Learning Treatment Representations for Downstream Instrumental Variable Regression

Shiangyi Lin [1] **Hui Lan** [1] **Vasilis Syrgkanis** [2 1 3]

Abstract

Traditional instrumental variable (IV) estimators face a fundamental constraint: they can only accommodate as many endogenous treatment variables as available instruments. This limitation becomes particularly challenging in settings where the treatment is presented in a high-dimensional and unstructured manner (e.g. descriptions of patient treatment pathways in a hospital). In such settings, researchers typically resort to applying unsupervised dimension reduction techniques to learn a low-dimensional treatment representation prior to implementing IV regression analysis. We show that such methods can suffer from substantial omitted variable bias due to implicit regularization in the representation learning step. We propose a novel approach to construct treatment representations by explicitly incorporating instrumental variables during the representation learning process. Our approach provides a framework for handling high-dimensional endogenous variables with limited instruments. We demonstrate both theoretically and empirically that fitting IV models on these instrument-informed representations ensures identification of directions that optimize outcome prediction. Our experiments show that our proposed methodology improves upon the conventional two-stage approaches that perform dimension reduction without incorporating instrument information.

[*]Equal contribution [1]Institute of Computational and Mathematical Engineering, Stanford University, United States. [2]Department of Management Science and Engineering, Stanford University, United States. [3]Supported by NSF Award IIS-2337916. Correspondence to: Shiangyi Lin <shiangyi@stanford.edu>.

Proceedings of the 42nd International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

1. Introduction

Instrumental-variable (IV) methods are among the most widely used tools for recovering causal effects in the presence of unmeasured confounding. Unfortunately, classical IV estimators scale poorly when the treatment variable X is itself high-dimensional, unstructured, or both. In modern applications—where the treatment might be provided in the form of clinical treatment pathways encoded as free-text, purchase histories, or genome-wide expression profiles—the number of potentially endogenous coordinates of X can dwarf the number of available instruments Z (e.g. variables related to capacity constraints in a hospital setting, see, e.g., (Dong et al., 2019; Dong et al.; Qin et al., 2023)). A common workaround is to compress X to a low-dimensional summary D with unsupervised techniques (e.g. PCA, auto-encoders) and then run a standard two-stage least squares (2SLS) on D. Because the dimension reduction step ignores Z, however, the resulting regression can suffer from severe omitted-variable bias: directions of X that matter for the first-stage relationship between Z and X may be discarded, violating the exclusion restriction and invalidating the causal inference step.

We propose *Instrument-Guided Representation Learning* (IGRL), a methodology for learning low-dimensional treatment representations that preserve the validity of downstream IV analysis. IGRL folds the instruments directly into the representation learner so that the learned features D capture the variation in X that is driven by Z. The procedure can be viewed as a regularization of the unsupervised learner toward directions that satisfy the exclusion restriction, thereby eliminating the spurious back-door paths that plague two-step approaches. The resulting representation can then be used in an IV analysis, to learn directions of intervention in the representation space that will improve the target outcome and can be translated back to interventions in the original treatment space.

Our work aligns closely to the recent contributions by Vafa et al. and Du et al., which also highlights the omitted variable bias problem in learned representations in the context where representation learning is used for a set of high-dimensional observed confounders of a treatment and designs representation learning techniques to alleviate it. In

that setting, the learned representation can implicitly omit important parts of the observed confounders, causing bias in the final causal estimate due to implicit unobserved confounding. Our goal is inherently different as we want to learn a latent representation of a highly confounded, high-dimensional treatment, as opposed to learning a latent representation of a high-dimensional confounder.

2. Problem Statement: Learning Interventions via Representations

We consider a setting where we are given data that contain samples of variables (Z, X, Y), where X is a high-dimensional "treatment" variable, Y is a scalar outcome of interest and Z is a low-dimensional vector of instruments. The treatment X is heavily confounded via unobserved confounding variables U that have a causal influence on the value of X and also on Y.

Our goal is to learn a latent representation of the highly confounded, high-dimensional treatment, so as to perform instrumental variable analysis on this learned representation and identify an outcome-improving direction of intervention in representation space and hence subsequently also in the original treatment space. Naive representation learning approaches for the treatment run the risk of an omitted variable problem that can invalidate the downstream causal analysis based on instrumental variables. For more discussion, see Appendix B.

Structural Equation Model. To formalize our problem we will consider the following data generating process (structural causal model) for our observed random variables:

$$\begin{aligned} D &= A \cdot Z + U, & U &\perp\!\!\!\perp Z \\ X &= f(D, V), & V &\perp\!\!\!\perp Z \\ Y &= h(D) + \eta(U, V, \epsilon), & \epsilon &\perp\!\!\!\perp Z \end{aligned} \tag{1}$$

where the random variables U, V, D, ϵ are latent and A is an $r \times k$ matrix that captures the effect of the instruments $Z \in \mathbb{R}^k$ on a vector of latent decisions $D \in \mathbb{R}^r$. For convenience of notation, we will assume that $\mathbb{E}[U] = \mathbb{E}[V] = \mathbb{E}[\eta(U, V, \epsilon)] = 0$.[1] U represents the unobserved confounder that drives the elements of the treatment that are also driven by the instrument. ϵ represents an outcome noise variable and is allowed to be correlated with U, V. D represents the aspects of the treatment X that are affected by the instrument and V represents the remaining aspects that describe the treatment X, but are independent of the instrument. In particular, we assume that the encoding/decoding between the latent representations and the observed treatment is invertible:

[1] Appropriate intercept constants need to be added to the equations in the absence of this convention.

Assumption 2.1 (Invertible Encoding)**.** The function f is invertible, and write the encoding function $e(X) = f^{-1}(X) = (D, V)$, i.e. there is a one-to-one correspondence between the high-dimensional treatment X and the characteristics (D, V) that describe the treatment.

From this perspective, (D, V) can be thought as a non-linear decomposition of the treatment into the instrument-dependent and the instrument-independent components. We will further denote with $e_D(X) = D$ and $e_V(X) = V$ for the encodings of the treatment that return the corresponding components. Moreover, we assume that the transformation between the instrument to the latent representation D is full rank.

Assumption 2.2 (Full-Rank Latents)**.** Assume that the matrix A has full row-rank and $\mathbb{E}[ZZ^\top] \succ 0$.

Note that the full rank assumption on $\mathbb{E}[ZZ^\top]$ can always be satisfied by a preprocessing step that applies a PCA transformation to the instruments and removes co-linear or almost co-linear instruments.

Learning Good Interventions via Representations. Given data containing observations (Z, X, Y) stemming from such a structural equation model, our goal is to learn a soft intervention mapping $t(X)$, such that the average intervened outcome is larger than the original outcome. We will denote with $Y^{(X \leftarrow x)}$ the random outcome from the intervention where we fix the value of X to be x. Thus we are searching for a soft intervention $t(X)$ such that:

$$\mathbb{E}\left[Y^{(X \leftarrow t(X))}\right] > \mathbb{E}[Y] \tag{2}$$

Note that due to the one-to-one correspondence of X with its decomposition, any such interventional outcome can equivalently be thought as an intervention on the latent components of the treatment, i.e. $Y^{(D \leftarrow e_D(x), V \leftarrow e_V(x))}$. Given the structural Equation (1), the expected outcome under a soft intervention $t(X)$ can be written as:

$$\mathbb{E}\left[Y^{(X \leftarrow t(X))}\right] = \mathbb{E}[h(e_D(t(X))) + \eta(U, e_V(t(X)), \epsilon)]$$

We will identify such an intervention via the means of intervention on a learned representation. In particular, given observations, we will learn an encoding $\tilde{e}_D(X) = \tilde{D}$ that respects the properties in Equation (1) (potentially together with a learned encoding $\tilde{e}_V(X) = \tilde{V}$) and a corresponding decoder $\tilde{f}(\tilde{D})$ (potentially also taking as input $\tilde{V}$) that maps the learned encoding back into a high-dimensional treatment. Subsequently, we will estimate an outcome improving direction u in the learned representation space via instrumental variable analysis, viewing $\tilde{D}$ as the "treatment" and Z as the instrument. We will apply the direction u to the learned representations, i.e. $\tilde{D} + \alpha u$, for some scalar intervention

amount α. For ease of notation, we denote with $(\cdot)_{\alpha u}$ to be the corresponding random variable $(\cdot)$ after this intervention. Then decode back to the high-dimensional treatment space $X_{\alpha u} = \tilde{f}(\tilde{D} + \alpha u)$ (potentially $X_{\alpha u} = \tilde{f}(\tilde{D} + \alpha u, \tilde{V})$ if an encoding of V was also learned). This process (depicted also visually in Figure **??** and described algorithmically in Algorithm 1) defines our soft-intervention mapping, formally defined as:

$$t(X) = \tilde{f}(\tilde{e}_D(X) + \alpha u, \tilde{e}_V(X)), \tag{3}$$

with the second input of $\tilde{f}$ omitted if an encoding $\tilde{e}_V$ is not learned.

3. Instrument Guided Representation Learning: The Linear Setting

To make matters more concrete, we will start this analysis with the case where the SEM that is associated with the causal graph in Figure 1c contains only linear relationships:

$$\begin{aligned} D &= A \cdot Z + U, & U &\perp\!\!\!\perp Z \\ X &= B \cdot D + B_\perp \cdot V, & V &\perp\!\!\!\perp Z \\ Y &= \theta^\top D + \eta(U, V, \epsilon), & \epsilon &\perp\!\!\!\perp Z \end{aligned} \tag{4}$$

where B is an $m \times r$ dimensional matrix that maps the k instrument-driven latent decisions D to the observed high-dimensional treatments $X \in \mathbb{R}^m$ *and is assumed to be full column rank*. $B_\perp$ is a matrix whose column space is orthogonal to the column space of B and is also *assumed to be full column rank*. U corresponds to a random vector of latent unobserved confounders that also affect decisions and outcomes. θ is an r dimension vector capturing the direct effects of the latent decisions on the outcome.

We will show that in this setting it is feasible to identify improving interventions, and we can always identify a representation $\tilde{D}$, such that $\tilde{D}$ is an invertible linear transformation of D. Algorithm 2 formalizes this Linear Instrument Regularized Representation (LIRR) procedure and the following theorem formalizes these arguments and provides the outcome improvement guarantee for this intervention.

Theorem 3.1. *Under the linear structural equation model in Equation (4) and assuming $B, B_\perp$ have full column rank and Assumption 2.2 holds, then the representation and intervention produced by the LIRR algorithm satisfy: $\tilde{D} = PD$, for the invertible matrix $P \triangleq \hat{B}^\top B$. Moreover, $\tilde{\theta} = (P^{-1})^\top \theta$ and the interventional outcome satisfies the guaranteed improvement property:*

$$\mathbb{E}[Y_{\alpha u}] = \mathbb{E}[Y] + \alpha \|(P^{-1})^\top \theta\|$$

4. Instrument Guided Representation Learning: The Non-Linear Setting

We will now investigate the general setting introduced in Equation (1). In this non-linear setting, we will require some further assumptions on the latent factors. In particular, we will be assuming that the latent components D are independent of the orthogonal components V that constitute X that are not driven by the instrument. In particular, we will assume the slightly stronger property of joint independence of Z, U, V, which implies that $D \perp\!\!\!\perp V$.

Assumption 4.1. $Z \perp\!\!\!\perp U \perp\!\!\!\perp V$ (jointly independent).

Assumption 4.2. f is a differentiable function with uniformly bounded derivatives.

Assumption 4.3. h is twice differentiable with a bounded second derivative. Moreover, outcome noise is fully exogeneous, i.e. $\epsilon \perp\!\!\!\perp \{Z, U, V\}$

Assumption 4.4. $\mathbb{E}[Z] = 0$ and the support of Z, $\mathcal{Z}$, is an open subset of $\mathbb{R}^k$.

Assumption 4.5 (Bounded Completeness)**.** D is bounded complete for Z, that is, for all bounded real functions h, we have that:

$$\mathbb{E}[h(D)|Z] = 0 \quad \text{a.s.} \quad \Rightarrow \quad h(D) = 0 \quad \text{a.s.}$$

We discuss sufficient conditions for the bounded completeness assumption in the Appendix (Lemma E.7). In particular, it involves characteristic function assumptions that have also been typical in the identifiable latent factor literature (Lu et al., 2021).

Theorem 4.6. *Suppose that the data generating process follows the SEM described in Equation 1, and satisfies Assumptions 2.1 & 2.2 & 4.1 & 4.2 & 4.3 & 4.4 & 4.5. Let $(\tilde{D}, \tilde{V}) := (\tilde{e}_D(X), \tilde{e}_V(X)) = \tilde{e}(X)$ denote the learned representations. Consider encoder-decoder pairs, $\tilde{e}, \tilde{f}$, with perfect reconstruction, i.e. $X = \tilde{f} \circ \tilde{e}(X)$, and full row rank matrix $\tilde{A}$ that minimizes the objective function*

$$\mathbb{E}[\|\tilde{e}_D(X) - \tilde{A}Z\|^2] \tag{5}$$

subject to the following constraints:

- *$\tilde{e}$ is a differentiable function with uniformly bounded derivatives.*
- *$\tilde{D} = \tilde{A}Z + \tilde{U}$ with $\mathbb{E}[\tilde{U}] = 0$.*
- *$\tilde{e}$ is an invertible function when restricted to inputs in the image of f and $\tilde{f} \circ \tilde{e}(x) = x$ for all $x \in Im(f)$.*
- *$\tilde{U} \perp\!\!\!\perp \tilde{V} \perp\!\!\!\perp Z$ (joint Independence)*
- *$\tilde{D} \perp\!\!\!\perp \tilde{V}$*

Then setting $u = \tilde{\theta}/\|\tilde{\theta}\|$, in Algorithm 1, with $\tilde{\theta} = \mathbb{E}[\nabla_{\tilde{D}} \tilde{h}(\tilde{D})]$ and $\tilde{h}$ the solution to the conditional moment restriction problem $\mathbb{E}[Y - \tilde{h}(\tilde{D}) \mid Z] = 0$, we have that:

$$\mathbb{E}[Y_{\alpha u} - Y] = \alpha \|(P^{-1})^\top \mathbb{E}[\nabla_D h(D)]\| + O(\alpha^2)$$

Hence, for small enough step size α, the identified intervention will achieve a positive improvement on the outcome (assuming that $\mathbb{E}[\nabla_D h(D)] \neq 0$).

Instrument Regularized Auto-Encoder To achieve the positive improvement as described in Theorem 4.6, then we need to incorporate loss components that are minimized only when i) e, f reconstruct the input X, ii) $e_D(X)$ is predicted linearly by Z with a full rank matrix A, iii) the residual of this regression $D - AZ - c$, which approximates U, needs to be independent of Z, iv) Z needs to be independent of $e_V(X)$ and v) $e_D(X)$ needs to be independent of $e_V(X)$. While we do not explicitly enforce $\tilde{A}$ to be full row rank, we expect this to be satisfied due to the reconstruction loss and the condition that $\tilde{D} \perp\!\!\!\perp \tilde{V}$. Moreover, note that instead of joint independence of $Z, \tilde{U}, \tilde{V}$ we only enforce pairwise independence of $Z, \tilde{V}$ and $\tilde{D}, \tilde{V}$ for computational reasons. The latter is implied by joint independence of $Z, \tilde{U}, \tilde{V}$ but is not an if and only if property.

We introduce the instrument-regularized auto-encoder loss, which incorporates all these elements:

$$\begin{aligned} \min_{e,f,A} \; & \mathbb{E}\left[\|X - f \circ e(X)\|^2\right] + \lambda \mathbb{E}\left[\|e_D(X) - AZ\|^2\right] \\ & + \mu_1 \mathcal{R}(e_D(X) - AZ, Z) + \mu_2 \mathcal{R}(Z, e_V(X)) \\ & + \mu_3 \mathcal{R}(e_D(X), e_V(X)) \end{aligned} \tag{IRAE}$$

$\mathcal{R}(A, B)$, denotes any regularizer that can be evaluated on a set of n samples and which takes small values the more independent the random variable A is from B. Many examples of such independence-regularizers have been introduced in the literature. Our methodology is agnostic to the exact regularizer used. In our experiments, we used a kernel-based independence test statistic (Gretton et al., 2007).

In experiments, for the purposes of ablation analysis, we will denote with IRAE[0] the variant that contains only the regularization parts that are multiplied by λ, with IRAE[1] the variant that contains the parts that are multiplied by λ, μ_1, with IRAE[2] the variant that contains the parts multiplied by λ, μ_1, μ_2 and IRAE the variant that contains all regularizers.

5. Experimental Evaluation

For linear SEMs, we bench-marked our proposed method against the naive approach of applying PCA on X to recover the latent representations using the top r principal components. Results for the linear experiments are in Appendix F.1. Beyond linear SEMs, we also examined the performance of the IRAE models under non-linear SEMs. We include experimental results for quadratic encoding function f in Appendix F.2. Below we present the results on MNIST experiments.

Table 1. Average Test Improvement Comparison of 4 Methods on MNIST Data (Mean $\pm$ Std)

	Reconstructed	Treated($\alpha = 0.2$)	Treated($\alpha = 0.1$)
Vanilla AE	-0.47 ± 0.02	-0.46 ± 0.03	-0.39 ± 0.06
IRAE[1]	-0.72 ± 0.05	0.15 ± 0.33	$\mathbf{0.26 \pm 0.35}$
IRAE[2]	$\mathbf{-0.23 \pm 0.05}$	1.03 ± 0.48	$\mathbf{1.12 \pm 0.48}$
IRAE	-0.25 ± 0.06	1.04 ± 0.52	$\mathbf{1.19 \pm 0.5}$

MNIST experiment We examine a case where the outcome is determined by the color of MNIST digits. In this experiment, we independently generated instrumental variables Z and confounders U. The color features D are represented as 3-dimensional RGB values determined by both Z and U. The outcome variable is calculated as the sum of R, G, and B values. The observed data X consists of MNIST digit pixels. All except IRAE[2] and IRAE has bottleneck size same as dimension Z and the IRAE[2] and IRAE methods had a bottleneck of size 10. Performance improvement results across 30 seeds are reported in Table 6. Additional visualizations are available in the appendix.

Our experiments reveal important insights about latent space representation and instrumental variables. The vanilla AE, with no specialized latent regularization, produces reconstructed digits that closely resemble the originals, indicating the latent space primarily focuses on digit reconstruction. When IV regression is applied to this representation, no meaningful directional information can be extracted, resulting in no improvement. When we introduce instrument regularization while maintaining the same dimensionality as Z in IRAE[1], the representation is forced to capture more color information at the expense of digit reconstruction. Ideally, we could increase the prediction error weight to infinity to enforce full capture of Z information, but in practice, some digit information remains in the representation. By expanding the latent dimension, we achieve both better digit reconstruction and color information preservation. The larger dimensional space accommodates more digit morphology without needing to compete space with color information, bringing reconstruction error closer to zero while enabling instrumental variables to recover the target direction. The improvement in IRAE[2] is less pronounced than in IRAE due to information leakage between components D and V, resulting in acceptable reconstruction and prediction error but less identifiable direction when IVs are applied solely to the D component. By adding a dependence penalty between D and V, IRAE achieves better improvement.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Ahuja, K., Mahajan, D., Syrgkanis, V., and Mitliagkas, I. Towards efficient representation identification in supervised learning. In *Conference on Causal Learning and Reasoning*, pp. 19–43. PMLR, 2022.

