Causal Balancing for Domain Generalization

Anonymous Authors[1]

Abstract

While machine learning models rapidly advance the state-of-the-art on various real-world tasks, out-of-domain (OOD) generalization remains a challenging problem given the vulnerability of these models to spurious correlations. We propose a balanced mini-batch sampling strategy to reduce the domain-specific spurious correlations in the observed training distributions. More specifically, we propose a two-phased method that 1) identifies the source of spurious correlations, and 2) builds balanced mini-batches free from spurious correlations by matching on the identified source. We provide an identifiability guarantee of the source of spuriousness and show that our proposed approach samples from a balanced, spurious-free distribution under ideal scenario. Experiments are conducted on three domain generalization datasets, demonstrating empirically that our balanced mini-batch sampling strategy improves the performance of four different established domain generalization model baselines compared to the random mini-batch sampling strategy.

1. Introduction

Machine learning is achieving tremendous success in many fields with useful real-world applications (Silver et al., 2016; Devlin et al., 2019; Jumper et al., 2021). However, machine learning models often fail to generalize to out-of-domain (OOD) data sampled from unseen environments (Quiñonero-Candela et al., 2009; Szegedy et al., 2014). One explanation for such a failure is that the models are prone to learning spurious correlations that change between environments.

Recently, various causal-inspired (Pearl, 2009) methods have been proposed to improve the OOD generalizability by considering the invariance of causal features or the underlying causal mechanism through which data is generated. Such methods often aim to find invariant data representations using new loss function designs that incorporate the invariance conditions across different domains into the training process (Arjovsky et al., 2020; Mahajan et al., 2021; Liu et al., 2021; Lu et al., 2022; Wald et al., 2021). Unfortunately, these approaches have to contend with trade-offs between weak simple models or approaches without theoretical guarantees (Arjovsky et al., 2020; Wald et al., 2021), and empirical studies have shown their utility in the real world to be questionable (Gulrajani & Lopez-Paz, 2020).

In this paper, we propose a two-step method to metigate spurious correlations: first learning the observed data distribution using a variational autoencoder (VAE) by *Latent Covariate Learning*, and then use the learned latent covariate to create a *balanced mini-batch sampling* strategy that is balanced with respect to the source of spurious correlations, to ensure the domain-invariant causal relations are learned by a classifier instead. Because the main modification at classifier training time is simply a resampling of train examples, it makes our method lightweight and highly flexible, enabling seamless incorporation with off-the-shelf domain generalization methods (Gulrajani & Lopez-Paz, 2020; Sagawa et al., 2019; Sun & Saenko, 2016).

Our contributions are as follows: (1) a general causality-based framework for the domain generalization problem of classification tasks; (2) a novel balanced mini-batch sampling algorithm that, in an ideal scenario with exact matches of the true source of spurious correlation, can remove the spurious correlations in the observed data distribution; (3) results demonstrating that the source of spurious correlation, as a latent variable, can be identified given a large enough set of training domains under mild conditions in a nonlinear setting; (4) empirical results showing that our two-phased method obtains significant performance gain on three domain generalization datasets, ColoredMNIST (Arjovsky et al., 2020), PACS (Li et al., 2017) and TerraIncognita (Beery et al., 2018), across four different domain generalization models over a random sampling strategy.

2. Problem Setting

We consider a standard domain generalization setting with a potentially high-dimensional variable X (e.g. an image), a label variable Y and a discrete environment (or domain) vari-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

(a) Observed distribution $p(X, Y|E = e)$

(b) Balanced distribution $\hat{p}(X, Y|E = e)$

Figure 1. The causal graphical model assumed for data generation process in environment $e \in \mathcal{E}$. Shaded nodes means being observed and white nodes means not being observed. Black arrows means causal relations invariant across different environments. Red dashed line means correlation varies across different environments.

able E in the sample spaces $\mathcal{X}, \mathcal{Y}, \mathcal{E}$, respectively. Here we focus on the classification problems with $\mathcal{Y} = \{1, 2, ..., m\}$ and $\mathcal{X} \in \mathbb{R}^d$. We assume that the training data are collected from a finite subset of training environments $\mathcal{E}_{\text{train}} \subset \mathcal{E}$. The training data $\mathcal{D}^e = \{(x_i^e, y_i^e)\}_{i=1}^{N^e}$ is then sampled from the distribution $P(X, Y|E = e)$ for all $e \in \mathcal{E}_{\text{train}}$. Our goal is to learn a model $h : \mathcal{X} \to \mathcal{Y}$ that can accurately predict Y given X in a new, unseen environment $e_{test} \notin \mathcal{E}_{\text{train}}$.

We assume that there is a data generation process of the observed data distribution $p(X, Y|E = e)$ represented by an underlying structural causal model (SCM) shown in Figure 1a. More specifically, we assume that X is caused by label Y, an unobserved latent variable Z and an independent noise variable ϵ with the following formulation:

$$X = \mathbf{f}(Y, Z) + \epsilon = \mathbf{f}_Y(Z) + \epsilon \tag{1}$$

We further assume that $\mathbf{f} : \{1, 2, ..., m\} \times \mathcal{Z} \to \mathcal{X}$ is injective and invariant across all environments $e \in \mathcal{E}_{\text{train}}$, where $\mathcal{Z} \in \mathbb{R}^n$ is the sample space of Z. Here, we denote by $\mathbf{f}^{-1} : \mathcal{X} \to \{1, 2, ..., m\} \times \mathcal{Z}$ the left inverse of $\mathbf{f}$. Note that this assumption forces the generation process of X to consider both Z and Y instead of only one of them. Suppose ϵ has a known probability density function p_ϵ. Then we have

$$p_{\mathbf{f}}(X|Z, Y) = p_\epsilon(X - \mathbf{f}_Y(Z)) \tag{2}$$

While the causal mechanism is invariant across environments, we assume that the correlation between label Y and latent Z is environment-variant and Z should exclude Y information. i.e., Y cannot be recovered as a function of Z. If Y is a function of Z, the generation process of X can completely ignore Y and f would not be injective. Z can be understood as a potentially massive set of latent features that do not determine the "Y-ness" and that can be spuriously correlated with environment E.

