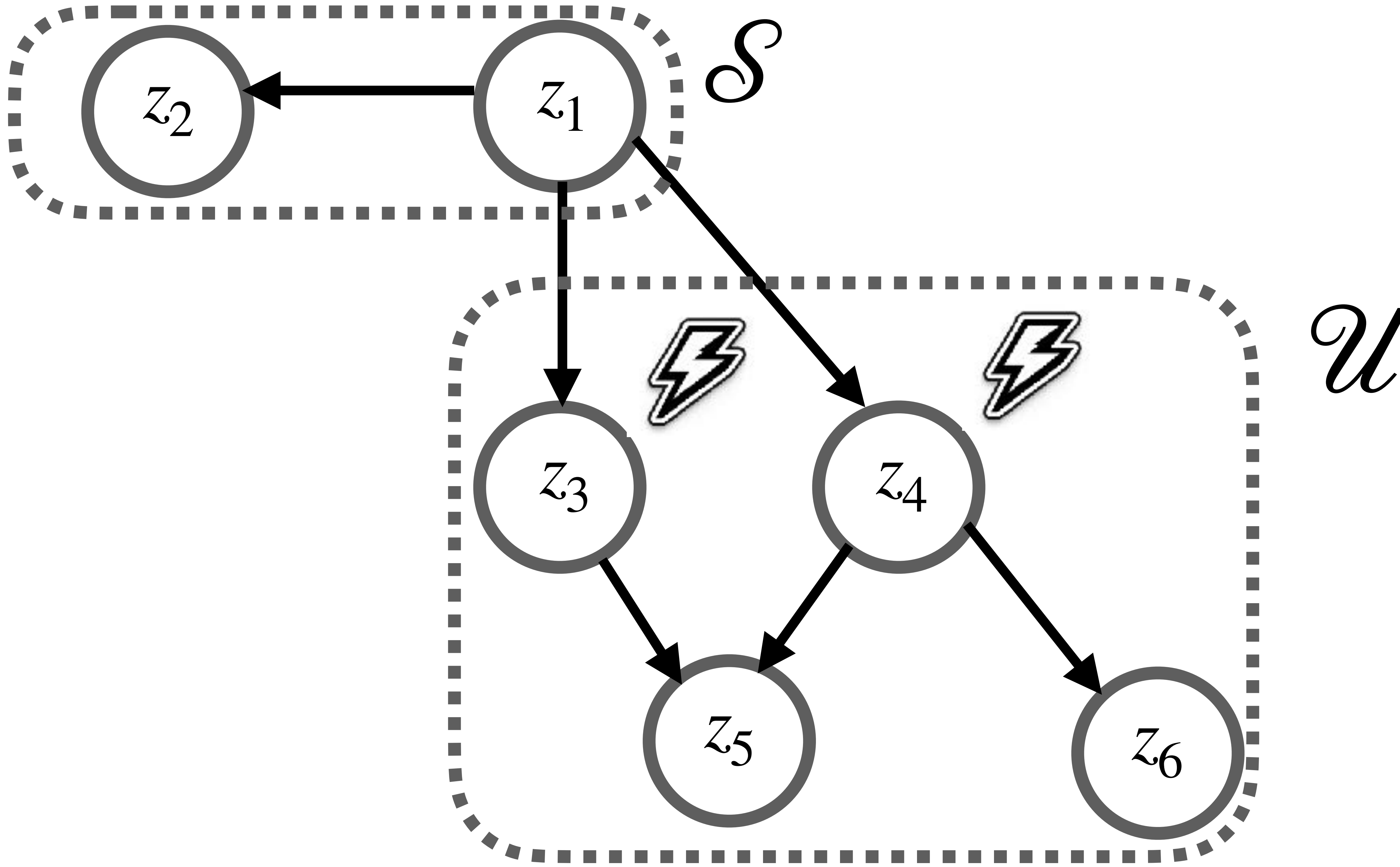
What just happened?

- Single-node perfect and some imperfect interventions
- One fixed DAG for entire observational data
- **These assumptions do not apply to complex multi-domain datasets**

One fixed DAG assumption

An invariance principle for causal representations

- Multi-node imperfect interventions
- Distributional properties (e.g. **support**) of **intervened** nodes and **downstream** nodes ($\mathscr{U}$) change
- Rest of the nodes ($\mathscr{S}$) are not impacted



An invariance principle for causal representations

Domain 1

Domain 2

$$F[p^{(1)}_{Z_{\mathcal{S}}}] = F[p^{(2)}_{Z_{\mathcal{S}}}]$$

Distributional properties of a subset of latents is same between two domains

Self-supervised learning: Instance-level invariance

$$\phi(x^1) = \phi(x^2)$$

Subset of latents between two augmentations is same

Invariance Constrained Autoencoder

$$h \circ f(x) = x, \forall x \in \mathcal{X}$$ **Reconstruction identity**

$$F[p^{(r)}_{\hat{z}_{\hat{\mathcal{S}}}}] = F[p^{(s)}_{\hat{z}_{\hat{\mathcal{S}}}}], \forall r \neq s$$ **Invariance constraint**