Andreu, O. C., Vlontzos, A., O'Riordan, M., and Gilligan-Lee, C. M. Contrastive representations of high-dimensional, structured treatments. *arXiv preprint arXiv:2411.19245*, 2024.

Bennett, A., Kallus, N., Mao, X., Newey, W., Syrgkanis, V., and Uehara, M. Inference on strongly identified functionals of weakly identified functions. *arXiv preprint arXiv:2208.08291*, 2022.

Bennett, A., Kallus, N., Mao, X., Newey, W., Syrgkanis, V., and Uehara, M. Source condition double robust inference on functionals of inverse problems. *arXiv preprint arXiv:2307.13793*, 2023.

Dong, J., Shi, P., Zheng, F., and Jin, X. Capacity management in networks: A structural estimation approach for hospital inpatient wards.

Dong, J., Shi, P., Zheng, F., and Jin, X. Off-service placement in inpatient ward network: Resource pooling versus service slowdown. *Columbia Business School Research Paper Forthcoming*, 2019.

Du, T., Kanodia, A., Brunborg, H., Vafa, K., and Athey, S. Labor-llm: Language-based occupational representations with large language models. *arXiv preprint arXiv:2406.17972*, 2024.

D'Haultfoeuille, X. On the completeness condition in nonparametric instrumental problems. *Econometric Theory*, 27(3):460–471, 2011.

Gretton, A., Fukumizu, K., Teo, C., Song, L., Schölkopf, B., and Smola, A. A kernel statistical test of independence. *Advances in neural information processing systems*, 20, 2007.

Hälvä, H. and Hyvarinen, A. Hidden markov nonlinear ica: Unsupervised learning from nonstationary time series. In *Conference on Uncertainty in Artificial Intelligence*, pp. 939–948. PMLR, 2020.

Hälvä, H., So, J., Turner, R. E., and Hyvärinen, A. Identifiable feature learning for spatial data with nonlinear ica. In *International Conference on Artificial Intelligence and Statistics*, pp. 3331–3339. PMLR, 2024.

Harada, S. and Kashima, H. Graphite: Estimating individual effects of graph-structured treatments. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pp. 659–668, 2021.

Hyvärinen, A. Independent component analysis: recent advances. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1984):20110534, 2013.

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

Hyvärinen, A., Khemakhem, I., and Morioka, H. Nonlinear independent component analysis for principled disentanglement in unsupervised deep learning. *Patterns*, 4(10), 2023.

Hyvärinen, A., Khemakhem, I., and Monti, R. Identifiability of latent-variable and structural-equation models: from linear to nonlinear. *Annals of the Institute of Statistical Mathematics*, 76(1):1–33, 2024.

Jin, J. and Syrgkanis, V. Learning causal representations from general environments: Identifiability and intrinsic ambiguity. *arXiv preprint arXiv:2311.12267, (to appear at NeurIPS24)*, 2023.

Kelly, B. T., Pruitt, S., and Su, Y. Instrumented principal component analysis. *Available at SSRN 2983919*, 2020.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. Variational autoencoders and nonlinear ica: A unifying framework. In *International conference on artificial intelligence and statistics*, pp. 2207–2217. PMLR, 2020.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Lopez, R., Li, C., Yan, X., Xiong, J., Jordan, M., Qi, Y., and Song, L. Cost-effective incentive allocation via structured counterfactual inference. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 4997–5004, 2020.

Lu, C., Wu, Y., Hernández-Lobato, J. M., and Schölkopf, B. Invariant causal representation learning for out-of-distribution generalization. In *International Conference on Learning Representations*, 2021.

Monti, R. P., Zhang, K., and Hyvärinen, A. Causal discovery with general non-linear relationships using non-linear ica. In *Uncertainty in artificial intelligence*, pp. 186–195. PMLR, 2020.

Muandet, K., Jitkrittum, W., and Kübler, J. Kernel conditional moment test via maximum moment restriction. In *Conference on Uncertainty in Artificial Intelligence*, pp. 41–50. PMLR, 2020.

Nabi, R., McNutt, T., and Shpitser, I. Semiparametric causal sufficient dimension reduction of multidimensional treatments. In *Uncertainty in Artificial Intelligence*, pp. 1445–1455. PMLR, 2022.

Qin, J., Chan, C. W., Dong, J., Homma, S., and Ye, S. Waiting online versus in-person: An empirical study on outpatient clinic visit incompletion. 2023.

Rao, C. R. The use and interpretation of principal component analysis in applied research. *Sankhyā: The Indian Journal of Statistics, Series A*, pp. 329–358, 1964.

Sabatier, R., Lebreton, J.-D., and Chessel, D. Principal component analysis with instrumental variables as a tool for modelling composition data. *Multiway data analysis*, pp. 341–352, 1989.

Saengkyongam, S., Rosenfeld, E., Ravikumar, P., Pfister, N., and Peters, J. Identifying representations for intervention extrapolation. *arXiv preprint arXiv:2310.04295*, 2023.

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

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

Vafa, K., Athey, S., and Blei, D. M. Estimating wage disparities using foundation models. *arXiv preprint arXiv:2409.09894*, 2024.

Wang, C. Counterfactual and synthetic control method: Causal inference with instrumented principal component analysis. *arXiv preprint arXiv:2408.09271*, 2024.

Y Takane, M. H. Constrained principal component analysis: A comprehensive theory. 2001. URL https://doi.org/10.1007/s002000100081.

A. Further Related Work

In this section we provide a more discussion on related work that is not covered in the main text.

Identifying Representations for Intervention Extrapolation Similar to our work, Saengkyongam et al. proposed the *Rep4Ex* approach which tries to solve the task of interventional outcome prediction by identifying the SCM. Importantly, although they work with a similar SCM as we do (Equation 1), the level of intervention differs - our work considers interventions on the latent treatment space (D), while Saengkyongam et al. considers intervening on Z (using notations in Equation 1). Moreover, our work is motivated by the presence of unobserved confounding between the latent representation of the treatment and the outcome, whereas their work is motivated by the need to extrapolate to unseen interventions, while the treatment that they consider is fully exogenous. Like our approach, they employ autoencoders to learn latent representations from potentially high-dimensional observed features, but use maximum moment restriction (MMR) regularization (Muandet et al., 2020) to enforce the constraint $E[e_D(X) - AZ|Z] = 0$. This can be achieved when $E[e_D(X) - AZ] = 0$ and $e_D(X) - AZ \perp\!\!\!\perp Z$, corresponding to our λ and μ_1 term in Equation (IRAE). Additionally, while *Rep4Ex* assumes a deterministic mixing function from the latent representation to the observables X, our method explicitly handles noisy observations of X through $e_V(X)$, which allows for broader generalization.

Representation Learning with Instrumental Variables Prior work on that combines elements of representation learning with elements of instrumental variable analysis is limited and confined to linear methods. Rao and Sabatier et al. described a procedure of performing principal component analysis (PCA) of a response variable with respect to its instruments. Y Takane studied constrained principal component analysis, which takes external information into consideration during dimensional reduction (Y Takane, 2001). More recently, Kelly et al. and Wang incorporates instrumental variables in estimating factor models that improves rate of convergence and avoid overfiting for high-dimensional data (Kelly et al., 2020),(Wang, 2024). The desiderata in all of these works are very different from identifying dimensions of variation that align with the instruments so that causal effects can be identified by downstream IV analysis.

Dimensionality Reduction for High Dimensional Treatments When learning a representation for the treatment, it is important for the learned representation to capture all causal factors so that the causal relationship is preserved for downstream estimation tasks like treatment effect estimation. Nabi et al.utilize semi-parametric inference theory for structural models to provide a generalized the sufficient dimension reduction approach for learning lower-dimensional representation for treatment, while capturing the relationship between the treatment and the mean counterfactual outcome. Andreu et al. employed a contrastive approach to learn a representation of the high-dimensional treatments. These works studied settings that did not involve the presence of unobserved confounders of the treatment, while we focus on heavily confounded high dimensional structured treatments. Moreover, in these works, the selection of causally relevant factors are guided by the outcome, where as we take an inherently different approach that learns the latent representations using auxiliary information from instrumental variables instead of the treatment.

Causal Represenation Learning Our work is also related to the literature on learning non-linear disentangled representations and causal representation learning (Hyvärinen & Oja, 2000; Hyvärinen, 2013; Khemakhem et al., 2020; Hälvä & Hyvarinen, 2020; Monti et al., 2020; Schölkopf et al., 2021a; Ahuja et al., 2022; Hyvärinen et al., 2023; Jin & Syrgkanis, 2023; Hyvärinen et al., 2024; Hälvä et al., 2024). However, the focus of this line of work has primarily been on discovering causal structure in data (Schölkopf et al., 2021b), rather than constructing representations for downstream causal tasks. Our work is closely related to the identifiable VAE (iVAE) (Khemakhem et al., 2020). The instrument can be viewed as the auxiliary information that can guide non-linear latent factor analysis. However, a crucial difference of our work is that we view the instrument Z as only privileged information that is available only when estimating the causal effects and not when performing interventions. Hence, crucially we want our encoder to only take as input the treatment X and not the instrument Z. Moreover, our desiderata is not the discovery of the true latent factors, but solely the discovery of valid decompositions of the treatment for downstream IV analysis. This allows us to relax many of the assumptions that are prevalent in this line of work.

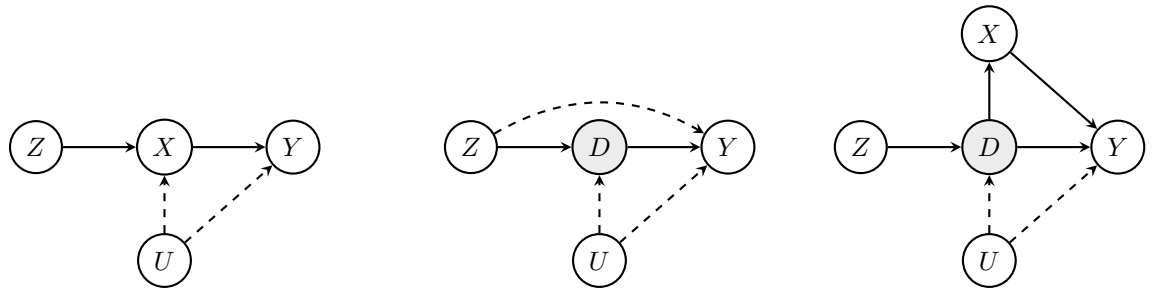
Independence Conditions In our work, we show that independence between certain variables (for more details, see Theorem E.1) is desirable for identification. We enforce the independence condition by incorporating a Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al., 2007) regularizer. This approach has also been adopted in prior research: for instance, Lopez et al. employed HSIC regularization to mitigate bias in observational datasets for applications in counterfactual policy optimization, while Harada & Kashima use it to learn a representations of the treatment that is independent with the target individual in order to mitigate selection bias.

B. Discussion on Valid Treatment Representations

We consider a setting where we are given data that contain samples of variables (Z, X, Y), where X is a high-dimensional "treatment" variable, Y is a scalar outcome of interest and Z is a low-dimensional vector of instruments. The treatment X is heavily confounded via unobserved confounding variables U that have a causal influence on the value of X and also on Y, as depicted in Figure 1a.

Our goal is to learn a latent representation of the highly confounded, high-dimensional treatment, so as to perform instrumental variable analysis on this learned representation and identify an outcome-improving direction of intervention in representation space and hence subsequently also in the original treatment space. Naive representation learning approaches for the treatment run the risk of an omitted variable problem that can invalidate the downstream causal analysis based on instrumental variables.

Causal analysis using instrumental variables crucially assumes that the instrument Z, the treatment X, and the outcome Y respect the causal graph depicted in Figure 1a. In particular, the instrument Z is assumed to only affect outcome Y through its effect on treatment X. When the high-dimensional treatment X is replaced by a learned representation D, we run the risk that the part of X that is not represented in D contains elements that are correlated with both the instrument Z and the outcome Y. As a result, D no longer absorbs the entire effect of the instrumental variable Z on the outcome Y. This creates causal pathways from the instrument Z to the outcome Y that do not flow through the representation D, as shown in Figure 1b. Therefore, we need to regularize the representation learning process to ensure that the causal influence through these omitted paths is minimal.



(a) Instrumental variable causal graph, with instruments Z, high-dimensional treatment X, outcome Y, unobserved confounders U.

(b) Causal graph when high-dimensional treatment X is replaced by learned representation D.

(c) Causal graph that an ideal representation D of the high-dimensional treatment X would satisfy.

Figure 1. Omitted variable bias in instrumental variable analysis with learned treatment representations.

An ideal latent representation D should satisfy the causal graph depicted in Figure 1c. In particular, the instrument Z should not have a causal effect on X that is not absorbed by the latent representation D. If the representation encodes all outcome-relevant information, then a direct edge from X to Y should not exist. However, the existence of such an edge does not invalidate the downstream instrumental variable analysis, and hence, it is not essential to exclude it.

C. Algorithms

In the section we present the algorithms of the proposed methods in the paper.

C.1. General Algorithm

Algorithm 1 Intervention in Latent Representation Space and evaluation

1: **Autoencoder fitting.** Learn encoder $\tilde{e}$ and decoder $\tilde{f}$ of X and using observed data (Z, X, Y).
2: **IV analysis.** Identify causal model $\tilde{h}(\tilde{D})$ using IV regression analysis with instrument Z, treatment $\tilde{D} \triangleq \tilde{e}_D(X)$ and outcome Y. Calculate average causal derivative $u = \mathbb{E}[\nabla_D \tilde{h}(\tilde{D})]$.
3: **Encode.** Transform X into latent representation $\tilde{D}$ using learned encoder $\tilde{D} = \tilde{e}_D(X)$
4: **Perturb.** Apply perturbation in the latent space: $\tilde{D}_{\alpha u} = \tilde{D} + \alpha u$ where α is a scalar factor controlling perturbation magnitude.
5: **Decode.** Map perturbed latent representation $\tilde{D}_{\alpha u}$ back to input space: $X_{\alpha u} = \tilde{f}(\tilde{D}_{\alpha u})$ (or $X_{\alpha u} = \tilde{f}(\tilde{D}_{\alpha u}, \tilde{e}_V(X))$ if the learned encoder also learns a representation of V).
6: **Evaluate.** Apply the true decomposition $e(X_{\alpha u}) = (D_{\alpha u}, V_{\alpha u})$ and evaluate outcome under intervention: $Y_{\alpha u} = h(D_{\alpha u}) + \eta(U, V_{\alpha u}, \epsilon)$.
7: Compare average original outcome Y to average perturbed outcome $Y_{\alpha u}$.

C.2. Algorithm for the Linear SEM

Algorithm 2 Linear Instrument Regularized Representation (LIRR) and Intervention

1: **Input:** magnitude of intervention α
2: Run linear regression of X on $Z \in \mathbb{R}^k$, to estimate a coefficient matrix C
3: Calculate the *thin* SVD decomposition of $C = \mathcal{U}\Sigma\mathcal{V}^\top$, keeping only the top k singular values
4: Define $\hat{B} = \mathcal{U}$ and $\hat{A} = \Sigma\mathcal{V}^\top$ and $\tilde{D} = \tilde{e}_D(X) = \hat{B}^\top X$
5: Run linear IV regression solving moment $\mathbb{E}[Z(Y - \tilde{\theta}^\top \tilde{D})] = 0$
6: Let $u = \tilde{\theta}/\|\tilde{\theta}\|$ and perform intervention on learned representation space $\tilde{D}_{\alpha u} = \tilde{D} + \alpha u$
7: Encode back to X-space intervention of $X_{\alpha u} = X + \alpha \hat{B}^\top u$

D. Identification Under Linear SEM

Before proving the main theorem, we first present some useful lemma.

Lemma D.1. *Suppose A is a $n \times k$ matrix with full row rank ($k > n$), and B is a $m \times n$ matrix, with full column rank ($m > n$). Then the columns of $C = BA$ spans the same space as the columns of B.*

Proof of Lemma D.1. Let $\mathcal{R}(\cdot)$ denote the column space of a matrix.

For any $x \in \mathcal{R}(B)$, there exist vector y such that $x = By$. Since A is full row rank, we know that $AA^+ = I_n$, and $x = By = BAA^+y = C(A^+y)$. Therefore $x \in \mathcal{R}(C)$, so $\mathcal{R}(B) \subseteq \mathcal{R}(C)$.

Similarly, for any $x \in \mathcal{R}(C)$, there exist vector y such that $x = BAy = B(Ay)$. So $x \in \mathcal{R}(B)$, and we have $\mathcal{R}(C) \subseteq \mathcal{R}(B)$.

Together, we have $\mathcal{R}(C) = \mathcal{R}(B)$.

□

Now we proceed to prove Theorem 3.1.

Proof of Theorem 3.1. From the linear SEM 4, we have that:

$$X = BAZ + B_\perp V + BU$$

Then taking the conditional expectation over Z, we have:

$$\begin{aligned}
\mathbb{E}[X|Z] &= BAZ + \mathbb{E}[B_\perp V + BU] \\
&= BAZ + \mathbb{E}[B_\perp \mathbb{E}[V|Z]] + \mathbb{E}[B\mathbb{E}[U|Z]] \\
&= BAZ + B_\perp \mathbb{E}[V] + B\mathbb{E}[U] && \text{(Since } V \perp\!\!\!\perp Z \text{ and } U \perp\!\!\!\perp Z\text{)} \\
&= BAZ
\end{aligned}$$

Thus $C := BA$ can be uniquely identified as the solution to the linear regression problem, regressing X on Z. Consider the SVD decomposition of $C = \mathcal{U}\Sigma\mathcal{V}^\top$. Let $\hat{B} = \mathcal{U}$, and $\hat{A} = \Sigma\mathcal{V}^\top$. Then by Lemma D.1, we have that the columns of $\hat{B}$ spans the same space as the columns of B. In other words, there exist an invertible change of basis matrix P such that $B = \hat{B}P$. Since $\hat{B}$ is orthonormal (by construction of SVD), we have that $\hat{B}^T\hat{B} = I_r$, and $P = \hat{B}^T B$. As a result, we also have:

$$\begin{aligned}
D = B^+X &= (B^TB)^{-1}B^TX \\
&= (P^T\hat{B}^T\hat{B}P)^{-1}P^T\hat{B}^TX \\
&= (P^TP)^{-1}P^T\hat{B}^TX \\
&= P^{-1}\hat{B}^TX = P^{-1}\tilde{D}
\end{aligned}$$

Next, we show that $\tilde{\theta} = (P^{-1})^T\theta$. The LIRR algorithm solves for $\tilde{\theta}$ from the following moment equation:

$$\begin{aligned}
0 &= \mathbb{E}[Z(Y - \tilde{\theta}^T\hat{D})] \\
&= \mathbb{E}[Z(\theta^TD + \eta(V,U,\epsilon) - \hat{\theta}^TPD)] \\
&= \mathbb{E}[Z(\theta^TD - \tilde{\theta}^TPD)] && \text{(Since } U,V,\epsilon \perp\!\!\!\perp Z \text{ and } \mathbb{E}[\eta(U,v,\epsilon)] = 0\text{)} \\
&= \mathbb{E}[ZD^T](\theta - P^T\tilde{\theta}) \\
&= \mathbb{E}[Z(Z^TA^T + U^T)](\theta - P^T\tilde{\theta}) \\
&= \mathbb{E}[ZZ^T]A^T(\theta - P^T\tilde{\theta})
\end{aligned}$$

Since the instruments are not co-linear, we have that $\mathbb{E}[ZZ^T] \succ 0$, i.e. $\mathbb{E}[ZZ^T]$ is invertible. Thus $\mathbb{E}[ZZ^T]A^T(\theta - P^T\tilde{\theta}) = 0$ if and only if $A^T(\theta - P^T\tilde{\theta}) = 0$. Since A^T has full column rank, then by the Rank-Nullity theorem, the null space of $A^T = 0$. Together, this shows that $\tilde{\theta} = (P^{-1})^T\theta$ is the unique solution to the moment condition.