We further specify the correlation between Z and Y by assuming that the conditional distribution $p(Z|Y, E = e)$ is conditional factorial with an exponential family distribution. We denote the i-th element of Z by Z_i. Then we have the following formulation:

$$p_{T,\lambda}(Z|Y, E = e) = \prod_{i=1}^{n} \frac{Q_i(Z_i)}{W_i^e(Y)} \exp\Big[\sum_{j=1}^{k} T_{ij}(Z_i)\lambda_{ij}^e(Y)\Big] \tag{3}$$

where $\mathbf{Q} = [Q_i]_i : \mathcal{Z} \to \mathbb{R}^n$ is the base measure, $\mathbf{W}^e = [W_i^e]_i : \mathcal{Y} \to \mathbb{R}^n$ is the normalizing constant, $\mathbf{T} = [T_{ij}]_{ij} : \mathcal{Z} \to \mathbb{R}^{nk}$ are the sufficient statistics, and $\lambda^e = [\lambda_{ij}^e]_{ij} : \mathcal{Y} \to \mathbb{R}^{nk}$ are the Y dependent parameters.

In this setting, we can see that the correlation between X and Y would vary for different values of e. We argue that the correlation $Y \leftrightarrow Z \to X$ is not stable in an unseen environment $e \notin \mathcal{E}_{\text{train}}$ as it involves E and we only want to learn the stable causal relation $Y \to X$. However, it is inevitable that the learned predictor may absorb the unstable relation between X and Y if we simply train it on the observed train distribution $P(X, Y|E = e)$ with empirical risk minimization. The causal graphs of two examples of the realization of our data generation model are shown in A.

To avoid learning the unstable relations, we propose to consider a balanced distribution $\hat{p}(X, Y|E = e)$ such that $Y \perp\!\!\!\perp Z$ while the causal mechanism $Z \to X \leftarrow Y$ unchanged, as shown in Figure 1b. In this new distribution, X and Y is only correlated through the stable causal relation $Y \to X$. So when taking expectation over the balanced distribution, the model will only learn the stable causal relation between X and Y.

3. Method

3.1. Balanced mini-batch sampling

We consider a classic method that has been widely used in the average treatment effect (ATE)—balancing score matching—to sample balanced mini-batches that mimic a balanced distribution shown in Figure 1b.

Causal effect estimation studies the effect a treatment would have had on a unit which in reality received another treatment. A causal graph similar to Figure 1a is usually considered in a causal effect estimation problem, where Z is called the covariate (e.g. a patient profile), which is observed before treatment $Y \in \{0, 1\}$ (e.g. taking drug or placebo) is applied. We denote the effect of receiving a specific treatment $Y = y$ as X_y. Note that this causal graph implies that we make the Strong Ignorability assumption. i.e. Z includes all variables that are related to both X and Y.

In the case of a binary treatment, the average treatment effect (ATE) is defined as the expected difference of effect after receiving different treatments: $\mathbb{E}[X_1 - X_0]$ (e.g. difference in blood pressure). For a randomized controlled trial, we can directly estimate the difference between $\mathbb{E}[X|Y = 1]$ and $\mathbb{E}[X|Y = 0]$ from the observed data as the true treatment effect, as in this case we force $Z \perp\!\!\!\perp Y$ and there would not be systematical difference between units exposed to one treatments and units exposed to another.

However, in most observed datasets, Z is correlated with Y. Thus $\mathbb{E}[X|Y = 1]$ and $\mathbb{E}[X|Y = 0]$ are not directly

comparable. We propose to use balancing score $b(Z)$ to de-correlate Z and Y:

Definition 3.1. A balancing score $b(Z)$ is a function of covariate Z s.t. $Z \perp\!\!\!\perp Y | b(Z)$.

The ATE can then be estimated by matching units with same balancing score but different treatments: $\mathbb{E}_{b(Z)}\left[\mathbb{E}[X|Y=1, b(Z)] - \mathbb{E}[X|Y=0, b(Z)]\right]$. There is a wide range of functions of Z that can be used as a balancing score, where the scalar propensity score $p(Y=1|Z)$ is the coarsest one and the covariate Z itself is the finest one (ROSENBAUM & RUBIN, 1983). To extend this statement to non-binary treatments, we first define propensity score $s(Z)$ for $Y \in \mathcal{Y} = \{1, 2, ..., m\}$ as a vector:

Definition 3.2. The propensity score for $Y \in \{1, 2, ..., m\}$ is $s(Z) = [p(Y=y|Z)]_{y=1}^{m}$.

We have the following theorem that applies to the vector version of propensity score $s(Z)$: [1]

Theorem 3.3. *Let $b(Z)$ be a function of Z. Then b(x) is a balancing score, if and only if $b(Z)$ is finer than $s(Z)$. i.e. exists a function g such that $s(Z) = g(b(Z))$.*

We use $b^e(Z)$ to denote the balancing score for a specific environment e. We propose to construct balanced mini-batches by matching $1 \leq a \leq m-1$ examples with a different label Y but the same/closest balancing score $b^e(Z)$ for each example sampled from the training environment e. (The detailed sampling algorithm is shown in Appendix A.)