Identification under Multi-Node Intervention

- Data generating process

$$z_i \leftarrow q_i\Big(\mathsf{Pa}(z_i)\Big) + \rho_i, \forall i \in [d]$$

$$x \leftarrow g(z)$$

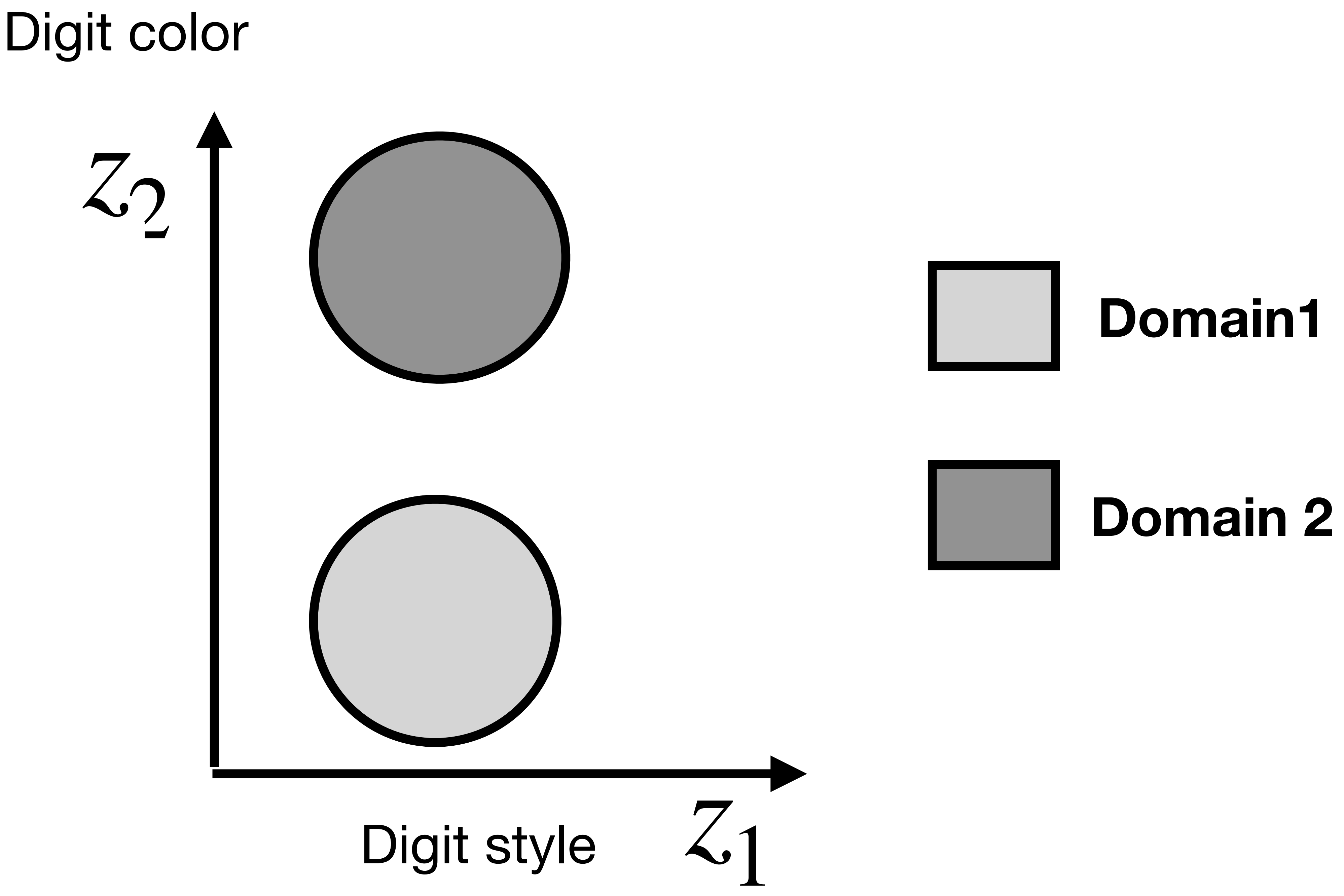
Identification under Multi-Node Intervention

- **Theorem** ([Ahuja, Mansouri, & W., 2023]
- Suppose
  - (1) the true mixing function g is an **injective polynomial**
  - (2) the support of latents $\mathscr{Z}$ has a **non-empty interior**
  - (3) each node undergoes interventions at least t times, $t \geq \dfrac{\log(d/\delta)}{\log(1/(1-1/2d))}$
- Then, wp $1-\delta$, the **un-stable** (intervened) latent can be **separated** from **stable** (un-intervened) latents
  - The algorithm returns representation $\hat{z} = f(x)$ that achieves block-affine identification, $\hat{z}_{\hat{\mathcal{S}}} = Az_{\mathcal{S}} + c$

General Multi-domain Datasets

Domain 1 Domain 2

$$F[p^{(1)}_{Z_S}] = F[p^{(2)}_{Z_S}]$$



Identification in General Multi-Domain Datasets

- **Theorem** ([Ahuja, Mansouri, & W., 2023]
- Suppose
  - (1) the true mixing function g is an **injective polynomial**
  - (2) the support of latents $\mathcal{Z}$ has a **non-empty interior**
  - (3) across domains, the stable latents $\mathcal{S}$ have invariant support
  - (4) There exist two domains p, q such that for each $z \in \mathcal{Z}^{(p)}$ there exists a $z^{'} \in \mathcal{Z}^{(q)}$ such that $z \geq z^{'}$ with strict domination in components in $\mathcal{U}$ (for each orthant)
- Then, the **un-stable** (intervened) latent can be **separated** from **stable** (un-intervened) latents
  - The algorithm returns representation $\hat{z} = f(x)$ that achieves block-affine identification, $\hat{z}_{\hat{\mathcal{S}}} = Az_{\mathcal{S}} + c$

Interventional Causal Representation Learning

Autoencoder with invariance penalty

- **Algorithm (Autoencoder with invariance penalty)**

- $\mathbb{E}\left[\|h \circ f(x) - x\|^2\right] + \lambda \sum_{j \neq k} D(p^j_{\hat{z}_{\mathcal{S}'}}, p^k_{\hat{z}_{\mathcal{S}'}})$