Lastly, we show that the intervened outcome is guaranteed improvement in expectation. Consider an intervention in the direction of $u = \tilde{\theta}/\|\tilde{\theta}\|$ in the $\tilde{D}$ space, this maps to an intervention in the D space as:

$$\begin{aligned}
e_D(t(X)) = B^+t(X) &= D + \alpha B^+\hat{B}\tilde{\theta} \\
&= D + \alpha P^{-1}\frac{\tilde{\theta}}{\|\tilde{\theta}\|} = D + \alpha P^{-1}\frac{(P^{-1})^\top\theta}{\|(P^{-1})^\top\theta\|}
\end{aligned}$$

Since, we intervene only in D, $e_V(t(X)) = V$. Then, we can compute the intervened outcome:

$$\begin{aligned}
\mathbb{E}[Y_{\alpha u}] &= \mathbb{E}[\theta^Te_D(t(X)) + \eta(e_V(t(X)),U,\epsilon)] \\
&= \mathbb{E}[\theta^Te_D(t(X))] && \text{(}e_V(t(X)) = V\text{, and } \mathbb{E}[\eta(U,v,\epsilon)] = 0\text{)} \\
&= \mathbb{E}\left[\theta^T\left(D + \alpha P^{-1}\frac{(P^{-1})^\top\theta}{\|(P^{-1})^\top\theta\|}\right)\right] \\
&= \mathbb{E}[\theta^TD + \alpha\|(P^{-1})^\top\theta\|] = \mathbb{E}[Y] + \alpha\|(P^{-1})^\top\theta\|
\end{aligned}$$

□

E. Non-linear Identification

Theorem E.1. *Suppose that the data generating process follows the SEM described in Equation 1, and satisfies Assumptions 2.1 & 2.2 & 4.1 & 4.2 & 4.4 & 4.5. Let $(\tilde{D}, \tilde{V}) := (\tilde{e}_D(X), \tilde{e}_V(X)) = \tilde{e}(X)$ denote the learned representations. Consider encoder-decoder pairs with perfect reconstruction, i.e. $X = \tilde{f} \circ \tilde{e}(X)$. Then, for the solution $\tilde{e}$, $\tilde{f}$, and full row rank matrix $\tilde{A}$ that minimizes the objective function*

$$\mathbb{E}[\|\tilde{e}_D(X) - \tilde{A}Z\|^2] \tag{6}$$

subject to the following constraints:

- *$\tilde{e}$ is a differentiable function with uniformly bounded derivatives.*
- *$\tilde{A}$ has full row rank.*
- *$\tilde{D} = \tilde{A}Z + \tilde{U}$ with $\tilde{U} \perp\!\!\!\perp Z$ and $\mathbb{E}[\tilde{U}] = 0$.*

we have that, with probability 1, $\tilde{D} = PD$ and $\tilde{U} = PU$, for $P = \tilde{A}A^+$. Moreover, the matrix P is invertible.

Lemma E.2. *Suppose the assumptions of Theorem E.1 hold, and additionally impose the following constraints on the learned functions $\tilde{e}, \tilde{f}, \tilde{A}$ that minimize the objective in Equation 6:*

- *$\tilde{D} \perp\!\!\!\perp \tilde{V}$*
- *$\tilde{e}$ is an invertible function when restricted to inputs in the image of f and $\tilde{f} \circ \tilde{e}(x) = x$ for all $x \in Im(f)$.*

Then, it must also hold with probability 1 that:

- *$\tilde{V} = q_2(D, V) = (\tilde{e} \circ f)_2(D, V)$[2] with the property that for all $d, d' \in \mathcal{D}$:*

$$Law(q_2(d, V)) = Law(q_2(d', V)).$$

- *$V = q_2^{-1}(D, V) = (e \circ \tilde{f})_2(\tilde{D}, \tilde{V})$ with the property that for all $d, d' \in \mathcal{D}$:*

$$Law(q_2^{-1}(d, \tilde{V})) = Law(q_2^{-1}(d', \tilde{V})).$$

Remark E.3. Note that the assumptions that $\mathbb{E}[Z] = 0$ and $\mathbb{E}[ZZ'] \succ 0$ are without loss of generality as we can always pre-process Z by centering it and removing co-linear instruments. Moreover, in practice the assumption that $\tilde{D} = \tilde{A}Z + \tilde{U}$, with $\tilde{U} \perp\!\!\!\perp Z$ and $\mathbb{E}[\tilde{U}] = 0$ can be achieved by minimizing a square loss with an intercept, i.e.

$$\min_{e, f, A, c: e, f \text{invertible}, e \circ f = \text{identity}} \mathbb{E}[\|e_D(X) - AZ - c\|^2]$$

and then defining $\tilde{D} = \tilde{e}_D(X) \triangleq e_D(X) - c$, $\tilde{f} = f + c$.

Subsequently, we identify an intervention as described in Algorithm 1. In particular, we will run an IV analysis, with Z as the instrument, $\tilde{D}$ as the treatment, and Y as the outcome, to estimate a causal model in representation space by finding a solution to the conditional moment restrictions:

$$\mathbb{E}[Y - \tilde{h}(\tilde{D}) \mid Z] = 0 \tag{7}$$

Note that since $\tilde{D} = PD$ and since $\mathbb{E}[Y \mid Z] = \mathbb{E}[h(D) \mid Z]$, we have by the completeness assumption that:

$$\mathbb{E}[h(D) - \tilde{h}(PD) \mid Z] = 0 \Rightarrow h(D) = \tilde{h}(PD) \text{ a.s.} \implies h(P^{-1}\tilde{D}) = \tilde{h}(\tilde{D}) \text{ a.s.}$$

If for instance, h is assumed to be linear, then $\tilde{h}$ is also a linear function and it suffices to run a linear instrumental variable analysis (e.g. two-stage-least-squares). If h is non-linear, then we calculate the average derivative of $\tilde{h}$, i.e.

$$\tilde{\theta} = \mathbb{E}[\nabla_{\tilde{D}} \tilde{h}(\tilde{D})] = (P^{-1})^\top \mathbb{E}[\nabla_D h(D)]$$

[2] With $(\tilde{e} \circ f)_2$ we denote the V component of the output of the function $\tilde{e} \circ f_0$.

and perform the intervention

$$u = \tilde{\theta}/\|\tilde{\theta}\|$$

as described in Algorithm 1. In finite samples, recently introduced doubly robust methods for estimation of average derivatives of solutions to non-parametric IV problems can be used (Bennett et al., 2022; 2023).

Theorem E.4. *Assume that:*

$$Y = h(D) + \eta(U, V, \epsilon), \quad \epsilon \perp\!\!\!\perp \{Z, U, V\}$$

and that h is twice differentiable with a bounded second derivative. Let $\tilde{e}, \tilde{f}, \tilde{A}$ be an optimal solution as prescribed in Lemma E.2 with the extra constraint that:

$$\tilde{U} \perp\!\!\!\perp \tilde{V} \perp\!\!\!\perp Z \qquad \text{(joint independence)}$$

and assume that the assumptions of Lemma E.2 are satisfied. Then setting $u = \tilde{\theta}/\|\tilde{\theta}\|$, in Algorithm 1, with $\tilde{\theta} = \mathbb{E}[\nabla_{\tilde{D}} \tilde{h}(\tilde{D})]$ and $\tilde{h}$ the solution to the conditional moment restriction problem in Equation (7), we have that:

$$\mathbb{E}[Y_{\alpha u} - Y] = \alpha \|(P^{-1})^\top \mathbb{E}[\nabla_D h(D)]\| + O(\alpha^2)$$

Hence, for small enough step size α, the identified intervention will achieve a positive improvement on the outcome (assuming that $\mathbb{E}[\nabla_D h(D)] \neq 0$).

E.1. Proof of Non-linear Identification

Proof of Theorem E.1. By definition of $(\tilde{D}, \tilde{V})$, we have:

$$(\tilde{D}, \tilde{V}) = \tilde{e}(X) = \tilde{e} \circ f(D, V) =: q(D, V)$$

Denote with $q_1(D, V)$ the $\tilde{D}$ component of the output of q and q_2 the $\tilde{V}$ component.

Since we have that $\tilde{D} = \tilde{A}Z + \tilde{U}$, with $\tilde{U} \perp\!\!\!\perp Z$ and $\mathbb{E}[\tilde{U}] = 0$, we can write:

$$\mathbb{E}[\tilde{D} \mid Z = z] = \mathbb{E}[\tilde{A}Z + \tilde{U} \mid Z = z] = \tilde{A}z$$

Moreover:

$$\begin{aligned}
\mathbb{E}[\tilde{D} \mid Z = z] &= \mathbb{E}[q_1(D, V) \mid Z = z] \\
&= \mathbb{E}[q_1(Az + U, V) \mid Z = z] \\
&= \mathbb{E}[q_1(Az + U, V)] && (Z \perp\!\!\!\perp \{U, V\}) \\
&= \mathbb{E}_U[\mathbb{E}_V[q_1(Az + U, V)]] && (U \perp\!\!\!\perp V) \\
&= \mathbb{E}_U[\tilde{q}_1(Az + U)] && (\tilde{q}_1(d) \triangleq \mathbb{E}_V[q_1(d, V)]) \\
&= \mathbb{E}[\tilde{q}_1(Az + U)]
\end{aligned}$$

Thus we can conclude that:

$$\tilde{A}z = \mathbb{E}[\tilde{D} \mid Z = z] = \mathbb{E}[\tilde{q}_1(Az + U)]$$

Since this holds for all $z \in \mathcal{Z}$ and since $\mathcal{Z}$ is an open set, we can take the derivative with respect to z, to derive:

$$\forall z \in \mathcal{Z} : \tilde{A} = \partial_z \mathbb{E}[\tilde{q}_1(Az + U)]$$

Since q_1 is continuously differentiable, the same holds for $\tilde{q}_1$ and therefore we can exchange the order of differentiation and expectation:

$$\tilde{A} = \mathbb{E}[\partial_z \tilde{q}_1(Az + U)]$$

Letting $\tilde{q}_1^{(1)}$ denote the gradient of the function $\tilde{q}_1(d)$, we can write by the chain rule:

$$\begin{aligned}
\tilde{A} &= \mathbb{E}[\tilde{q}_1^{(1)}(Az+U)A] \\
&= \mathbb{E}[\tilde{q}_1^{(1)}(Az+U)]A \\
&= \mathbb{E}[\tilde{q}_1^{(1)}(Az+U) \mid Z=z]A && (Z \perp\!\!\!\perp U) \\
&= \mathbb{E}[\tilde{q}_1^{(1)}(AZ+U) \mid Z=z]A \\
&= \mathbb{E}[\tilde{q}_1^{(1)}(D) \mid Z=z]A
\end{aligned}$$

Since A is full row rank, we have that AA^+ is invertible. Thus we can write:

$$\tilde{A}A^+ = \mathbb{E}[\tilde{q}^{(1)}(D) \mid Z=z]$$

or equivalently:

$$\forall z \in \mathcal{Z} : \mathbb{E}[\tilde{q}_1^{(1)}(D) - \tilde{A}A^+ \mid Z=z] = 0$$

By the bounded completeness assumption and since both $\tilde{A}A^+$ and $\tilde{q}_1^{(1)}$ are bounded, the latter implies that:

$$\forall d \in \mathcal{D} : \tilde{q}_1^{(1)}(d) = \tilde{A}A^+$$

or equivalently that:

$$\tilde{q}_1(d) = \tilde{A}A^+ d + \tilde{\nu}$$

for some constant vector ν. Moreover,

$$\begin{aligned}
\mathbb{E}[\tilde{D}] &= \mathbb{E}[\tilde{q}_1(D)] \\
&= \tilde{A}A^+\mathbb{E}[D] + \tilde{\nu} \\
&= \tilde{A}A^+A\mathbb{E}[Z] + \tilde{A}A^+\mathbb{E}[U] + \tilde{\nu} \\
&= \tilde{\nu}
\end{aligned}$$

But we also have $\mathbb{E}[\tilde{D}] = A\mathbb{E}[Z] + \mathbb{E}[\tilde{U}] = 0$. Hence, we have that $\tilde{\nu} = 0$. Thus:

$$\forall d \in \mathcal{D} : \tilde{q}_1(d) = \tilde{A}A^+ d$$

Next, we argue that $\tilde{A}A^+$ is an invertible matrix. Note that:

$$\begin{aligned}
\mathbb{E}[\tilde{D}Z^\top] &= \mathbb{E}[(\tilde{A}Z + \tilde{U})Z^\top] \\
&= \tilde{A}\mathbb{E}[ZZ^\top] && (\tilde{U} \perp\!\!\!\perp Z, \mathbb{E}[Z]=0, \mathbb{E}[\tilde{U}=0])
\end{aligned}$$

Moreover:

$$\begin{aligned}
\mathbb{E}[\tilde{D}Z^\top] &= \mathbb{E}[q_1(D,V)Z^\top] \\
&= \mathbb{E}[q_1(AZ+U,V)Z^\top] \\
&= \mathbb{E}[\mathbb{E}[q_1(AZ+U,V) \mid Z,U]Z^\top] \\
&= \mathbb{E}[\tilde{q}_1(AZ+U)Z^\top] && (Z \perp\!\!\!\perp U \perp\!\!\!\perp V) \\
&= \mathbb{E}[\tilde{q}_1(D)Z^\top] \\
&= \mathbb{E}[(\tilde{A}A^+D)Z^\top] \\
&= \tilde{A}A^+\mathbb{E}[DZ^\top] \\
&= \tilde{A}A^+\mathbb{E}[(AZ+U)Z^\top] \\
&= \tilde{A}A^+A\mathbb{E}[ZZ^\top] && (Z \perp\!\!\!\perp U, \mathbb{E}[U]=0)
\end{aligned}$$

Thus we have concluded that:

$$\tilde{A}\mathbb{E}[ZZ^\top] = \mathbb{E}[\tilde{D}Z^\top] = \tilde{A}A^+A\mathbb{E}[ZZ^\top]$$

Since $\mathbb{E}[ZZ^\top]$ is assumed to be invertible, the latter implies that:

$$\tilde{A} = \tilde{A}A^+A$$

By Lemma E.5, since $\tilde{A}$ and A have full row rank, the row span of $\tilde{A}$ is equal to the row span of A and the matrix $\tilde{A}A^+$ is invertible.

We have thus concluded that:

$$\forall d \in \mathcal{D} : \tilde{q}_1(d) = \tilde{A}A^+d$$

and $\tilde{A}A^+$ is invertible.

Consider any solution with perfect encoder-decoder pair $(\tilde{e}, \tilde{f})$, and $\tilde{A}$ that satisfies the conditions of the theorem and minimizes the objective function:

$$\mathbb{E}[\|\tilde{e}_D(X) - \tilde{A}Z\|^2] = \mathbb{E}[\|\tilde{D} - \tilde{A}Z\|^2]$$

For any feasible solution, we can decompose this objective into two components by centering around

$$\mu_{\tilde{A}}(d) \triangleq \tilde{A}A^+d$$

i.e.:

$$\begin{aligned}\mathbb{E}[\|\tilde{D} - \tilde{A}Z\|^2] &= \mathbb{E}[\|\tilde{D} - \mu_{\tilde{A}}(D) + \mu_{\tilde{A}}(D) - \tilde{A}Z\|^2] \\ &= \mathbb{E}[\|\tilde{D} - \mu_{\tilde{A}}(D)\|^2 + \|\mu_{\tilde{A}}(D) - \tilde{A}Z\|^2] + 2\mathbb{E}[(\tilde{D} - \mu_{\tilde{A}}(D))^\top(\mu_{\tilde{A}}(D) - \tilde{A}Z)]\end{aligned}$$

Consider the inner product term. Since we have that:

$$\begin{aligned}\mathbb{E}[\tilde{D} - \mu_{\tilde{A}}(D) \mid D, Z] &= \mathbb{E}[q_1(D, V) - \tilde{q}_1(D) \mid D, Z] \\ &= \mathbb{E}[q_1(D, V) \mid D, Z] - \tilde{q}_1(D) \\ &= \mathbb{E}[q_1(D, V) \mid D, Z] - \mathbb{E}[q_1(D, V) \mid D]\end{aligned}$$

Since $Z \perp\!\!\!\perp U \perp\!\!\!\perp V$, we have by Lemma E.6 that $Z \perp\!\!\!\perp V \mid \mathbb{1}\{AZ + U = d\}$:

$$\mathbb{E}[q_1(D, V) \mid D = d, Z] = \mathbb{E}[q_1(d, V) \mid D = d, Z] = \mathbb{E}[q_1(d, V) \mid D = d] = \mathbb{E}[q_1(D, V) \mid D = d]$$

Thus:

$$\mathbb{E}[\tilde{D} - \mu_{\tilde{A}}(D) \mid D, Z] = 0$$

From this we conclude that for any feasible solution $\tilde{e}, \tilde{f}, \tilde{A}$, we have that the objective can be decomposed as:

$$\begin{aligned}\mathbb{E}[\|\tilde{e}_D(X) - \tilde{A}Z\|^2] &= \mathbb{E}[\|\tilde{D} - \mu_{\tilde{A}}(D)\|^2] + \mathbb{E}[\|\mu_{\tilde{A}}(D) - \tilde{A}Z\|^2] \\ &= \mathbb{E}[\|q_1(D, V) - \mu_{\tilde{A}}(D)\|^2] + \mathbb{E}[\|\mu_{\tilde{A}}(D) - \tilde{A}Z\|^2]\end{aligned}$$

Suppose that with positive probability, we have that $q_1(D, V) \neq \mu_{\tilde{A}}(D) = \tilde{A}A^+D$. Then we have that:

$$\mathbb{E}[\|q_1(D, V) - \mu_{\tilde{A}}(D)\|^2] > 0$$

In this case, we will provide an alternative feasible solution, which achieves smaller objective than $\tilde{e}, \tilde{f}, \tilde{A}$. Consider the solution:

$$\begin{aligned}\tilde{e}'(x) &= (\tilde{A}A^+e_D(x), e_V(x)) \\ \tilde{f}'(d, v) &= f((\tilde{A}A^+)^{-1}d, v)\end{aligned}$$

Note that we used the fact that for any feasible solution, we have already shown that $\tilde{A}A^+$ is invertible. Moreover, note that for this solution we have that, $\tilde{e}', \tilde{f}'$ is invertible, since e, f is invertible and $\tilde{A}A^+$ is invertible. Finally,

$$\tilde{f}' \circ \tilde{e}'(x) = x$$
$$\tilde{e}'(f(d,v)) = (\tilde{A}A^+ e_D(f(d,v)), e_V(f(d,v))) = (\tilde{A}A^+ d, v)$$

Thus:

$$\tilde{D} = \tilde{e}'_D(X) = \tilde{e}'_D(f(D,V)) = \tilde{A}A^+ D = \tilde{A}A^+ AZ + \tilde{A}A^+ U = \tilde{A}Z + \tilde{A}A^+ U$$

Where we used the fact that we have already shown (in the proof of Theorem E.1) that $\tilde{A}A^+ A = \tilde{A}$. Thus, we also have that:

$$\tilde{D} = \tilde{A}Z + \tilde{U}$$

where $\tilde{U} = \tilde{A}A^+ U$ and satisfies $\tilde{U} \perp\!\!\!\perp Z$ and $\mathbb{E}[\tilde{U}] = 0$.