With perfect match at every step (i.e., $d(b^e(z_j), b^e(z)) = 0$ for some distance metric $d(\cdot,\cdot)$) and $a = m-1$, we can obtain a completely balanced mini-batch sampled from the balanced distribution with $Y \perp\!\!\!\perp Z$. Here, we define the balanced distribution in Figure 1b, $\hat{p}(X, Y|E=e)$:

Definition 3.4. A balanced distribution in environment e, $\hat{p}(X, Y|E=e)$, maintains the causal mechanism $Z \rightarrow X \leftarrow Y$ while $Y \perp\!\!\!\perp Z$. Here we set $\hat{p}(Y|Z, E) = \hat{p}(Y|E) = \mathrm{U}\{1, 2, ..., m\}$ and $\hat{p}(Z|E) = p(Z|E)$.

For a set of training environments $\mathcal{E}_{\text{train}}$, we further define the balanced distribution over all training environments:

Definition 3.5. A balanced distribution over all training environments $\hat{p}(X, Y)$, is obtained by uniformly sample a train environment, then sample from a balanced distribution in that environment. That is $\hat{p}(X, Y) = \frac{1}{|\mathcal{E}_{\text{train}}|}\sum_{e \in \mathcal{E}_{\text{train}}} \hat{p}(X, Y|E=e)$.

However, an exact match of balancing score is unlikely in reality, so the quality of matched data point would likely be lower than the referencing data point in terms of having the same balancing score. This can be mitigated by choosing a smaller a. However, this would make Y and Z not completely independent. In fact, if we have exact match of balancing score, the larger a is, the weaker the correlation between Y and Z would be. So in practice, the choice of a reflects a trade-off between the balancing score matching quality and the degree of dependency between Y and Z. The above arguments can be summarized as follow:

Theorem 3.6. *A balanced mini-batch with exact matches of balancing score and $a = m-1$ can be regarded as sampling from the balanced distribution over all training environments $\hat{p}(X, Y)$. In general, the balanced mini-batch can be regarded as sampling from a semi-balanced distribution with $\hat{p}(Y|Z, E) = \frac{1}{a+1}(\frac{a}{m-1} + \frac{m-a-1}{m-1}p(Y|Z, E))$.*

Here we assume the balancing score of each data point is known. However, in domain generalization, covariate Z is not observed and so there is no way to calculate the balancing score $b(Z)$. The remaining question is how to accurately infer the latent covariate Z from the observed distributions $p(X, Y|E=e)$, where $e \in \mathcal{E}_{\text{train}}$.

3.2. Latent Covariate Learning

We argue that the underlying joint distribution of $p(X, Z|Y, E=e)$ can be learned by a VAE under the previous assumptions. We consider the following conditional generative model in each environment $e \in \mathcal{E}_{\text{train}}$, with parameters $\theta = (\mathbf{f}, \mathbf{T}, \lambda)$ defined in Section 2:

$$p_\theta(X, Z|Y, E=e) = p_{\mathbf{f}}(X|Z, Y)p_{\mathbf{T},\lambda}(Z|Y, E=e) \quad (4)$$

We use a VAE to estimate the above generative model with a variational approximation $q_\phi(Z|X, Y, E)$ of the prior probability of latent variable $p_\theta(Z|X, Y, E)$. We denote the empirical data distribution given by dataset $\mathcal{D}^e = \{(x_i^e, y_i^e)\}_{i=1}^{N^e}$ collected from environment e by $q_{\mathcal{D}^e}$. The evidence lower bound (ELBO) of the data log-likelihood in each environment $e \in \mathcal{E}_{\text{train}}$ is then defined as follows:

$$\begin{aligned}
&\mathbb{E}_{q_{\mathcal{D}^e}}\left[\log p_\theta(X|Y, E=e)\right] \geq \mathcal{L}^e(\theta, \phi) := \\
&\mathbb{E}_{q_{\mathcal{D}^e}}\Big[\mathbb{E}_{q_\phi(Z|X,Y,E=e)}\left[_\theta(X|Z, Y)\right] \\
&- KL(q_\phi(Z|X, Y, E=e)||p_\theta(Z|Y, E=e)) \quad (5)
\end{aligned}$$

The KL-divergence term can be calculated analytically. To sample from the variational distribution $q_\phi(Z|X, Y, E=e)$, we use reparameterization trick (Kingma & Welling, 2013).

We then maximize the above ELBO $\frac{1}{|\mathcal{E}_{\text{train}}|}\sum_{e \in \mathcal{E}_{\text{train}}} \mathcal{L}^e(\theta, \phi)$ over all training environments to obtain model parameters (θ, ϕ). The propensity score in a training environment e would be $s^e(Z) = [p_\theta(Y=y|Z, E=e)]_{y=1}^{m}$, which can be derived from the learned conditional prior $p_\theta(Z|Y, E=e)$:

$$p_\theta(Y=y|Z, E) = \frac{p_\theta(Z|Y=y, E)p(Y=y|E)}{\sum_{i=1}^{m} p_\theta(Z|Y=i, E)p(Y=i|E)} \quad (6)$$

where $p(Y=i|E=e)$ can be directly estimated from the training data $\mathcal{D}^e$. To show that we can uniquely recover the latent variable Z up to some simple transformations,

[1] See Appendix B for all the proofs.

we want to show that the model parameter θ is identifiable up to some simple transformations. That is, for any $\{\theta = (\mathbf{f}, \mathbf{T}, \lambda), \theta' = (\mathbf{f}', \mathbf{T}', \lambda')\} \in \Theta$,

$$p_\theta(X|Y, E = e) = p_{\theta'}(X|Y, E = e), \forall e \in \mathcal{E}_{\text{train}} \implies \theta \sim \theta', \tag{7}$$

where Θ is the parameter space and $\sim$ represents an equivalent relation. Specifically, we consider the following equivalence relation from (Motiian et al., 2017):

Definition 3.7. If $(\mathbf{f}, \mathbf{T}, \lambda) \sim_A (\mathbf{f}', \mathbf{T}', \lambda')$, then there exists an invertible matrix $A \in \mathbb{R}^{nk \times nk}$ and a vector $\mathbf{c} \in \mathbb{R}^{nk}$, such that $\mathbf{T}(\mathbf{f}^{-1}(x)) = A\mathbf{T}'(\mathbf{f}'^{-1}(x)) + \mathbf{c}, \forall x \in \mathcal{X}$.