Therefore, this new solution is a feasible solution. Moreover, since under this solution we have that $\tilde{D} = \tilde{A}A^+ D = \mu_{\tilde{A}}(D)$, the first part of the objective vanishes and the objective takes the value:

$$\mathbb{E}[\|\mu_{\tilde{A}}(D) - \tilde{A}Z\|^2] < \mathbb{E}[\|q_1(D,V) - \mu_{\tilde{A}}(D)\|^2] + \mathbb{E}[\|\mu_{\tilde{A}}(D) - \tilde{A}Z\|^2]$$

contradicting the optimality of the original solution.

Thus we have derived that for any optimal feasible solution, it must hold that with probability 1:

$$\tilde{D} = q_1(D,V) = \mu_{\tilde{A}}(D) = \tilde{A}A^+ D \tag{8}$$

with $\tilde{A}A^+$ an invertible matrix. Moreover, this implies that $\tilde{U} = \tilde{D} - \tilde{A}Z = \tilde{A}A^+ U$. □

Proof of Lemma E.2. In this proof, we argue about the properties of the second part of the function q. Note that since $\tilde{D} \perp\!\!\!\perp \tilde{V}$ and since $\tilde{D} = PD$, for some invertible P, with probability 1, we have that $\tilde{V} \perp\!\!\!\perp D$. Thus:

$$q_2(D,V) \equiv \tilde{V} \perp\!\!\!\perp D$$

Since, $D \perp\!\!\!\perp V$, this implies that $\text{Law}(q_2(d,V)) = \text{Law}(q_2(d',V))$ for all $d, d' \in \mathcal{D}$.

Since $\tilde{e}$ is a bijection when restricted to inputs that are outputs of f and since f is an injection, we have that $\tilde{e} \circ f$ is an injection. Thus there exists a well-defined inverse function $q^{-1} = e \circ \tilde{f}$, such that $D, V = q^{-1}(D,V)$. Let q_2^{-1} be the V component of its output. Since $D \perp\!\!\!\perp V$ and $\tilde{D} = PD$, we have that:

$$V \equiv q_2^{-1}(\tilde{D}, \tilde{V}) \perp\!\!\!\perp \tilde{D}$$

Since $\tilde{D} \perp\!\!\!\perp \tilde{V}$, this implies that $\text{Law}(q_2^{-1}(d,\tilde{V})) = \text{Law}(q_2^{-1}(d',\tilde{V}))$ for all $d, d' \in \mathcal{D}$ □

E.2. Proof of Positive Improvement

Proof of Theorem E.4. In this proof, we show that intervention in the direction of average derivatives of $\tilde{h}$ guarantees positive improvement for sufficiently small α, assuming that h is twice differentiable. If we perform the intervention $\tilde{D} + \alpha u$, then we have by Lemma E.2 that:

$$D_{\alpha u}, V_{\alpha u} = (D + \alpha P^{-1} u, q_2^{-1}(\tilde{D} + \alpha u, \tilde{V}))$$

However, note also that:

$$\text{Law}(q_2^{-1}(d,\tilde{V})) = \text{Law}(q_2^{-1}(d + \alpha u, \tilde{V}))$$

for any $d \in \mathcal{D}$. By Theorem E.1, we have that, with probability 1, $\tilde{D} = PD$ and $\tilde{U} = PU$. Moreover, by assumption, we have that $\tilde{V} \perp\!\!\!\perp \tilde{U} \perp\!\!\!\perp Z$, which implies

$$\tilde{V} \perp\!\!\!\perp \{\tilde{A}Z + \tilde{U}, P^{-1}\tilde{U}\} \implies \tilde{V} \perp\!\!\!\perp \{\tilde{D}, U\},$$

we also have that:

$$\text{Law}(q_2^{-1}(d, \tilde{V}) \mid \tilde{D} = d, U) = \text{Law}(q_2^{-1}(d + \alpha u, \tilde{V}) \mid \tilde{D} = d, U)$$
$$\implies \text{Law}(q_2^{-1}(\tilde{D}, \tilde{V}) \mid \tilde{D}, U) = \text{Law}(q_2^{-1}(\tilde{D} + \alpha u, \tilde{V}) \mid \tilde{D}, U)$$

which by the definition of V and V_α is equivalent to:

$$\text{Law}(V \mid \tilde{D}, U) = \text{Law}(V_{\alpha u} \mid \tilde{D}, U) \implies \text{Law}(V \mid U) = \text{Law}(V_{\alpha u} \mid U)$$

By the outcome structural equation

$$Y = h(D) + \eta(U, V, \epsilon)$$

we have that:

$$Y_{\alpha u} = h(D + \alpha P^{-1} u) + g(V_{\alpha u}) + \eta(U, \epsilon_Y)$$

and that:

$$\mathbb{E}[Y_{\alpha u} - Y] = \mathbb{E}[h(D + \alpha P^{-1} u) - h(D)] + \mathbb{E}[\eta(U, V_{\alpha u}, \epsilon) - \eta(U, V, \epsilon)]$$

Since $\epsilon \perp\!\!\!\perp \{Z, U, V\}$ and since $V_{\alpha u}$ is a measurable function of these random variables, we have that $\epsilon \perp\!\!\!\perp \{V_{\alpha u}, V, U\}$. Letting $\tilde{\eta}(u, v) = \mathbb{E}_\epsilon[\eta(u, v, \epsilon)]$, we can write:

$$\begin{aligned}
\mathbb{E}[Y_{\alpha u} - Y] &= \mathbb{E}[h(D + \alpha P^{-1} u) - h(D)] + \mathbb{E}[\tilde{\eta}(U, V_{\alpha u}) - \tilde{\eta}(U, V)] \\
&= \mathbb{E}[h(D + \alpha P^{-1} u) - h(D)] + \mathbb{E}[\mathbb{E}[\tilde{\eta}(U, V_{\alpha u}) - \tilde{\eta}(U, V) \mid U]] \\
&= \mathbb{E}[h(D + \alpha P^{-1} u) - h(D)] && (\text{Law}(V \mid U) = \text{Law}(V_{\alpha u} \mid U))
\end{aligned}$$

By a first-order Taylor expansion and since h is twice differentiable with bounded first and second derivatives:

$$\mathbb{E}[Y_{\alpha u} - Y] = \mathbb{E}[\alpha \nabla_D h(D)^\top P^{-1} u] + O(\alpha^2) = \alpha \|(P^{-1})^\top \mathbb{E}[\nabla_D h(D)]\| + O(\alpha^2)$$

□

E.3. Auxiliary Lemmas

Lemma E.5. *Suppose A and B are $r \times k$ matrices with full row rank. If $A = AB^+B$, then rowspan(A) = rowspan(B) and AB^+ is invertible.*

Proof. Consider the short SVDs of $B = U_B \Sigma_B V_B^\top$ and $A = U_A \Sigma_A V_A^\top$. Then

$$B^+ B = V_B \Sigma_B^{-1} U_B^\top U_B \Sigma_B V_B^\top = V_B V_B^\top$$

is the projection onto the row space of B. Then, we have:

$$A = AB^+B \quad \Leftrightarrow \quad AV_B V_B^\top = A \tag{9}$$

First, we prove by contradiction that rowspan(A) = rowspan(B). Suppose $x \in$ rowspan(A) = span(V_A), but $x \notin$ rowspan(B) = span(V_B). Let $V_B^\perp$ denote an orthogonal completion of V_B, then

$$x \notin \text{span}(V_B) \Rightarrow x = V_B V_B^\top x + V_B^\perp (V_B^\perp)^\top x$$

where $V_B^\perp (V_B^\perp)^\top x \neq 0$, which implies $(V_B^\perp)^\top x \neq 0$ as $V_B^\perp$ is orthogonal. Hence, we have the following:

$$\|x\|^2 = x^\top x = x^\top V_B V_B^\top x + x^\top V_B^\perp (V_B^\perp)^\top x = \|V_B^\top x\|^2 + \|(V_B^\perp)^\top x\|^2 > \|V_B^\top x\|^2$$

However, we also have that $AV_BV_B^\top x - Ax = 0$, which implies $u \triangleq V_BV_B^\top x - x \in$ null-space$(A) =$ span$(V_A^\perp)$. Thus, it should be orthogonal with $x \in$ span(V_A).

$$0 = x^\top u = x^\top (V_BV_B^\top x - x) = \|V_B^\top x\|^2 - \|x\|^2 \neq 0$$

This yields a contradiction! Thus, rowspan$(A) \subseteq$ rowspan(B). Since, both matrices have full row rank, then A and B have the same row space.

Now we show that AB^+ is invertible:

$$AB^+ = U_A\Sigma_A V_A^\top V_B \Sigma_B^{-1} U_B^\top$$

Since A, B are full row rank, $U_A, U_B, \Sigma_A, \Sigma_B$ are $r \times r$ invertible matrices. So it suffices to show that $V_A^\top V_B$ is invertible. Since span$(V_A) =$ span(V_B), there exists an invertible change-of-basis matrix P such that

$$V_B = V_AP \Rightarrow V_A^\top V_B = V_A^\top V_A P = I_r P = P \Rightarrow AB^+ \text{ is invertible.}$$

□

Lemma E.6. *If $U \perp\!\!\!\perp V \perp\!\!\!\perp Z$ (jointly independent), then $V \perp\!\!\!\perp Z \mid f(Z, U)$, for any measurable function U.*

Let $W = f(Z, U)$. Then:

$$\begin{aligned}
p(v \mid w, z) &= \int_u p(v \mid w, z, u) p(u \mid w, z) du \\
&= \int_u p(v \mid z, u) p(u \mid w, z) du \\
&= \int_u p(v) p(u \mid w, z) du && (V \perp\!\!\!\perp Z \perp\!\!\!\perp U) \\
&= p(v) \int_u p(u \mid w, z) du \\
&= p(v)
\end{aligned}$$

Lemma E.7 (Sufficient Conditions for Bounded Completeness)**.** *Consider $D = A \cdot Z + U, \quad U \perp\!\!\!\perp Z$. D is bounded complete for Z if the following holds:*

- *The measure of AZ is continuous and is supported on $\mathbb{R}^r$.*
- *The density of U is continuous.*
- *The characteristic function of the distribution of U is infinitely often differentiable and does not vanish on the real line.*

Proof of Lemma E.7. This result follows as a Corollary of Theorem 2.1 in D'Haultfoeuille, where we consider the special case of linear mappings from Z to D. □

F. Further Details on Experimental Evaluation

F.1. Linear

We benchmark LIRR against PCA under the setting of linear data generating process. As a baseline, we consider using PCA to extract the top $r = k = 4$ principal components of X as the learned latent representation. After the representation is generated, we run 2SLS with representation $\tilde{D}$ as "treatment", outcome Y, and instruments Z to identify the direction of perturbation. We apply steps 4-6 in Algorithm 1 with $\alpha = 1$ to compute the improvement $\mathbb{E}[Y_{\alpha u} - Y]$.

We tested our method across three distinct noise cases: 1) independent Gaussian distributions for both U and V, 2) correlated Uniform distribution for U, independent Gaussian distribution for V, 3) correlated Uniform distribution for U, correlated

Table 2. Average Test Improvement Comparison on Linear Data: LIRR vs. PCA (Mean ± Std). DGP 1 corresponds to independent U and V, DGP 2 corresponds to correlated U and independent V, and lastly DGP 3 corresponds to correlated U and V.

Size m	Method	DGP 1	DGP 2	DGP 3
50	LIRR	$\mathbf{3.7283 \pm 2.7360}$	$\mathbf{5.4706 \pm 4.1242}$	$\mathbf{5.4944 \pm 4.0596}$
	PCA	3.1035 ± 3.6229	3.1717 ± 4.0468	2.5171 ± 4.8519
100	LIRR	$\mathbf{2.4189 \pm 2.0164}$	$\mathbf{4.0806 \pm 3.5969}$	$\mathbf{3.8931 \pm 3.3116}$
	PCA	2.1249 ± 2.7203	2.4044 ± 3.5741	2.5713 ± 3.7491
500	LIRR	$\mathbf{1.0355 \pm 0.9698}$	$\mathbf{1.6996 \pm 1.5957}$	$\mathbf{1.5934 \pm 1.7305}$
	PCA	1.0098 ± 1.0786	0.9005 ± 1.3995	1.1716 ± 1.6904

Gaussian distribution for V. Each experiment was repeated 100 times with different random seeds, each containing a sample size of 10000 with 80-20 train-test split. We also varied the dimensionality of X, m, to examine the dimension effects.

To determine the true outcome after perturbation, We used the formula

$$Y_{\alpha u} = \theta^T (B^\dagger X_{\alpha u}).$$

The results are included in Table 2 and Figure 2.

We note that when noise follows independent Gaussian distributions across coordinates of U and V, PCA method performs comparably to LIRR. However, PCA fails to generalize effectively under non-independent noise conditions. The average improvement of our proposed method exceeds that of SVD in case 2 and 3, and being more than 1 standard deviation from zero. We can also observe that the test improvements of LIRR are shifted to the right compared to the baseline PCA method in Figure 2.

More details on data generating process is included below.

Linear DGP 1 Independent Gaussian U and V

Draw DGP parameters

$$A \sim \{N(0, 0.1^2)\}^{r \times k} \qquad B \sim \{N(0,1)\}^{m \times r} \qquad \theta \sim \{N(0,1)\}^{r \times 1}$$

Then generate n samples as:

$$Z_i \sim \mathcal{N}(0, I_k) \qquad \text{(instrument)}$$

$$U_i \sim \mathcal{N}(0, 20^2 \cdot I_r) \qquad \text{(confounder 1)}$$

$$V_i \sim \mathcal{N}(0, 10^2 \cdot I_m) \qquad \text{(confounder 2)}$$

$$\eta_i(U_i, V_i) = \sum_{j=1}^{r} U_{ij} + 0.2 \cdot \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1) \qquad \text{(confounder 3)}$$

$$D_i = AZ_i + U_i \qquad \text{(latent representation)}$$

$$X_i = BD_i + V_i \qquad \text{(observed representation)}$$

$$Y_i = \theta^\top D + \eta_i(U_i, V_i)$$

With dimensions $n = 10000$, $r = k = 4$, where $i \in \{1, 2, \ldots, n\}$ indexes the samples.

Linear DGP 2 Correlated Uniform U and Independent Gaussian V

Draw DGP parameters

$$A \sim \{N(0, 0.1^2)\}^{r \times k} \qquad B \sim \{N(0,1)\}^{m \times r} \qquad \theta \sim \{N(0,1)\}^{r \times 1}$$
$$E \sim \{N(0,1)\}^{h \times r}$$

Then generate n samples as:

$$Z_i \sim \mathcal{N}(0, I_k) \qquad \text{(instrument)}$$
$$U_i \sim E \cdot \{\text{Unif}(-1,-1)\}^h \qquad \text{(correlated Uniform confounder 1)}$$
$$V_i \sim \mathcal{N}(0, 10^2 \cdot I_m) \qquad \text{(confounder 2)}$$
$$\eta_i(U_i, V_i) = \sum_{j=1}^{r} U_{ij} + 0.2 \cdot \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1) \qquad \text{(confounder 3)}$$
$$D_i = AZ_i + U_i \qquad \text{(latent representation)}$$
$$X_i = BD_i + V_i \qquad \text{(observed representation)}$$
$$Y_i = \theta^\top D_i + \eta_i(U_i, V_i)$$

With dimensions $n = 10000$, $r = k = 4, h = 3$, where $i \in \{1, 2, \ldots, n\}$ indexes the samples.

Linear DGP 3 Correlated Uniform U and Correlated Gaussian V

Draw DGP parameters

$$A \sim \{N(0, 0.1^2)\}^{r \times k} \qquad B \sim \{N(0,1)\}^{m \times r} \qquad \theta \sim \{N(0,1)\}^{r \times 1}$$
$$E \sim \{N(0,1)\}^{h_1 \times r} \qquad F \sim \{N(0,1)\}^{h_2 \times r}$$

Then generate n samples as:

$$Z_i \sim \mathcal{N}(0, I_k) \qquad \text{(instrument)}$$
$$U_i \sim E \cdot \{\text{Unif}(-1,-1)\}^h \qquad \text{(correlated Uniform confounder 1)}$$
$$V_i \sim F \cdot \mathcal{N}(0, 5^2 \cdot I_{h_2}) \qquad \text{(correlated Gaussian confounder 2)}$$
$$\eta_i(U_i, V_i) = \sum_{j=1}^{r} U_{ij} + 0.2 \cdot \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1) \qquad \text{(confounder 3)}$$
$$D_i = AZ_i + U_i \qquad \text{(latent representation)}$$
$$X_i = BD_i + V_i \qquad \text{(observed representation)}$$
$$Y_i = \theta^\top D_i + \eta_i(U_i, V_i)$$

With dimensions $n = 10000$, $r = k = 4, h_1 = 3, h_2 = 5$, where $i \in \{1, 2, \ldots, n\}$ indexes the samples.

Table 3. Average Test Improvement Comparison of 9 Methods on Quadratic Data (Mean $\pm$ Std). DGP 1 corresponds to independent U and V, DGP 2 corresponds to correlated U and independent V, and lastly DGP 3 corresponds to correlated U and V.