When the underlying model parameter θ^* can be recovered by perfectly fitting the data distribution $p_{\theta*}(X|Y, E = e)$ for all $e \in \mathcal{E}_{\text{train}}$, the joint distribution $p_{\theta^*}(X, Z|Y, E = e)$ is also recovered. This further implies the recovery of the prior $p_{\theta^*}(Z|Y, E = e)$ and the true latent variable Z^*. The identifiablity of our proposed latent covariate learning model is summarized as follows:

Theorem 3.8. *Suppose we observe data sampled from the generative model defined according to Equation (4), with parameters $\theta = (\mathbf{f}, \mathbf{T}, \lambda)$ as defined in Section 2. In addition, we assume the following conditions holds: (1) The set $\{x \in \mathcal{X} | \varphi_(x) = 0\}$ has measure zero, where φ is the characteristic function of the density p. (2) The sufficient statistics T_{ij} are differentiable almost everywhere, and $(T_{ij})_{1 \le j \le k}$ are linearly independent on any subset of $\mathcal{X}$ of measure greater than zero. (3) There exist $nk + 1$ distinct points $(y_0, e_0), \ldots, (y_{nk}, e_{nk})$ such that the $nk \times nk$ matrix*

$$L = (\lambda^{e_1}(y_1) - \lambda^{e_0}(y_0), \ldots, \lambda^{e_{nk}}(y_{nk}) - \lambda^{e_0}(y_0)) \tag{8}$$

is invertible. Then we have the parameters $\theta = (\mathbf{f}, \mathbf{T}, \lambda)$ are $\sim_A$-identifiable.

Note that the last assumption in Theorem 3.8 implies that the product space $\mathcal{Y} \times \mathcal{E}_{\text{train}}$ has to be large enough to ensure the identifiability of θ when perfectly fitting the given training data distribution. i.e. we need $m|\mathcal{E}_{\text{train}}| > nk$.

4. Experiments

To verify the effectiveness of our proposed mini-batch balancing method, we conduct experiments on three domain generalization datasets: **ColoredMNIST** (Arjovsky et al., 2020), **PACS** (Li et al., 2017) and **TerraIncognita** (Beery et al., 2018). We use the training-domain validation as defined in (Gulrajani & Lopez-Paz, 2020) for model selection.

As our method only modifies the mini-batch sampling strategy, we applied our proposed method along with four widely-used domain generalization baselines: **ERM** (Vapnik, 1998), **IRM** (Gulrajani & Lopez-Paz, 2020), **DRO** (Sagawa et al., 2019) and **CORAL** (Sun & Saenko, 2016), and compare the performance of using our method with using the usual random mini-batch sampling strategy.

Table 1. Out-of-domain test accuracy on ColoredMNIST, PACS and TerraIncognita dataset. Numbers are averaged over all test environments with 3 runs with standard deviation.

	Alg	ColoredMNIST	PACS	TerraIncognita
Random	ERM	51.5 ± 0.1	83.7 ± 0.1	46.1 ± 1.2
	IRM	42.5 ± 1.0	81.2 ± 0.4	39.3 ± 1.8
	DRO	51.9 ± 0.0	83.7 ± 0.4	43.6 ± 1.2
	CORAL	51.4 ± 0.1	84.8 ± 0.2	44.8 ± 0.3
Ours-Z	ERM	46.5 ± 0.9	**85.2** ± 0.3	47.1 ± 0.6
	IRM	44.6 ± 4.7	82.6 ± 0.3	40.1 ± 1.4
	DRO	49.2 ± 0.7	84.7 ± 0.6	41.6 ± 1.5
	CORAL	48.4 ± 1.7	84.3 ± 0.5	47.3 ± 0.4
Ours-$s^e(Z)$	ERM	**58.8** ± 0.5	**85.2** ± 0.4	**48.1** ± 0.3
	IRM	47.6 ± 1.6	82.1 ± 0.6	40.2 ± 2.9
	DRO	56.6 ± 2.2	84.3 ± 0.4	43.1 ± 0.6
	CORAL	58.6 ± 0.2	85.1 ± 0.4	46.2 ± 0.4

We use both $b(Z) = Z$ and $b(Z) = s^e(Z)$ to construct balanced mini-batches. For details, see Appendix C.

ColoredMNIST: In Table 1, $b(Z) = s^e(Z)$ significantly outperforms $b(Z) = Z$, and outperforms the random mini-batch sampling baseline by more than 5% (absolute) with all four classifiers. This is likely due to the relatively large noise (25%) in the label assignment: Z captures all other features except the ones directly determines the label, while $s^e(Z)$ only capture the features that has strong correlation with the label Y. i.e. the color information. As Z is finer and contains more information than $s^e(Z)$, it is likely that the closest match by $b(Z) = Z$ would be an image of the same digit with the same color. i.e. The matched example is likely to be an image labeled with the "wrong" class. This will hurt the performance as we up-sample the noise data.

PACS and TerraIncognita: Table 1 shows that our balanced mini-batch sampling method outperforms the random mini-batch sampling baseline on two real-world datasets. We observe that the performance difference between $b(Z) = s^e(Z)$ and $b(Z) = Z$ is not as large as that on the ColoredMNIST dataset. It is partially because these two datasets have less label noise than ColoredMNIST.