Method	Case 1	Case 2	Case 3
PCA	0.1322 ± 0.3216	0.0545 ± 0.2994	0.0848 ± 0.2382
LIRR	3.5086 ± 2.0455	3.4711 ± 1.9683	3.5682 ± 2.1296
Vanilla AE	0.4138 ± 2.2000	0.8418 ± 1.1560	0.7801 ± 1.7335
IRAE[0]	6.1055 ± 7.1634	2.2898 ± 6.9957	4.8993 ± 6.3310
IRAE[1]	$\mathbf{6.4174 \pm 5.2602}$	4.6175 ± 5.0479	$\mathbf{5.8023 \pm 7.1041}$
IRAE[2]	5.5471 ± 4.6573	4.5554 ± 4.0707	5.2145 ± 4.4358
IRAE	5.7740 ± 4.7664	$\mathbf{6.5253 \pm 6.0132}$	4.9113 ± 4.0009
Vanilla VAE	0.3651 ± 0.4629	0.2725 ± 0.5071	0.2055 ± 0.3394
iVAE	0.2709 ± 0.3672	0.1192 ± 0.2503	0.1652 ± 0.2929

To determine the true outcome after perturbation, We used the formula

$$Y_{\alpha u} = \theta^T (B^\dagger X_{\alpha u}).$$

In addition to the summary statistics, we also plotted the distribution of average test improvements across seeds in Figure 2. We can observe that the test improvements of LIRR are shifted more to the right compared to the baseline PCA method.

F.2. Quadratic

We also consider a non-linear data generating process, where the data is generated by Equation (1) where f is quadratic and h is linear. The U, V follow the same 3 cases as the linear setting.

We benchmark LIRR and IRAE against PCA and vanilla Autoencoder (vanilla AE), variational autoencoder (VAE), and iVAE. Here, vanilla AE refers to autoencoder with only reconstruction loss. VAE refers variational autoencoder that maximizes the likelihood $p_f(X)$ with Gaussian latent representation. iVAE (Khemakhem et al., 2020) utilizes both Z and X in encoding and decoding, maximizing the conditional likelihood of $p_{f,A}(X|Z)$ as information of Z is available in simulations. For LIRR, PCA, IRAE[1], vanilla AE, VAE, iVAE the bottleneck is of the same dimension as the instrument, i.e. $k = 4$, so that downstream 2SLS will not be ill-posed, whereas the bottleneck size of IRAE[2] and IRAE was 10. Algorithm 1 is then applied to evaluate the average improvement in outcome, when each of the aforementioned representation learning methods is used. For probabilistic autoencoder VAE and iVAE, we sampled 10 representations for each observation X and compared them to the original outcome. In particular, to determine the true outcome after perturbation, we used the formula

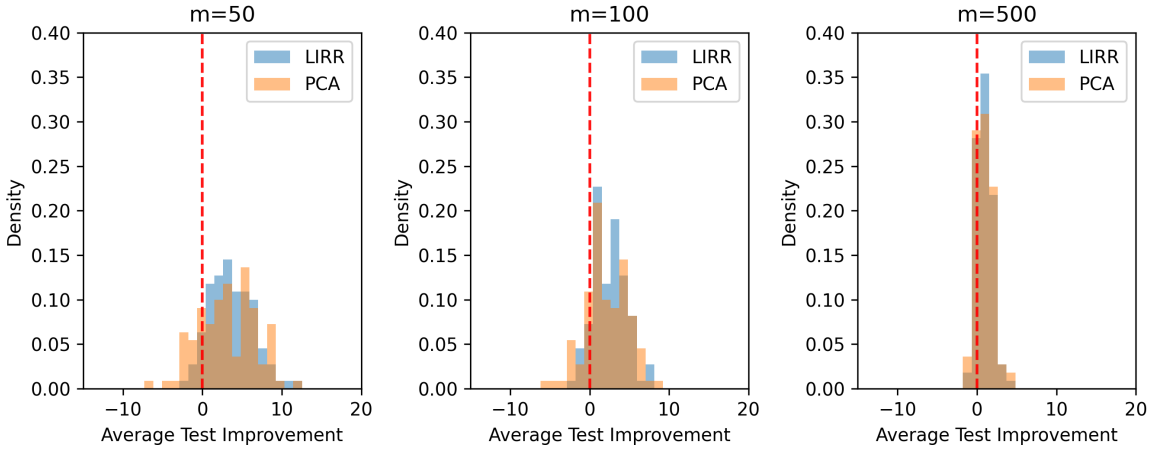
$$Y_{\alpha u} = \theta^T ((B^\dagger X_{\alpha u})[: r]),$$

where $[: r]$ index into the first order terms (excluding the quadratic and cross terms) of D.

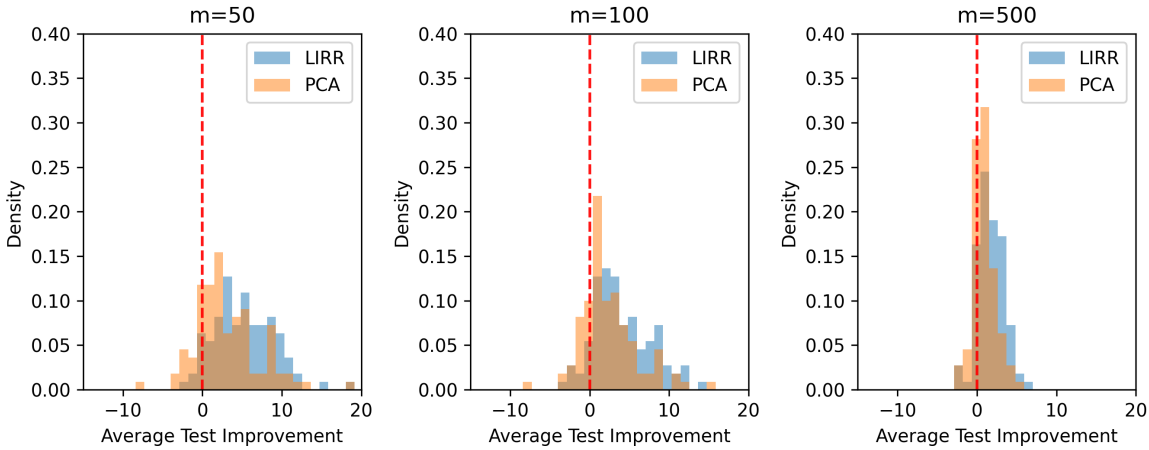
All encoder architectures incorporate a Random Fourier Feature layer, followed by three feedforward layers and a final linear projection. Decoders consist of three feedforward layers and a final linear projection layer. The hyperparameters used in the training procedure are described in Table 4.

We repeated the experiment 30 times across different random seeds, each containing a sample size of 10000 with 70-10-20 train-val-test split. Results on average improvement are depicted in Table 3. Our findings reveal that dimension reduction methods which operate without Z information (PCA, vanilla AE, vanilla VAE) yield minimal outcome improvement. In contrast, methods that incorporate Z consistently demonstrate positive mean improvements. The most substantial improvement is achieved by our IRAE[1] and IRAE method, with IRAE having performance gains at more than one standard deviation above zero.

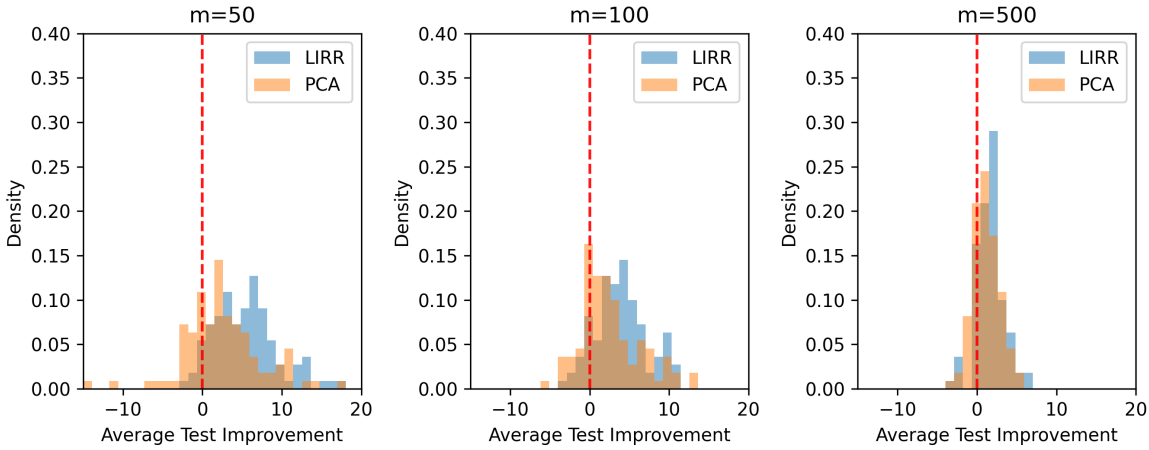
The data are generated using the following 3 cases.



(a) Linear DGP 1: Independent U and V



(b) Linear DGP 2: Correlated U and Independent V



(c) Linear DGP 3: Correlated U and V

Figure 2. Distribution of Average Improvement for Linear Experiment

Quadratic DGP 1 Independent Gaussian U, V

Draw DGP parameters

$$A \sim \{N(0,1)\}^{r\times k} \qquad B \sim \{N(0,1)\}^{m\times(2*r+r*(r-1)/2)} \qquad \theta \sim \{N(0,1)\}^{r\times 1}$$

Then generate samples as:

$$\begin{aligned}
Z_i &\sim \mathcal{N}(0, I_k) && \text{(instrument)}\\
U_i &\sim \mathcal{N}(0, 0.2^2 \cdot I_r) && \text{(confounder 1)}\\
V_i &\sim \mathcal{N}(0, 0.2^2 \cdot I_m) && \text{(confounder 2)}\\
\eta_i(U_i, V_i) &= \sum_{j=1}^{r} U_{ij} + 0.2 \cdot \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1) && \text{(confounder 3)}\\
D_i &= AZ_i + U_i && \text{(latent representation)}\\
X_i &= B \cdot [D_{i1}, D_{i2}, ..., D_{i1}D_{i2}, ...D_{ir}^2] + V_i && \text{(observed representation)}\\
Y_i &= \theta^\top D + \eta_i(U_i, V_i)
\end{aligned}$$

With dimensions $n = 10000$, $r = k = 4$, where $i \in \{1, 2, \ldots, n\}$ indexes the samples.

Quadratic DGP 2 Correlated Uniform U and Independent Gaussian V

Draw DGP parameters

$$A \sim \{N(0,1)\}^{r\times k} \qquad B \sim \{N(0,1)\}^{m\times(2*r+r*(r-1)/2)} \qquad \theta \sim \{N(0,1)\}^{r\times 1}$$
$$E \sim \{N(0,1)\}^{h\times r}$$

Then generate samples as:

$$\begin{aligned}
Z_i &\sim \mathcal{N}(0, I_k) && \text{(instrument)}\\
U_i &\sim E \cdot \{\text{Unif}(-0.2, -0.2)\}^h && \text{(correlated Uniform confounder 1)}\\
V_i &\sim \mathcal{N}(0, 0.2^2 \cdot I_m) && \text{(confounder 2)}\\
\eta_i(U_i, V_i) &= \sum_{j=1}^{r} U_{ij} + 0.2 \cdot \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1) && \text{(confounder 3)}\\
D_i &= AZ_i + U_i && \text{(latent representation)}\\
X_i &= B \cdot [D_{i1}, D_{i2}, ..., D_{i1}D_{i2}, ...D_{ir}^2] + V_i && \text{(observed representation)}\\
Y_i &= \theta^\top D_i + \eta_i(U_i, V_i)
\end{aligned}$$

With dimensions $n = 10000$, $r = k = 4, h = 3$, where $i \in \{1, 2, \ldots, n\}$ indexes the samples.

Quadratic DGP 3 Correlated Uniform U and Correlated Gaussian V

Draw DGP parameters

$$A \sim \{N(0,1)\}^{r\times k} \qquad B \sim \{N(0,1)\}^{m\times(2*r+r*(r-1)/2)} \qquad \theta \sim \{N(0,1)\}^{r\times 1}$$
$$E \sim \{N(0,1)\}^{h_1\times r} \qquad F \sim \{N(0,1)\}^{h_2\times r}$$

Then generate samples as:

$$Z_i \sim \mathcal{N}(0, I_k) \qquad \text{(instrument)}$$
$$U_i \sim E \cdot \{\text{Unif}(-0.2,-0.2)\}^h \qquad \text{(correlated Uniform confounder 1)}$$
$$V_i \sim F \cdot \mathcal{N}(0, 0.05^2 \cdot I_{h_2}) \qquad \text{(correlated Gaussian confounder 2)}$$
$$\eta_i(U_i, V_i) = \sum_{j=1}^{r} U_{ij} + 0.2 \cdot \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0,1) \qquad \text{(confounder 3)}$$
$$D_i = AZ_i + U_i \qquad \text{(latent representation)}$$
$$X_i = B \cdot [D_{i1}, D_{i2}, ..., D_{i1}D_{i2}, ...D_{ir}^2] + V_i \qquad \text{(observed representation)}$$
$$Y_i = \theta^\top D_i + \eta_i(U_i, V_i)$$

With dimensions $n = 10000$, $r = k = 4, h_1 = 3, h_2 = 5$, where $i \in \{1, 2, \ldots, n\}$ indexes the samples.

All encoder architectures incorporate a Random Fourier Feature layer, followed by three feedforward layers and a final linear projection. Decoders consist of three feedforward layers and a final linear projection layer. For our IRAE[2] and IRAE models, we set the bottleneck dimension to 10, larger than the instrumental variable dimension $r = k = 4$. By construction, Vanilla and IRAE[1] has bottleneck equal to $k = 4$. To determine the true outcome after perturbation, we used the formula

$$Y_{\alpha u} = \theta^T((B^\dagger X_{\alpha u})[:r]),$$

where $[:r]$ index into the first order terms (excluding the quadratic and cross terms) of D.

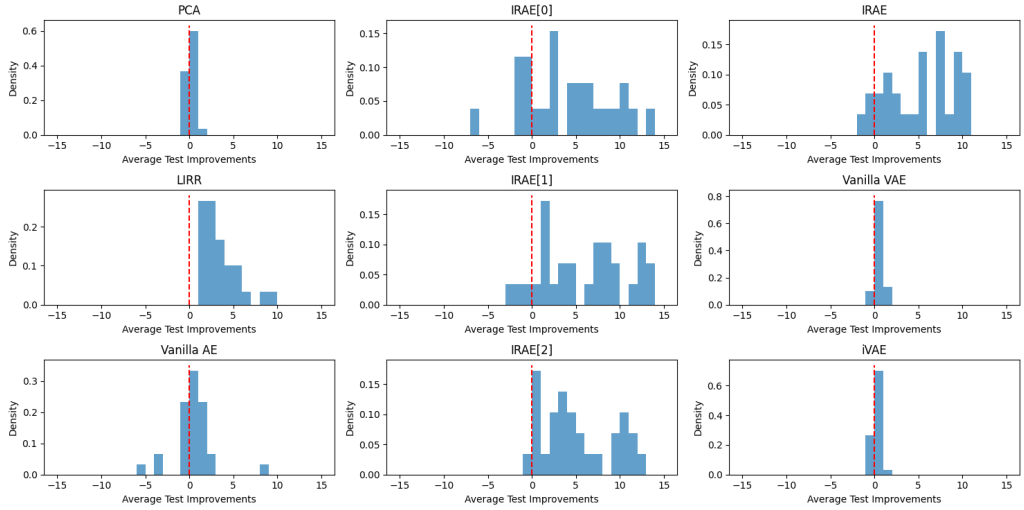
The hyperparameters used in the training procedure are described in Table 4.

Additional plots corresponding to Table 4 are included in Figure 3.

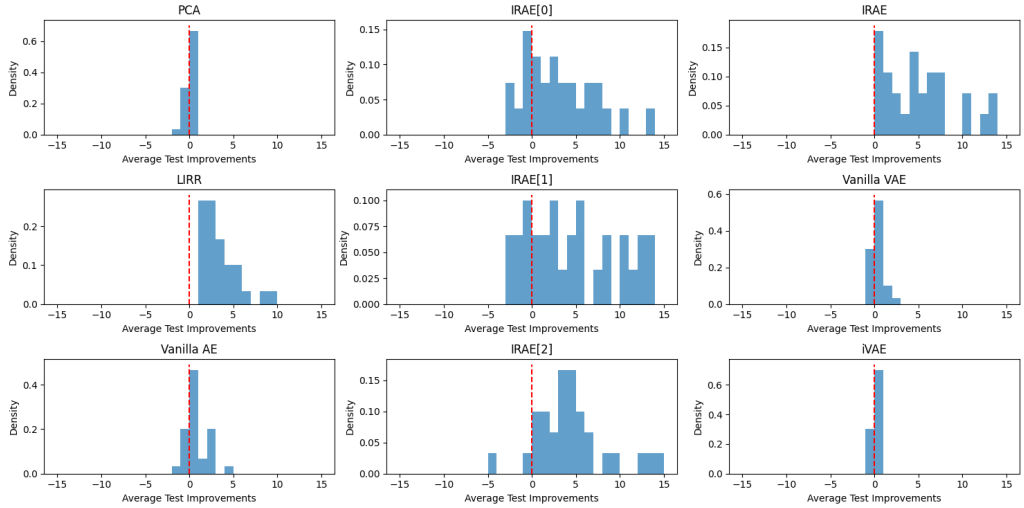
F.3. MNIST Experiment 1

This section provides details of the MNIST experiments briefly described in Section 5 of the main paper. Here we included detailed data generating equations, model hyperparameter, and plots for IRAE[0], IRAE[1], IRAE[2] that were not included in the main paper.

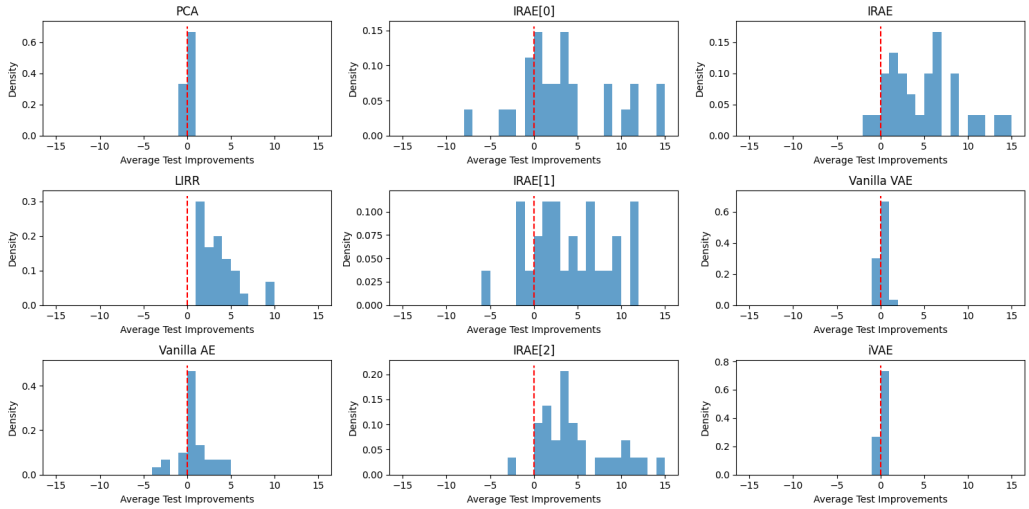
The data for MNIST experiment is generated using *Case 1 DGP*.