5. Conclusion

We propose a causality-based domain generalization method that samples balanced mini-batches to reduce spurious correlations. We show that our assumed data generation model with invariant causal mechanism can be identified up to sample transformations. We demonstrate theoretically that the balanced mini-batch is approximately sampled from a spurious free data distribution with the same causal mechanism under idea scenarios. Our experiments empirically show the effectiveness of our method on both semi-synthetic dataset and real-world datasets.

References

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization, 2020.

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

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

Gulrajani, I. and Lopez-Paz, D. In search of lost domain generalization. In *International Conference on Learning Representations*, 2020.

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

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

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

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 5542–5550, 2017.

Liu, C., Sun, X., Wang, J., Tang, H., Li, T., Qin, T., Chen, W., and Liu, T.-Y. Learning causal semantic representation for out-of-distribution prediction. *Advances in Neural Information Processing Systems*, 34, 2021.

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*, 2022. URL https://openreview.net/forum?id=-e4EXDWXnSn.

Mahajan, D., Tople, S., and Sharma, A. Domain generalization using causal matching. In *International Conference on Machine Learning*, pp. 7313–7324. PMLR, 2021.

Motiian, S., Piccirilli, M., Adjeroh, D. A., and Doretto, G. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 5715–5725, 2017.

Pearl, J. *Causality*. Cambridge university press, 2009.

Quiñonero-Candela, J., Sugiyama, M., Lawrence, N. D., and Schwaighofer, A. *Dataset shift in machine learning*. Mit Press, 2009.

ROSENBAUM, P. R. and RUBIN, D. B. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 04 1983. ISSN 0006-3444. doi: 10.1093/biomet/70.1.41. URL https://doi.org/10.1093/biomet/70.1.41.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Sun, B. and Saenko, K. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pp. 443–450. Springer, 2016.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks, 2014.

Vapnik, V. Statistical learning theory wiley. 1998.

Wald, Y., Feder, A., Greenfeld, D., and Shalit, U. On calibration and out-of-domain generalization. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=XWYJ25-yTRS.

A. Algorithm details

(a) As realized by ColoredMNIST (b) As realized by PACS

Figure 2. Annotated example causal graphs of two realizations of the joint distribution $p(X, Y, E)$.

Algorithm 1 Balanced Mini-batch sampling.

Input: $|\mathcal{E}_{\text{train}}|$ training datasets $\mathcal{D}^e = \{(x_i^e, y_i^e)\}_{i=1}^{N^e}$ sampled from distribution $P(X, Y|E = e)$ for all $e \in \mathcal{E}_{\text{train}}$, a balancing score $b^e(z_i)$ calculated for each training data point (x_i^e, y_i^e), and a distance metrics $d(\cdot, \cdot)$ that calculates the distance between two balancing scores

$D_{balanced} \leftarrow$ Empty
for $e \in \mathcal{E}_{\text{train}}$ **do**
  Randomly sample B data points D_{random}^e from $\mathcal{D}^e$
  Add D_{random}^e to $D_{balanced}$
  for $(x^e, y^e) \in D_{random}^e$ **do**
    Uniformly sample a different labels Y_{alt} from $\mathcal{Y} = \{1, 2, ..., m\}$ such that $y \neq y^e$ for all $y \in Y_{alt}$
    Suppose the balancing score of (x^e, y^e) is $b^e(z)$
    for $y \in Y_{alt}$ **do**
      Search across $\mathcal{D}^e$ for the data point (x_j^e, y_j^e) such that $y_j^e = y$ and has the smallest $d(b^e(z_j), b^e(z))$
      Add (x_j^e, y_j^e) to $D_{balanced}$
    end for
  end for
end for

B. Proofs

In this section, we give full proofs of the main theorems in the paper.

B.1. Balanced mini-batch sampling

B.1.1. Proof for Theorem 3.3

Our proof of all possible balancing scores is an extension of the Proof of Theorem 2 from (ROSENBAUM & RUBIN, 1983), by generalizing the binary treatment to multiple treatments.

Proof. First, suppose the balancing score $b(Z)$ is finer than the propensity score $s(Z)$. By the definition of a balancing score (Definition 1) and Bayes' rule, we have:

$$p(Y|Z, b(Z)) = p(Y|b(Z)) \tag{9}$$

On the other hand, since $b(Z)$ is a function of Z, we have:

$$p(Y|Z, b(Z)) = p(Y|Z) \tag{10}$$

Equation (9) and Equation (10) give us $p(Y|b(Z)) = p(Y|Z)$. So to show $b(Z)$ is a balancing score, it is sufficient to show $p(Y|b(Z)) = p(Y|Z)$.

Let the yth entry of $S(Z)$ be $s_y(Z) = p(Y = y|Z)$, then:

$$\mathbb{E}[s_y(Z)|b(Z)] = \int_{\mathcal{Z}} p(Y = y|Z = z)p(Z = z|b(Z))dz = p(Y = y|b(Z)) \tag{11}$$

But since $b(Z)$ is finer than $s(Z)$, $b(Z)$ is also finer than $s_y(Z)$, then

$$\mathbb{E}[s_y(Z)|b(Z)] = s_y(Z) \tag{12}$$

Then by Equation (11) and Equation (12) we have $P(Y = y|Z) = P(Y = y|b(Z))$ as required. So $b(Z)$ is a balancing score.

For the converse, suppose $b(Z)$ is a balancing score, but that $b(Z)$ is not finer than $s(Z)$, so that there exists z_1 and z_2 such that $s(z_1) \neq s(z_2)$, but $b(z_1) = b(z_2)$. But then, by the definition of $s(\cdot)$, there exists y such that $P(Y = y|z_1) \neq P(Y = y|z_2)$. This means, Y and Z are not conditionally independent given $b(Z)$, thus $b(Z)$ is not a balancing score. Therefore, to be a balancing score, $b(Z)$ must be finer than $s(Z)$.

Note that $s(Z)$ is also a balancing score, since $s(Z)$ is also a function of itself.

□

B.1.2. Proof for Theorem 3.6

We provide a proof for Theorem 3.6, demonstrating the feasibility of balanced mini-batch sampling.

Proof. In Algorithm 1, by uniformly sampling different labels we mean sample $Y_{\text{out}} = \{y_1, y_2, ..., y_\alpha\}$ by the following procedure:

$$\begin{aligned}
&y_1 \sim U\{1, 2, ..., m\} \setminus \{y_e\} \\
&y_2 \sim U\{1, 2, ..., m\} \setminus \{y_e, y_1\} \\
&\quad\vdots \\
&y_\alpha \sim U\{1, 2, ..., m\} \setminus \{y_e, y_1, y_2 ... y_\alpha\}
\end{aligned}$$

Where U denotes the uniform distribution. Suppose $D_{\text{balanced}} \sim \hat{p}(X, Y)$, and data distribution $\mathcal{D}^e \sim p(X, Y|E = e), \forall e \in \mathcal{E}_{\text{train}}$.

Suppose we have an exact match every time we match a balancing score, then for all $e \in \mathcal{E}_{\text{train}}$, we have

$$\begin{aligned}
\hat{p}(Y|b^e(Z), E = e) &= \frac{1}{a+1}p(Y|b^e(Z), E = e) + \frac{1}{a+1}(1 - p(Y|b^e(Z), E = e)\frac{1}{m-1} + \\
&+ \frac{1}{a+1}(1 - p(Y|b^e(Z), E = e)(1 - \frac{1}{m-1})\frac{1}{m-2} + ... \\
&+ \frac{1}{a+1}(1 - p(Y|b^e(Z), E = e)(1 - \frac{1}{m-1})(1 - \frac{1}{m-2})...(1 - \frac{1}{m-a+1})\frac{1}{m-a} \\
&= \frac{1}{a+1}(\frac{a}{m-1} + \frac{m-a-1}{m-1}p(Y|b^e(Z), E = e))
\end{aligned}$$

By the definition of balancing score, $p(Y|Z, E = e) = p(Y|b^e(Z), E = e)$ and $\hat{p}(Y|Z, E = e) = \hat{p}(Y|b^e(Z), E = e)$, then we have

$$\hat{p}(Y|Z, E) = \frac{1}{a+1}(\frac{a}{m-1} + \frac{m-a-1}{m-1}p(Y|Z, E)) \tag{13}$$

When $a = m - 1$, $\hat{p}(Y|Z, E) = \frac{1}{m}$, the labels in D_{balanced} can be regarded as sampled from $U\{1, 2, ..., m\}$. Then D_{balanced} can be regarded as sampled from the balanced distribution as defined in Definitions 3 and 4.

□

B.2. Latent Covariate Learning

B.2.1. Proof for Theorem 3.8

We now prove Theorem 3.8 setting up the identifiability of the necessary parameters that capture the spuriously correlated covariate features in the VAE. The proof of this Theorem is based on the proof of Theorem 1 in (Motiian et al., 2017), with the following modifications:

1. We use both E and Y as auxiliary variables.
2. We include Y in the causal mechanism of generating X by $X = \mathbf{f}(Y, Z) + \epsilon = \mathbf{f}_Y(Z) + \epsilon$.

Proof. **Step I.** In this step, we transform the equality of the marginal distributions over observed data into the equality of a noise-free distribution. Suppose we have two sets of parameters $\theta = (\mathbf{f}, \mathbf{T}, \lambda)$ and $\theta' = (\mathbf{f}', \mathbf{T}', \lambda')$ such that $p_\theta = (X|Y, E = e) = p$-
$\theta(X|Y, E = e)$, $\forall e \in \mathcal{E}_{\text{train}}$, then:

$$\int_{\mathcal{Z}} p_{\mathbf{T},\lambda}(Z|Y, E = e) p_{\mathbf{f}}(X|Z, Y) dZ = \int_{\mathcal{Z}} P_{\mathbf{T}',\lambda'}(Z|Y, E = e) p'_{\mathbf{f}}(X|Z, Y) dZ \tag{14}$$

$$\Rightarrow \int_{\mathcal{Z}} p_{\mathbf{T},\lambda}(Z|Y, E = e) p_\epsilon(X - \mathbf{f}_Y(Z)) dZ = \int_{\mathcal{Z}} p_{\mathbf{T}',\lambda'}(Z|Y, E = e) p_\epsilon(X - \mathbf{f}'_Y(Z)) dZ \tag{15}$$

$$\Rightarrow \int_{\mathcal{X}} p_{\mathbf{T},\lambda}(\mathbf{f}^{-1}(\bar{X})|Y, E = e) \text{vol} J_{\mathbf{f}^{-1}}(\bar{X}) p_\epsilon(X - \bar{X}) d\bar{X} = \int_{\mathcal{X}} p_{\mathbf{T}',\lambda'}(\mathbf{f}'^{-1}(\bar{X})|Y, E = e) \text{vol} J_{\mathbf{f}'^{-1}(\bar{X})} p_\epsilon(X - \bar{X}) d\bar{X} \tag{16}$$