(a) Quadratic DGP 1: Independent U and V



(b) Quadratic DGP 2: Correlated U and Independent V



(c) Quadratic DGP 3: Correlated U and V

Figure 3. Distribution of Average Improvement for Quadratic Experiment

Table 4. Training Parameters for Quadratic Simulations

	Vanilla AE	IRAE[0]	IRAE[1]	IRAE[2]	IRAE	VAE	iVAE
Architecture							
Encoder dimensions	$100 \to 50 \to 20$						
Decoder dimensions	$20 \to 50 \to 100$						
RFF bandwidth σ	20						
Bottleneck dimension	4	4	4	10	10	4	4
Optimization							
Optimizer	RMSprop						
Learning rate	5×10^{-4}					1×10^{-4}	5×10^{-4}
Alpha	0.9						
Epsilon	1×10^{-8}						
Weight decay	1×10^{-6}						
Momentum	None						
Regularization Parameters							
λ	0	1	1	1	1	NA	NA
μ_1	0	0	1	1	1	NA	NA
μ_2	0	0	0	1	1	NA	NA
μ_3	0	0	0	0	1	NA	NA
weight for kl term	NA	NA	NA	NA	NA	3	3
Training Protocol (with early stopping of patience 20)							
	1000 epcohs						

Case 1 DGP

Draw DGP parameters $\alpha, \beta \sim \text{Unif}(0.1, 0.7)$. Then generate samples as:

$$G_i \in [0,1]^{28\times 28} \qquad \text{(grayscale MNIST image)}$$

$$Z_i,\ U_i \sim \mathcal{N}\big(0, I_2\big), \qquad Z_i \perp\!\!\!\perp U_i \qquad \text{(instrument \& confounder)}$$

$$r_i = \text{clip}\big(0.5 + \alpha\, Z_{i1} + \beta\, U_{i1},\ 0,\ 1\big) \qquad \text{(red channel)}$$

$$g_i = \text{clip}\big(0.5 + \alpha\, Z_{i2} + \beta\, U_{i2},\ 0,\ 1\big) \qquad \text{(green channel)}$$

$$b_i = \text{clip}\big(0.5 + \alpha\, \frac{Z_{i1} + Z_{i2}}{2},\ 0,\ 1\big) \qquad \text{(blue channel)}$$

$$X_i(k,\ell,c) = G_i(k,\ell)\ \cdot\ (r_i, g_i, b_i)_c, \qquad \begin{array}{l} c \in \{R, G, B\}, \\ (k,\ell) \in \{1,\ldots,28\}^2 \end{array} \qquad \text{(colour image)}$$

$$Y_i = r_i + g_i + b_i. \qquad \text{(outcome, details below)}$$

Returns the tuples $\big(Z_i, X_i, Y_i\big)$.

All encoders consist of three Conv2D layers, followed by additional feedforward layers, and conclude with a linear projection. Decoders mirror this architecture in reverse order. For our IRAE[2] and IRAE models, we set the bottleneck dimension to 10 which is larger than $k = 2$. For vanilla and IRAE[0], IRAE[1], the bottle neck is 2. The autoencoder with multiple HSIC regularization terms presents greater training challenges due to the complexity of term. To address this, we initialized IRAE[2] and IRAE with weights from the simpler IRAE[1] model. All of models are trained with 60k training samples and evaluated on 10k test set. More training details can be found in Table 5.

Table 6. Average Test Improvement Comparison of 5 Methods on MNIST Data (Mean $\pm$ Std)

sample size	image	Vanilla AE	IRAE[0]	IRAE[1]	IRAE[2]	IRAE
1000	reconstructed	$\mathbf{-0.57 \pm 0.03}$	-0.64 ± 0.17	-0.6 ± 0.15	-0.63 ± 0.09	-0.6 ± 0.12
	intervened(0.2)	-0.56 ± 0.04	-0.44 ± 0.14	-0.5 ± 0.14	-0.52 ± 0.16	$\mathbf{-0.43 \pm 0.24}$
	intervened(1.0)	-0.51 ± 0.05	-0.39 ± 0.15	-0.42 ± 0.15	-0.34 ± 0.2	$\mathbf{-0.22 \pm 0.36}$
10000	reconstructed	-0.51 ± 0.03	-0.73 ± 0.04	-0.73 ± 0.04	-0.33 ± 0.05	$\mathbf{-0.32 \pm 0.04}$
	intervened(0.2)	-0.5 ± 0.03	-0.07 ± 0.15	0.07 ± 0.26	0.72 ± 0.48	$\mathbf{0.76 \pm 0.36}$
	intervened(1.0)	-0.47 ± 0.04	0.04 ± 0.2	0.15 ± 0.29	0.92 ± 0.47	$\mathbf{0.95 \pm 0.43}$
30000	reconstructed	-0.51 ± 0.03	-0.74 ± 0.04	-0.73 ± 0.04	$\mathbf{-0.33 \pm 0.05}$	-0.34 ± 0.04
	intervened(0.2)	-0.49 ± 0.03	-0.11 ± 0.08	-0.17 ± 0.11	$\mathbf{0.38 \pm 0.4}$	0.33 ± 0.4
	intervened(1.0)	-0.44 ± 0.05	-0.06 ± 0.12	-0.13 ± 0.2	$\mathbf{0.78 \pm 0.43}$	0.74 ± 0.47
60000	reconstructed	-0.47 ± 0.02	-0.71 ± 0.05	-0.71 ± 0.04	$\mathbf{-0.25 \pm 0.05}$	$\mathbf{-0.25 \pm 0.06}$
	intervened(0.2)	-0.46 ± 0.03	-0.06 ± 0.26	0.14 ± 0.3	0.98 ± 0.42	$\mathbf{0.99 \pm 0.48}$
	intervened(1.0)	-0.41 ± 0.06	0.05 ± 0.29	0.26 ± 0.36	$\mathbf{1.13 \pm 0.44}$	1.12 ± 0.43

Table 5. Training Parameters for MNIST Simulations

Parameter	**Vanilla AE**	**IRAE[0]**	**IRAE[1]**	**IRAE[2]**	**IRAE**
Architecture					
Kernel Size			3		
Encoder channels			$16 \to 32 \to 64$		
Decoder channels			$64 \to 32 \to 16$		
Bottleneck dimension	2	2	2	10	10
Optimization					
Optimizer			Adam (default parameters in torch)		
Learning rate			1×10^{-3}		
Weight initialization	None	None	None	From IRAE[1]	From IRAE[1]
Loss Weights					
λ	0	10	10	10	10
μ_1	0	0	10	10	10
μ_2	0	0	0	10	10
μ_3	0	0	0	0	10
Training Epochs (with early stopping of patience 5)					
	50	50	50	50*	50*

* Additional epochs after initializing with weights from IRAE[1]

Remark F.1 (Calculation of Outcome from Image). To calculate expected $Y_{\alpha u}$, we first perform 2-mean clustering on the image pixels and extract the red, green, blue values from the center of the colored cluster. Then, we take the sum of these values as Y. Note that is this similar to taking the average colors over the gray scale mask so the colors would be slightly smaller than the original colors. We tested the methods on the original image and the result is 0.2 smaller on average.

Remark F.2 (Calculation of Outcome Improvement). When calculating the outcome improvement of the intervention, take the difference between the kmeans calculation described in the previous paragraph applied to the image produced by the intervention and we subtract the outcome of the kmeans calculation when applied to the original image.

Remark F.3. We use a linear kernel for HSIC in order to perform benchmarking at a large scale in fast speed, which may not capture all nonlinear dependencies in this complex image representation setting. More complex independence statistics based on domain knowledge, could perhaps lead to more disentanglement, albeit they might also be harder to train. In subsequent section experiments we also examine a pairwise RBF Kernel based HSIC and we find that it does not lead to improved performance as compared to the linear kernel.

Remark F.4. We observe that this example does not perfectly align with the formulation in Equation (1). Here, the number of instruments is 2, which is fewer than the natural representation of D of 3 colors. We may be able to interpret the learned representation as a 2-dimensional subspace of the 3-dimensional color representation, but the mapping from Z to D is still not immediately invertible as assumed in the theory. Additionally, while our theoretical analysis assumes a mapping from color D to outcome directly, our calculation employs k-means clustering on X instead. Nevertheless, this example

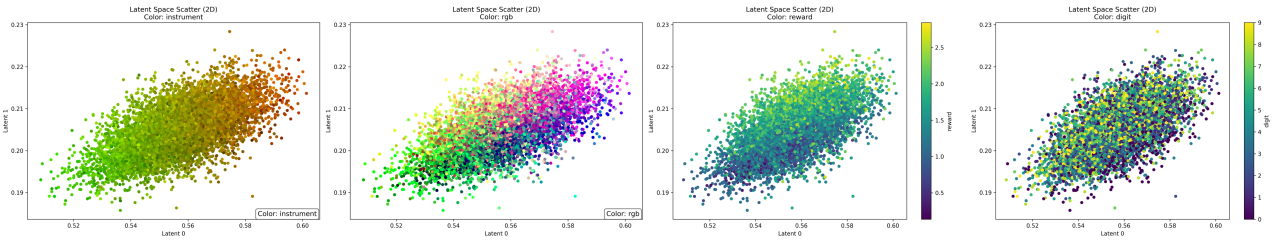


Figure 4. Original gray, original color, reconstructed, treated($\alpha = 0.2$) and treated($\alpha = 1.0$) for the IRAE[2] trained model (Case 1 DGP).

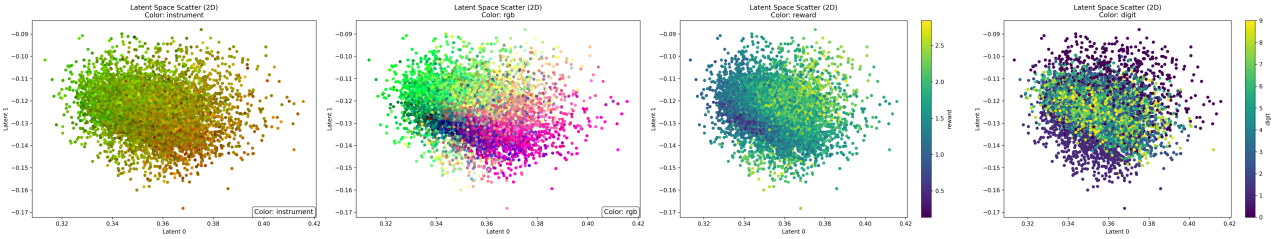


Figure 5. Original gray, original color, reconstructed, treated($\alpha = 0.2$) and treated($\alpha = 1.0$) for the IRAE[1] trained model (Case 1 DGP).

demonstrates that our method performs robustly even in settings beyond those covered by our theoretical guarantees, and offers potential future directions of theoretical investigation.

F.4. MNIST Experiment 2

Building on the results from our MNIST experiments in Section 5, we conducted a more comprehensive evaluation by exploring additional hyperparameter configurations and data generating processes. Given that independence test statistics are often complex and challenging to train, we systematically investigated various model architectures, independence test statistics calculation, and initialization strategies to identify optimal configurations. To align with the full row-rank assumption on A, we evaluated our approach on a supplementary dataset with three instruments, denoted as *Case 2 DGP*.

Our findings reveal that simpler dense architectures perform at least as well as, and often better than, more complex convolutional neural networks for this task. Furthermore, we observed that larger bottleneck dimensions in IRAE[2] and IRAE models better preserve the original digit morphology in treated images — a potentially valuable property when morphological features is confounded the outcome variable.

The full set of hyperparameters explored are included in Table 7. All of models are trained with 60k training samples and evaluated on 10k test set, for 40 random seeds. Regularization weights are 0 or 1. All models are trained with 50 epochs after initialization with early stopping of patience 5.

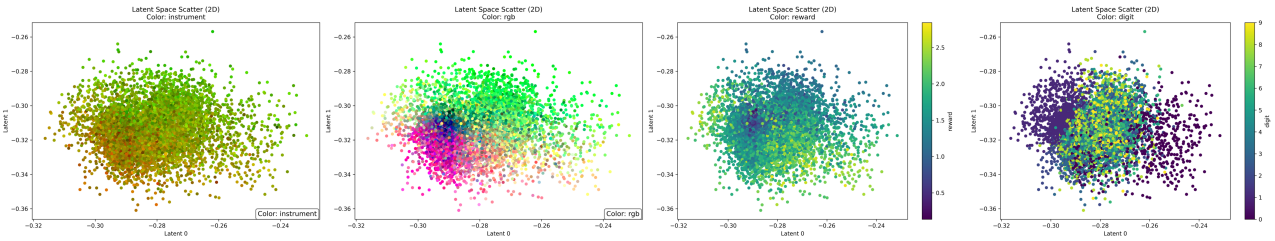


Figure 6. Original gray, original color, reconstructed, treated($\alpha = 0.2$) and treated($\alpha = 1.0$) for the IRAE[0] trained model (Case 1 DGP).

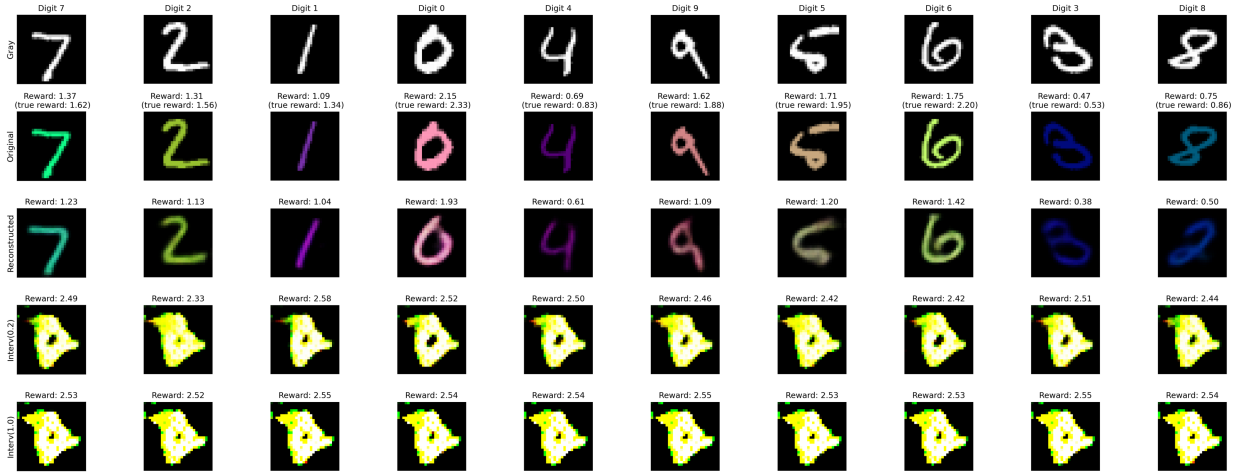


Figure 7. **IRAE[2]** on Case 1 DGP for one random seed (random seed 22), with a Conv AutoEncoder, linear HSIC as independence criterion, **latent dimension 10**, regularization weights $\lambda = \mu_1 = \mu_2 = 10$ and training for 50 epochs with early stopping (patience 5 epochs) warm start from IRAE1.

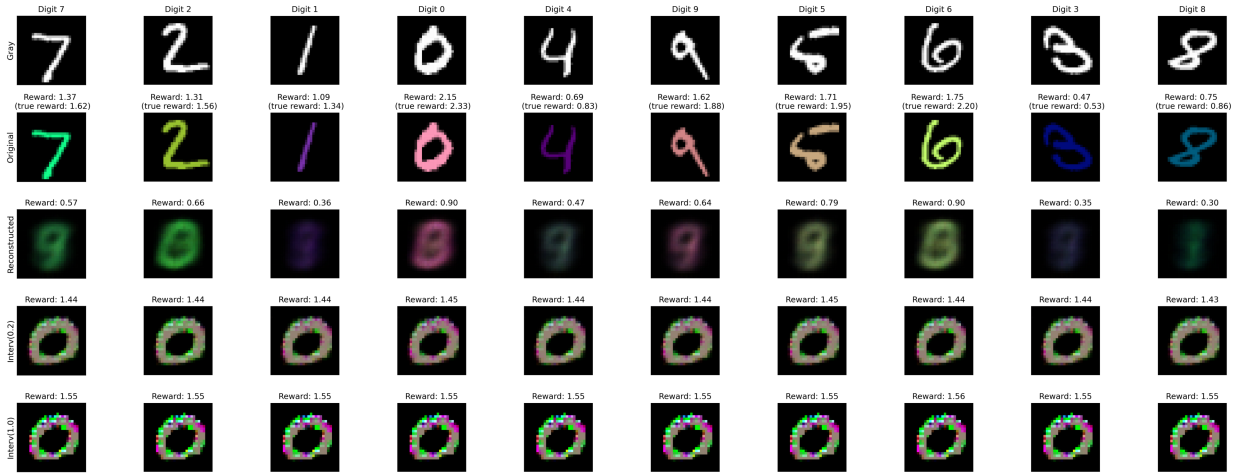


Figure 8. **IRAE[1]** on Case 1 DGP for one random seed (random seed 22), with a Conv AutoEncoder, linear HSIC as independence criterion, **latent dimension 2**, regularization weights $\lambda = \mu_1 = 10$ and training for 50 epochs with early stopping (patience 5 epochs) from scratch

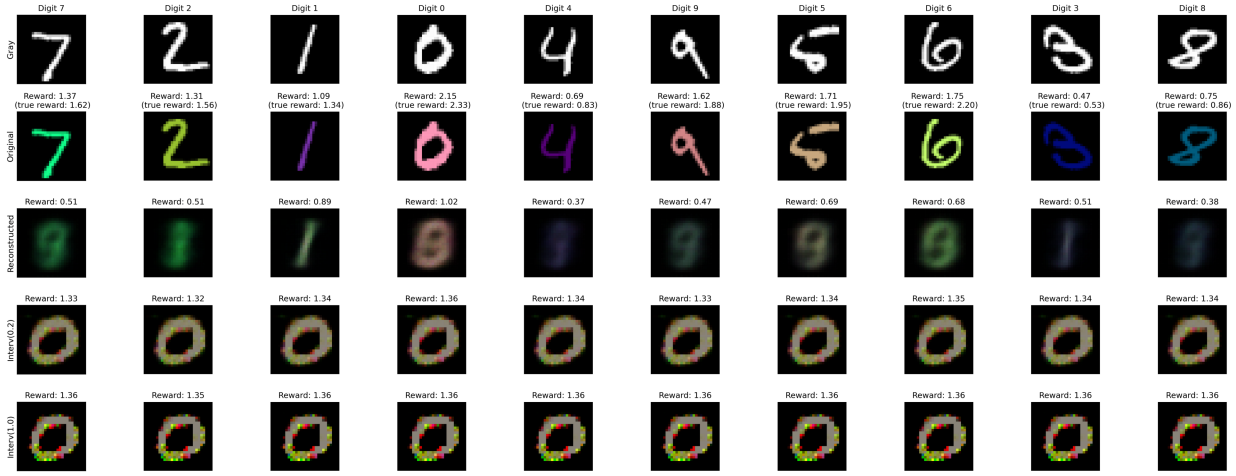


Figure 9. **IRAE[0]** on Case 1 DGP for one random seed (random seed 22), with a Conv AutoEncoder, linear HSIC as independence criterion, **latent dimension 2**, regularization weights $\lambda = \mu_1 = 10$ and training for 50 epochs with early stopping (patience 5 epochs) from scratch