$$\Rightarrow \int_{\mathbb{R}^d} \tilde{p}_{\mathbf{T},\lambda,\mathbf{f},Y,e}(\bar{X}) p_\epsilon(X - \bar{X}) d\bar{X} = \int_{\mathbb{R}^d} \tilde{p}_{\mathbf{T}',\lambda',\mathbf{f}',Y,e}(\bar{X} p_\epsilon(X - \bar{X}) d\bar{X}) \tag{17}$$

$$\Rightarrow (\tilde{p}_{\mathbf{T},\lambda,\mathbf{f},Y,e} * p_\epsilon)(X) = (\tilde{p}_{\mathbf{T}',\lambda',\mathbf{f}',Y,e} * P_{\mathcal{E}})(X) \tag{18}$$

$$\Rightarrow \mathscr{F}[\tilde{p}_{\mathbf{T},\lambda,\mathbf{f},Y,e}](\omega)\phi_\epsilon(\omega) = \mathscr{F}[\tilde{p}_{\mathbf{T}',\lambda',\mathbf{f}',Y,e}](\omega)\phi_\epsilon(\omega) \tag{19}$$

$$\Rightarrow \mathscr{F}[\tilde{p}_{\mathbf{T},\lambda,\mathbf{f},Y,e}](\omega) = \mathscr{F}[\tilde{p}_{\mathbf{T}',\lambda',\mathbf{f}',Y,e}](\omega) \tag{20}$$

$$\Rightarrow \tilde{p}_{\mathbf{T},\lambda,\mathbf{f},Y,e}(X) = \tilde{p}_{\mathbf{T}',\lambda',\mathbf{f}',Y,e}(X) \tag{21}$$

- In Equation (16), we denote the volume of a matrix $\mathbf{A}$ as $\text{vol}\mathbf{A} := \sqrt{\det \mathbf{A}^T \mathbf{A}}$. J denotes the Jacobian. We made the change of variable $\bar{X} = \mathbf{f}_Y(Z)$ on the left hand side and $\bar{X} = \bar{\mathbf{f}}_Y(Z)$ on the right hand side. Since $\mathbf{f}$ is injective, we have $\mathbf{f}^{-1}(\bar{X}) = (Y, Z)$. Here we abuse $\mathbf{f}^{-1}(\bar{X})$ to specifically denote the recovery of Z, i.e. $\mathbf{f}^{-1}(\bar{X}) = Z$.
- In Equation (17), we introduce

$$\tilde{p}_{\mathbf{T},\lambda,\mathbf{f},Y,e}(X) = p_{\mathbf{T},\lambda}(\mathbf{f}_Y^{-1}(X)|Y, E = e) \text{vol} J_{\mathbf{f}_Y^{-1}}(X) \mathbb{1}_{\mathcal{X}}(X) \tag{22}$$

  on the left hand side, and similarly on the right hand side.
- In Equation (18), we use $*$ for the convolution operator.
- In Equation (19), we use $\mathscr{F}[\cdot]$ to designate the Fourier transform, and the characteristic function of ϵ is $\phi_\epsilon = \mathscr{F}[p_\epsilon]$.
- In Equation (20), we dropped $\phi_\epsilon(\omega)$ from both sides as it is non-zero almost everywhere (by assumption (1) of the Theorem).

Step II. In this step, we remove all terms that are either a function of X or Y or e. By taking logarithm on both sides of Equation (21) and replacing $P_{\mathbf{T},\lambda}$ by its expression from Equation (3) we get:

$$\log \text{vol} J_{\mathbf{f}^{-1}}(X) + \sum_{i=1}^{n}(\log Q_i(\mathbf{f}_i^{-1}(X)) - \log W_i^e(Y) + \sum_{j=1}^{k} \mathbf{T}_{i,j}(\mathbf{f}_i^{-1}(X))\lambda_{i,j}^e(Y))$$
$$= \log \text{vol} J_{\mathbf{f}'^{-1}}(X) + \sum_{i=1}^{n}(\log Q_i'(\mathbf{f}_i'^{-1}) - \log W_i^{e\prime}(Y) + \sum_{j=1}^{k} \mathbf{T}_{i,j}'(\mathbf{f}_i'^{-1}(X))\lambda_{i,j}^e{}'(Y)) \tag{23}$$

Let $(e_0, y_0), (e_1, y_1), ..., (e_{nk}, y_{nk})$ be the points provided by assumption (3) of the Theorem. We evaluate the above equations at these points to obtain $k+1$ equations, and subtract the first equation from the remaining k equations to obtain:

$$\langle \mathbf{T}(\mathbf{f}^{-1}(X)), \lambda^{e_l}(y_l) - \lambda^{e_0}(y_0)\rangle + \sum_{i=1}^{n} \log \frac{W_i^{e_0}(y_0)}{W_i^{e_l}(y_l)}$$
$$= \langle \mathbf{T}'(\mathbf{f}^{-1}(X)), \lambda^{e_l\prime}(y_l) - \lambda^{e_0\prime}(y_0)\rangle + \sum_{i=1}^{n} \log \frac{W_i^{e_0\prime}(y_0)}{W_i^{e_l\prime}(y_l)} \tag{24}$$

Let $\mathbf{L}$ be the matrix defined in assumption (3) and $\mathbf{L}'$ similarly defined for λ' ($\mathbf{L}'$ is not necessarily invertible). Define $b_l = \sum_{i=1}^{n} \log \frac{W_i^{e_0'}(y_0) W_i^{e_l}(y_l)}{W_i^{e_0}(y_0) W_i^{e_L'}(y_l)}$ and $\mathbf{b} = [b_l]_{l=1}^{nk}$.