Table 7. Summary of parameters explored in MNIST Experiment 2

Setting Category	**Options**	**Description**
Data Generating Process	DGP2	Three Instruments
Autoencoder Architecture	Dense	**Encoder:** Dense layer $3 \times 28 \times 28 \to 512$, followed by linear projection to latent dimension **Decoder:** Linear layer from latent dimension to 512, followed by dense layer $512 \to 3 \times 28 \times 28$
	Convolution	**Encoder:** Three Conv2D layers with channel $16 \to 32 \to 64$ of kernel size 3, followed by a dense layer of size 256 and linear projection to latent dimension **Decoder:** Linear layer from latent dimension to size 256, followed by dense layer and three Conv2D layers with channel $64 \to 32 \to 16$ of kernel size 3
Latent Dimension IRAE[2] and IRAE	10 32	Used for IRAE[2] and IRAE models Used for IRAE[2] and IRAE models
Regularization Type	Linear HSIC Pairwise HSIC	Applied as independence measure on the entire vector Applied between pairwise coordinates
Weight Initialization IRAE[2] and IRAE	Without warmstart With warmstart	Training from randomly initialized weights for 50 epochs Initializing with weights transferred from a pre-trained IRAE[1] model, and training for additional 50 epochs

Case 2 DGP

Draw DGP parameters $\alpha, \beta \sim \text{Unif}(0.1, 0.7)$. Then generate samples as:

$$G_i \in [0,1]^{28\times 28} \qquad \text{(grayscale MNIST image)}$$

$$Z_i,\ U_i \sim \mathcal{N}\big(0, I_3\big), \qquad Z_i \perp\!\!\!\perp U_i \qquad \text{(instrument \& confounder)}$$

$$r_i = \text{clip}\big(0.5 + \alpha\, Z_{i1} + \beta\, U_{i1},\ 0,\ 1\big) \qquad \text{(red channel)}$$

$$g_i = \text{clip}\big(0.5 + \alpha\, Z_{i2} + \beta\, U_{i2},\ 0,\ 1\big) \qquad \text{(green channel)}$$

$$b_i = \text{clip}\big(0.5 + \alpha\, Z_{i3} + \beta\, U_{i3},\ 0,\ 1\big) \qquad \text{(blue channel)}$$

$$X_i(k,\ell,c) = G_i(k,\ell)\ \cdot\ (r_i, g_i, b_i)_c, \qquad \begin{array}{l} c \in \{R, G, B\}, \\ (k,\ell) \in \{1,\ldots,28\}^2 \end{array} \qquad \text{(colour image)}$$

$$Y_i = r_i + g_i + b_i. \qquad \text{(outcome)}$$

Returns the tuples $\big(Z_i, X_i, Y_i\big)$.

We highlight some findings from our exploration of the performance of our proposed methods across various hyperparameter dimensions:

Architecture: We found that simple dense layers can achieve better performance than convolutional architectures for this task, suggesting that Conv2D layers may be unnecessarily complex for this particular example.

Data Generating Process: Our experimental results demonstrate that the relative performance of our methods remains consistent across both DGP1 and DGP2.

Latent Dimension: When using larger latent dimensions (32), both the reconstructed and treated images preserved more of the original digit morphology although the improvement is smaller (c.f. Figures 11 to 16). This may be a desired property in some cases, especially in the case that the digit morphology is a confounder (not tested in our experiment) and has a direct effect on the outcome.

Regularization Type: While pairwise HSIC may theoretically capture more nonlinear dependencies, we found that it was often more difficult to train in practice. Linear HSIC consistently yielded better performance with greater training stability.

Weight Initialization: Dense architectures performed well without warm start initialization, while convolutional architectures benefited significantly from weight transfer. This difference likely stems from the higher complexity and larger parameter space of convolutional networks.

Overall, the best improvement model stems from the IRAE method with all regularizers, a Dense architecture, latent = 10, linear HSIC with no warm start.

Arch.	Latent Dim	Reg Type	Warm Start	image	Vanilla AE	IRAE[0]	IRAE[1]	IRAE[2]	IRAE
dense	10	linear	False	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.67 (0.02)	**-0.27 (0.01)**	**-0.27 (0.01)**
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.1)**	1.39 (0.15)	1.35 (0.16)
				intervened(1.0)	-0.37 (0.02)	1.54 (0.11)	1.54 (0.09)	1.57 (0.12)	**1.58 (0.08)**
			True	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.67 (0.02)	**-0.36 (0.14)**	-0.43 (0.2)
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.1)**	1.17 (0.53)	0.92 (0.64)
				intervened(1.0)	-0.37 (0.02)	**1.54 (0.11)**	**1.54 (0.09)**	1.32 (0.5)	1.09 (0.58)
		pairwise	False	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.68 (0.01)	**-0.3 (0.03)**	-0.34 (0.02)
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.14)**	-0.09 (0.37)	0.17 (0.59)
				intervened(1.0)	-0.37 (0.02)	**1.54 (0.11)**	1.53 (0.13)	0.09 (0.57)	0.46 (0.69)
			True	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.68 (0.01)	**-0.33 (0.1)**	-0.63 (0.25)
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.14)**	1.31 (0.24)	0.6 (0.92)
				intervened(1.0)	-0.37 (0.02)	**1.54 (0.11)**	1.53 (0.13)	1.49 (0.15)	0.86 (0.79)
	32	linear	False	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.67 (0.02)	-0.14 (0.02)	**-0.13 (0.01)**
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.1)**	0.74 (0.34)	0.63 (0.35)
				intervened(1.0)	-0.37 (0.02)	**1.54 (0.11)**	**1.54 (0.09)**	1.43 (0.31)	1.34 (0.35)
			True	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.67 (0.02)	**-0.26 (0.12)**	-0.33 (0.25)
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.1)**	1.08 (0.36)	0.8 (0.7)
				intervened(1.0)	-0.37 (0.02)	**1.54 (0.11)**	**1.54 (0.09)**	1.29 (0.42)	1.05 (0.68)
		pairwise	False	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.68 (0.01)	**-0.13 (0.01)**	-0.19 (0.02)
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.14)**	-0.15 (0.05)	-0.21 (0.1)
				intervened(1.0)	-0.37 (0.02)	**1.54 (0.11)**	1.53 (0.13)	-0.2 (0.18)	-0.22 (0.28)
			True	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.68 (0.01)	**-0.19 (0.05)**	-0.34 (0.2)
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.14)**	0.07 (0.41)	0.13 (0.53)
				intervened(1.0)	-0.37 (0.02)	**1.54 (0.11)**	1.53 (0.13)	0.42 (0.65)	0.5 (0.67)
conv	10	linear	False	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.21 (0.03)**	-0.23 (0.03)
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.4 (0.4)	**0.98 (0.23)**	0.8 (0.39)
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.69 (0.58)	**1.25 (0.55)**	1.12 (0.65)
			True	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.2 (0.04)**	**-0.2 (0.04)**
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.4 (0.4)	**1.0 (0.45)**	0.9 (0.57)
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.69 (0.58)	**0.89 (0.75)**	0.73 (0.77)
		pairwise	False	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.22 (0.05)**	-0.26 (0.06)
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.04 (0.45)	**0.47 (0.42)**	0.45 (0.45)
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.12 (0.57)	**0.86 (0.47)**	0.8 (0.63)
			True	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.26 (0.07)**	-0.27 (0.1)
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.04 (0.45)	**0.82 (0.47)**	0.7 (0.48)
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.12 (0.57)	**1.08 (0.55)**	0.9 (0.56)
	32	linear	False	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.1 (0.03)**	-0.11 (0.03)
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.4 (0.4)	**0.7 (0.33)**	0.62 (0.38)
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.69 (0.58)	**1.26 (0.39)**	1.04 (0.52)
			True	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.1 (0.03)**	**-0.1 (0.02)**
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.4 (0.4)	1.05 (0.49)	**1.15 (0.51)**
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.69 (0.58)	1.11 (0.57)	**1.22 (0.6)**
		pairwise	False	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.11 (0.03)**	-0.13 (0.05)
				intervened(0.2)	-0.36 (0.03)	**0.21 (0.34)**	0.04 (0.45)	0.02 (0.26)	0.13 (0.28)
				intervened(1.0)	-0.31 (0.07)	**0.4 (0.56)**	0.12 (0.57)	0.21 (0.49)	0.35 (0.54)
			True	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.14 (0.08)**	-0.18 (0.08)
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.04 (0.45)	**0.4 (0.56)**	0.35 (0.66)
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.12 (0.57)	0.68 (0.68)	**0.7 (0.73)**

Figure 10. Experimental results for the **Case 2** data generating process. Mean improvement and standard deviation of improvement is reported. *reconstructed* refers to the mean outcome improvement of the reconstructed image from the autoencoder with no intervention in the latents, as compared to the original image. *intervened(α)* refers to the mean outcome improvement of the image produced by intervening on the latents in direction $\alpha \cdot u$, where $u = \theta / \|\theta\|$ and θ is estimated by 2SLS in latent space.

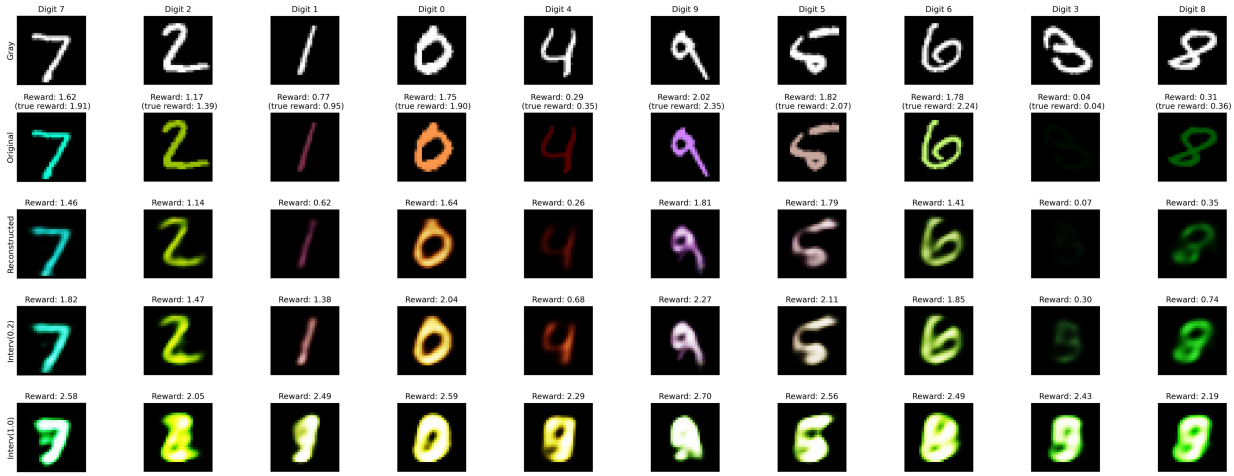


Figure 11. **IRAE** on Case 2 DGP for one random seed (random seed 22), with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 32**, regularization weights $\lambda = \mu_1 = \mu_2 = \mu_3 = 1$ and training for 50 epochs with early stopping (patience 5 epochs) from scratch (no warm start from IRAE1).

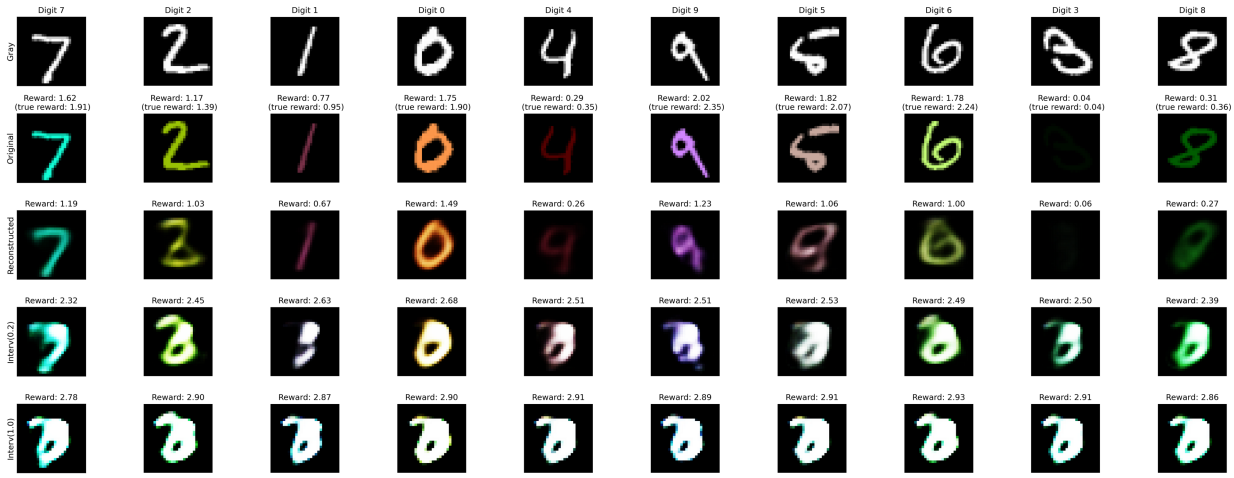


Figure 12. **IRAE** on Case 2 DGP for one random seed (random seed 22), with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 10**, regularization weights $\lambda = \mu_1 = \mu_2 = \mu_3 = 1$ and training for 50 epochs with early stopping (patience 5 epochs) from scratch (no warm start from IRAE1).

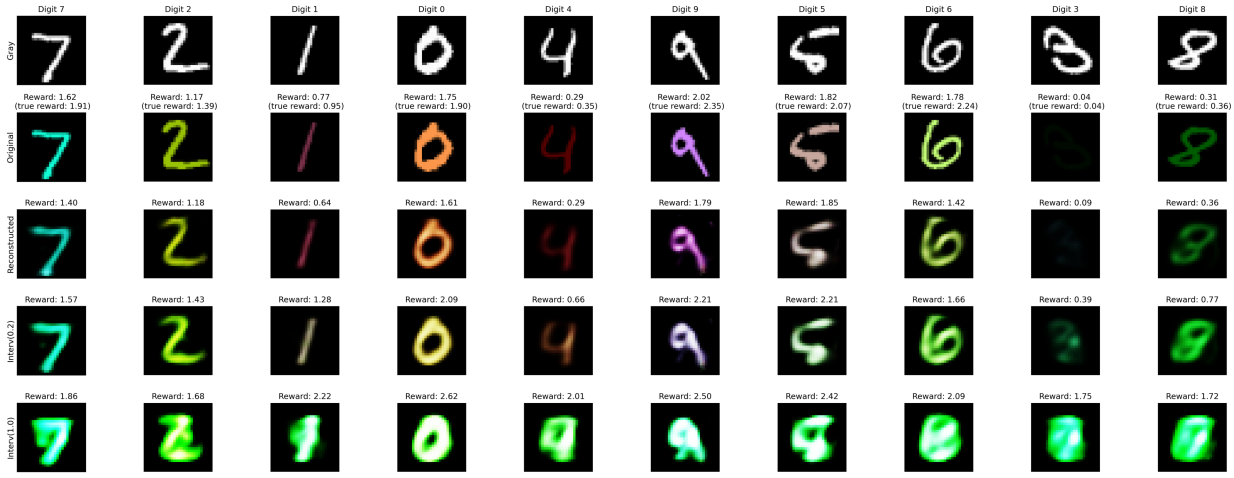


Figure 13. **IRAE[2]** on Case 2 DGP for one random seed (random seed 22), with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 32**, regularization weights $\lambda = \mu_1 = \mu_2 = 1$ and training for 50 epochs with early stopping (patience 5 epochs) from scratch (no warm start from IRAE[1]).

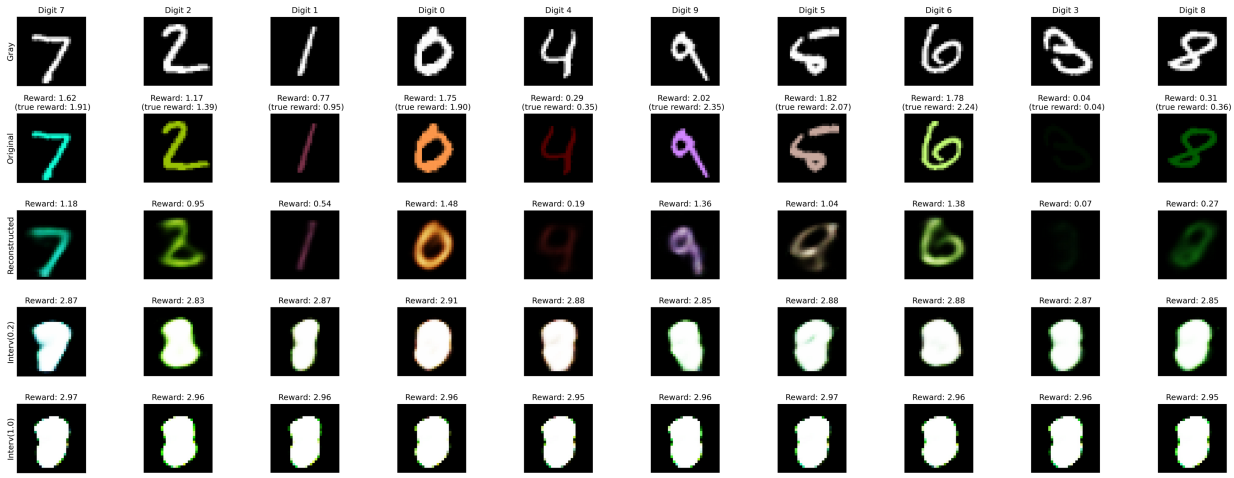


Figure 14. **IRAE[2]** on Case 2 DGP for one random seed (random seed 22), with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 10**, regularization weights $\lambda = \mu_1 = \mu_2 = 1$ and training for 50 epochs with early stopping (patience 5 epochs) from scratch (no warm start from IRAE[1]).

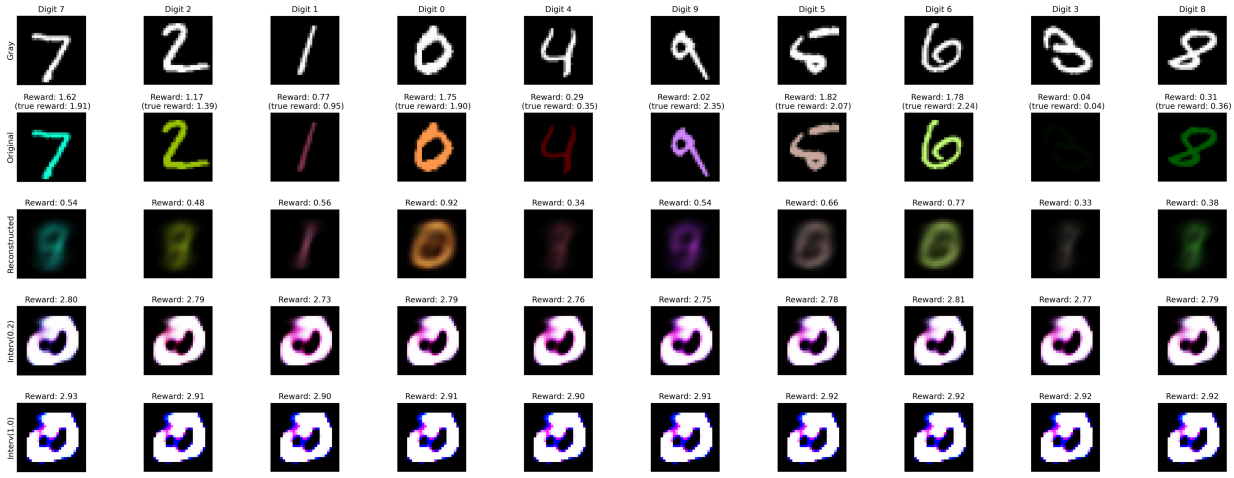


Figure 15. **IRAE[1]** on Case 2 DGP for one random seed (random seed 22), with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 3 = number of instruments**, regularization weights $\lambda = \mu_1 = 1$ and $\mu_2 = \mu_3 = 0$ and training for 50 epochs with early stopping (patience 5 epochs).

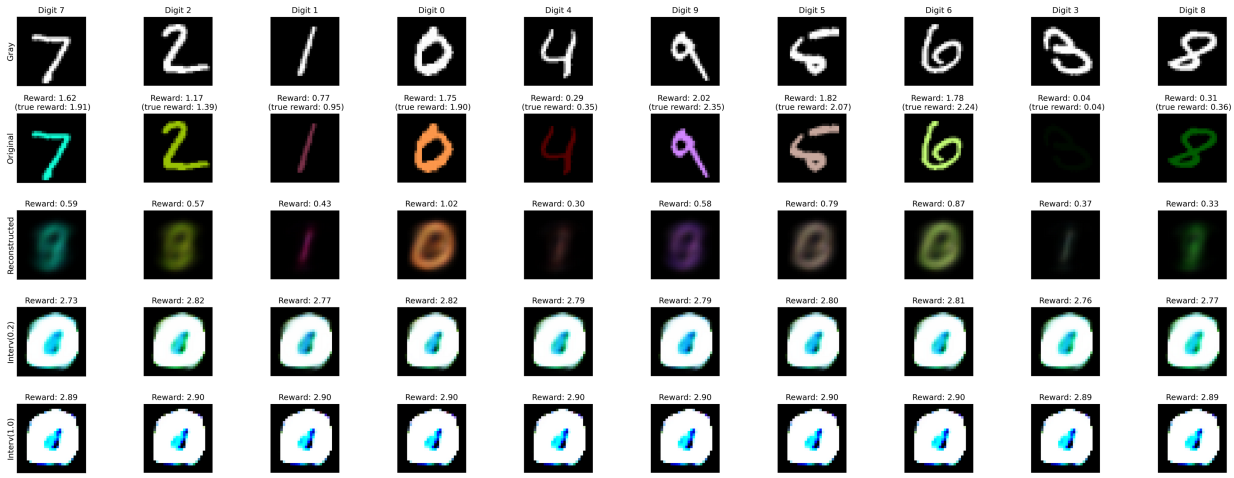


Figure 16. **IRAE[0]** on Case 2 DGP for one random seed (random seed 22), with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 3 = number of instruments**, regularization weights $\lambda = \mu_1 = 1$ and $\mu_2 = \mu_3 = 0$ and training for 50 epochs with early stopping (patience 5 epochs).

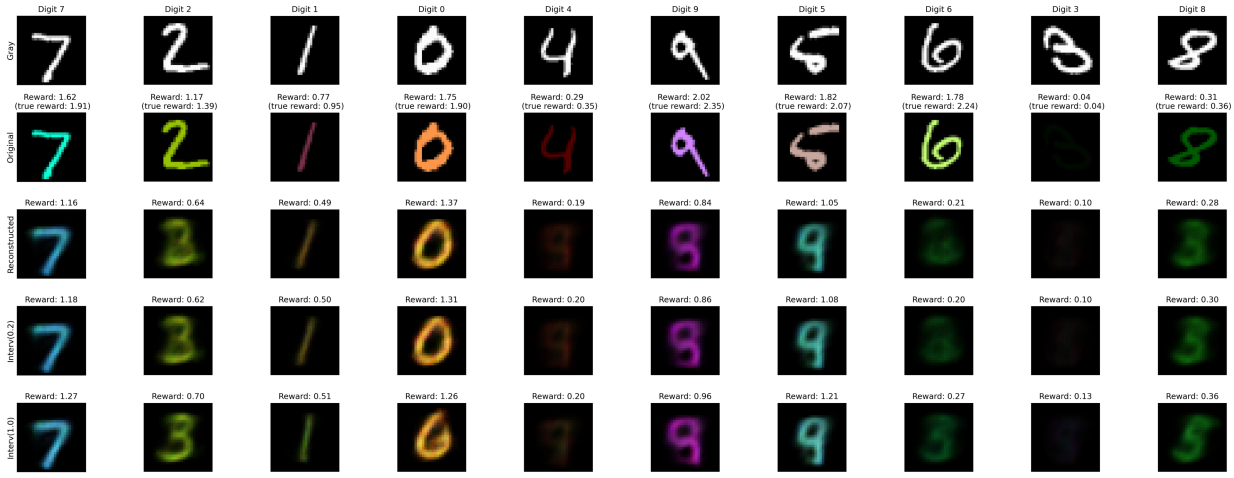


Figure 17. **Vanilla AE** on Case 2 DGP for one random seed (random seed 22), with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 3 = number of instruments**, regularization weights $\lambda = \mu_1 = \mu_2 = \mu_3 = 0$ and training for 50 epochs with early stopping (patience 5 epochs).

F.5. Case 3: Confounded Outcome

We examine the following confounded outcome generating process, where the instruments now affect the colors in a more convoluted intertwined manner. We denote this as *Case 3 DGP*.

All of models are trained with 60k training samples and evaluated on 10k test set, for 40 random seeds. Regularization weights are 0 or 1. All models are trained with 50 epochs after initialization with early stopping of patience 5.

Case 3 DGP

Draw DGP parameters $\alpha, \beta \sim \text{Unif}(0.1, 0.7)$. Then generate samples as:

$$G_i \in [0,1]^{28\times 28} \qquad \text{(grayscale MNIST image)}$$

$$Z_i,\ U_i \sim \mathcal{N}\big(0, I_3\big), \qquad Z_i \perp\!\!\!\perp U_i \qquad \text{(instrument \& confounder)}$$

$$r_i = \text{clip}\big(0.5 + \alpha\, Z_{i1} + \beta\, U_{i1},\ 0,\ 1\big) \qquad \text{(red channel)}$$

$$g_i = \text{clip}\big(0.5 + \alpha\, Z_{i2} + \beta\, U_{i2},\ 0,\ 1\big) \qquad \text{(green channel)}$$

$$b_i = \text{clip}\big(0.5 + \alpha\, Z_{i3} + \beta\, U_{i3},\ 0,\ 1\big) \qquad \text{(blue channel)}$$

$$X_i(k,\ell,c) = G_i(k,\ell)\ \cdot\ (r_i, g_i, b_i)_c, \qquad \begin{array}{l} c \in \{R,G,B\}, \\ (k,\ell) \in \{1,\ldots,28\}^2 \end{array} \qquad \text{(colour image)}$$

$$Y_i = r_i + g_i + b_i - U_{i1} - U_{i2} - U_{i3}. \qquad \text{(confounded outcome)}$$

Returns the tuples $\big(Z_i, X_i, Y_i\big)$.

In this confounding setting, we found that IRAE[0], IRAE[1], IRAE[2], IRAE still led to improved outcome, whereas Vanilla AE did not.

Arch	Latent Dim	Reg Type	Warm Start	image	Vanilla AE	IRAE[0]	IRAE[1]	IRAE[2]	IRAE
dense	10	linear	False	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.67 (0.02)	**-0.27 (0.01)**	**-0.27 (0.01)**
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.1)**	1.38 (0.15)	1.35 (0.16)
				intervened(1.0)	-0.37 (0.03)	1.54 (0.11)	1.54 (0.09)	1.57 (0.12)	**1.58 (0.08)**
	32	linear	False	reconstructed	-0.46 (0.02)	-0.67 (0.02)	-0.67 (0.02)	-0.14 (0.02)	**-0.13 (0.01)**
				intervened(0.2)	-0.45 (0.02)	**1.4 (0.12)**	**1.4 (0.1)**	0.74 (0.34)	0.63 (0.35)
				intervened(1.0)	-0.37 (0.03)	**1.54 (0.11)**	**1.54 (0.09)**	1.42 (0.32)	1.34 (0.35)
conv	10	linear	False	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.21 (0.03)**	-0.23 (0.03)
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.4 (0.4)	**0.98 (0.23)**	0.8 (0.39)
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.69 (0.58)	**1.25 (0.54)**	1.12 (0.65)
	32	linear	False	reconstructed	-0.37 (0.02)	-0.6 (0.06)	-0.6 (0.05)	**-0.1 (0.03)**	-0.11 (0.03)
				intervened(0.2)	-0.36 (0.03)	0.21 (0.34)	0.4 (0.4)	**0.7 (0.33)**	0.62 (0.38)
				intervened(1.0)	-0.31 (0.07)	0.4 (0.56)	0.69 (0.58)	**1.26 (0.39)**	1.04 (0.52)

Figure 18. Experimental results for the **Case 3** data generating process. Mean improvement and standard deviation of improvement is reported.

F.6. Case 4: Confounded DGP with One Outcome Relevant Dimension

We examine the following confounded outcome generating process, where the instruments now affect the colors in a more convoluted intertwined manner. Moreover, only the red channel is relevant for the outcome and the outcome is confounded. We denote this as *Case 4 DGP*.

All of models are trained with 60k training samples and evaluated on 10k test set, for 40 random seeds. Regularization weights are 0 or 1. All models are trained with 50 epochs after initialization with early stopping of patience 5.

Arch	Latent Dim	Reg Type	Warm Start	image	Vanilla AE	IRAE[0]	IRAE[1]	IRAE[2]	IRAE
dense	10	linear	False	reconstructed	-0.16 (0.01)	-0.22 (0.01)	-0.22 (0.01)	**-0.09 (0.01)**	**-0.09 (0.01)**
				intervened(0.2)	-0.15 (0.01)	**0.51 (0.03)**	0.5 (0.03)	**0.51 (0.02)**	**0.51 (0.02)**
				intervened(1.0)	-0.1 (0.02)	**0.55 (0.01)**	**0.55 (0.02)**	**0.55 (0.01)**	**0.55 (0.01)**
	32	linear	False	reconstructed	-0.16 (0.01)	-0.22 (0.01)	-0.22 (0.01)	**-0.05 (0.01)**	**-0.05 (0.01)**
				intervened(0.2)	-0.15 (0.01)	**0.51 (0.03)**	0.5 (0.03)	0.5 (0.01)	0.49 (0.04)
				intervened(1.0)	-0.1 (0.02)	**0.55 (0.01)**	**0.55 (0.02)**	0.54 (0.01)	0.54 (0.01)
conv	10	linear	False	reconstructed	-0.13 (0.01)	-0.2 (0.02)	-0.2 (0.02)	**-0.07 (0.03)**	**-0.07 (0.02)**
				intervened(0.2)	-0.13 (0.01)	0.26 (0.12)	0.28 (0.14)	**0.45 (0.03)**	0.43 (0.13)
				intervened(1.0)	-0.11 (0.04)	0.42 (0.19)	0.44 (0.21)	**0.54 (0.01)**	0.51 (0.14)
	32	linear	False	reconstructed	-0.13 (0.01)	-0.2 (0.02)	-0.2 (0.02)	-0.04 (0.03)	**-0.03 (0.02)**
				intervened(0.2)	-0.13 (0.01)	0.26 (0.12)	0.28 (0.14)	**0.45 (0.04)**	**0.45 (0.05)**
				intervened(1.0)	-0.11 (0.04)	0.42 (0.19)	0.44 (0.21)	**0.53 (0.01)**	0.52 (0.05)

Figure 19. Experimental results for the **Case 4** data generating process. Mean improvement and standard deviation of improvement is reported.

Case 4 DGP

Draw DGP parameters $\alpha, \beta \sim \text{Unif}(0.1, 0.7)$. Then generate samples as:

$$G_i \in [0,1]^{28\times 28} \qquad \text{(grayscale MNIST image)}$$

$$Z_i,\ U_i \sim \mathcal{N}\big(0, I_3\big), \qquad Z_i \perp\!\!\!\perp U_i \qquad \text{(instrument \& confounder)}$$

$$r_i = \text{clip}\big(0.5 + \alpha\,(Z_{i1} - Z_{i2}) + \beta\, U_{i1},\ 0,\ 1\big) \qquad \text{(red channel)}$$

$$g_i = \text{clip}\big(0.5 + \alpha\,(Z_{i2} - Z_{i3}) + \beta\, U_{i2},\ 0,\ 1\big) \qquad \text{(green channel)}$$

$$b_i = \text{clip}\big(0.5 + \alpha\,(Z_{i3} - Z_{i1}) + \beta\, U_{i3},\ 0,\ 1\big) \qquad \text{(blue channel)}$$

$$X_i(k,\ell,c) = G_i(k,\ell)\ \cdot\ (r_i, g_i, b_i)_c, \qquad \begin{array}{l} c \in \{R,G,B\}, \\ (k,\ell) \in \{1,\ldots,28\}^2 \end{array} \qquad \text{(colour image)}$$

$$Y_i = r_i - U_{i1}. \qquad \text{(confounded outcome)}$$

Returns the tuples $\big(Z_i, X_i, Y_i\big)$.

We demonstrate in this data generating process the importance of running an instrumental variable regression in the latent space. We see below that if instead we had run OLS regressing the outcome on the identified latent factors, then the direction would be erroneous and the interventional images will not be moving the image towards more red colors.

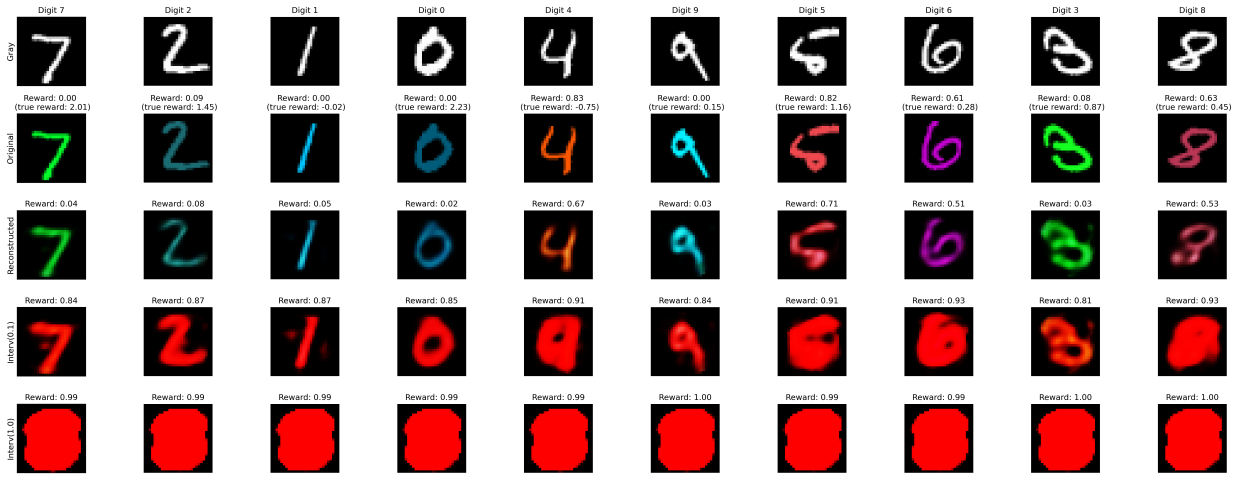


Figure 20. **IRAE** on **Case 4 DGP** for one random seed, with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 32**, regularization weights $\lambda = \mu_1 = \mu_2 = \mu_3 = 1$ and training for 50 epochs with early stopping (patience 5 epochs) from scratch (no warm start from IRAE1). Interventional images are intervened in the **direction identified by 2SLS** in the latent space with instrument Z, treatment D and outcome Y. The outcome is larger when the color of the image is changed to red.

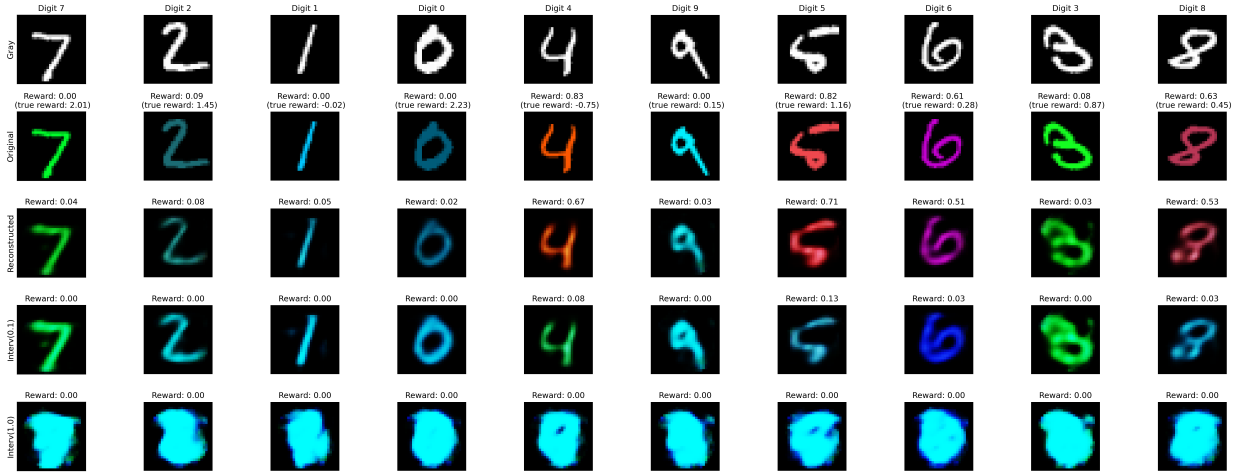


Figure 21. **IRAE** on **Case 4 DGP** for one random seed, with a Dense AutoEncoder, linear HSIC as independence criterion, **latent dimension 32**, regularization weights $\lambda = \mu_1 = \mu_2 = \mu_3 = 1$ and training for 50 epochs with early stopping (patience 5 epochs) from scratch (no warm start from IRAE[1]). Interventional images are intervened in the **direction identified by OLS**($Y \sim D$) in the latent space. The outcome is larger when the color of the image is changed to red.