Then Equation (24) can be rewritten in the matrix form:

$$\mathbf{L}^T \mathbf{T}(\mathbf{f}^{-1}(X)) = \mathbf{L}'^T \mathbf{T}'(\mathbf{f}'^{-1}(X)) + \mathbf{b} \tag{25}$$

We multiply both sides of Equation (25) by $\mathbf{L}^{-T}$ to get:

$$\mathbf{T}(\mathbf{f}^{-1}(X)) = \mathbf{A}\mathbf{T}'(\mathbf{f}'^{-1}(X)) + \mathbf{c} \tag{26}$$

Where $\mathbf{A} = \mathbf{L}^{-T}\mathbf{L}'$ and $\mathbf{c} = \mathbf{L}^{-T}\mathbf{b}$.

Step III. To complete the proof, we need to show that $\mathbb{A}$ is invertible. By definition of $\mathbf{T}$ and according to Assumption (2), its Jacobian exists and is an $nk \times n$ matrix of rank n. This implies that the Jacobian of $\mathbf{T}'$

f'$^{-1}$ exists and is of rank n and so is A.

We distinguish two cases:

1. If $k = 1$, then $\mathbf{A}$ is invertible as $\mathbf{A} \in \mathbb{R}^{n \times n}$.
2. If $k > 1$, define $\bar{\mathbf{x}} = \mathbf{f}^{-1}(\mathbf{x})$ and $\mathbf{T}_i(\bar{x}_i) = (T_{i,1}(\bar{x}_i), ..., T_{i,k}(\bar{x}_i))$.

   Suppose for any choice of $\bar{x}_i^1, \bar{x}_i^2, ..., \bar{x}_i^k$, the family $(\frac{d\mathbf{T}_i(\bar{x}_i^1)}{d\bar{x}_i^1}, ..., \frac{d\mathbf{T}_i(\bar{x}_i^k)}{d\bar{x}_i^k})$ is never linearly independent. This means that $\mathbf{T}_i(\mathbb{R})$ is included in a subspace of $\mathbb{R}^k$ of dimension of most $k-1$. Let $\mathbf{h}$ be a non-zero vector that is orthogonal to $T_i(\mathbb{R})$. Then for all $x \in \mathbb{R}$, we have $\langle \frac{d\mathbf{T}_i(x)}{dx}, \mathbf{h}\rangle = 0$. By integrating we find that $\langle \mathbf{T}_i(x), \mathbf{h}\rangle = \text{const}$.

   Since this is true for all $x \in \mathbb{R}$ and for a $h \neq 0$, we conclude that the distribution is not strongly exponential. So by contradiction, we conclude that there exist k points $\bar{x}_i^1, \bar{x}_i^2, ...\bar{x}_i^k$ such that $(\frac{d\mathbf{T}_i(\bar{x}_i^1)}{d\bar{x}_i^1}, ..., \frac{d\mathbf{T}_i(\bar{x}_i^k)}{d\bar{x}_i^k})$ are linearly independent.

   Collect these points into k vectors $(\bar{\mathbf{x}}^1, ..., \bar{\mathbf{x}}^k)$ and concatenate the k Jacobians $J_{\mathbf{T}}(\bar{\mathbf{x}}^l)$ evaluated at each of those vectors horizontally into the matrix $\mathbf{Q} = (J_{\mathbf{T}}(\bar{\mathbf{x}}^1), ..., J_{\mathbf{T}}(\bar{\mathbf{x}}^k))$ and similarly define $\mathbf{Q}'$ as the concatentation of the Jacobians of $\mathbf{T}'(\mathbf{f}'^{-1}\mathbf{f}(\bar{\mathbf{x}}))$ evaluated at those points. Then the matrix Q is invertible. By differentiating Equation (26) for each $\mathbf{x}^l$, we get:

$$\mathbf{Q} = \mathbf{A}\mathbf{Q}' \tag{27}$$

The invertibility of $\mathbf{Q}$ implies the invertibility of $\mathbf{A}$ and $\mathbf{Q}'$. This completes the proof.

□

C. Experiment Details

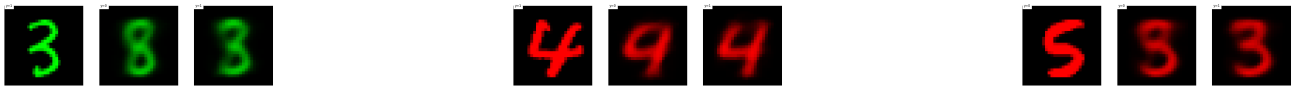

Figure 3. Reconstructed ColoredMNIST images from our VAE model. In each sub-figure, we infer Z from the leftmost image, then generate images with label 0 (middle) and 1 (right).

Baselines: Empirical risk minimization (ERM) is a default training scheme for most machine learning problems, merging all training data into one dataset and minimizing the training errors across all training domains. Invariant risk minimization (IRM) learns a data representation such that the optimal linear classifier on top of it is invariant across training domains. Group distributionally robust optimization (GroupDRO) performs ERM while increasing the weight of the environments with larger errors. Deep CORAL matches the mean and covariance of feature distributions across training domains.

We use a multi-layer perceptron based VAE (Kingma & Welling, 2013) to learn the latent covariate Z and we choose the conditional prior $(Z|Y, E = e)$ to be a Gaussian distribution with diagonal covariance matrix. We also choose the noise distribution p_ϵ to be a Gaussian distribution with zero mean and fixed variance. For the architecture of the image classifiers (ERM (Vapnik, 1998), IRM (Gulrajani & Lopez-Paz, 2020), GroupDRO (Sagawa et al., 2019), CORAL (Sun & Saenko, 2016)), following the setting of DomainBed (Gulrajani & Lopez-Paz, 2020), we train a convolutional neural network from scratch for ColoredMNIST (Arjovsky et al., 2020) dataset, and use a pretrained ResNet50 (He et al., 2016) for PACS (Li et al., 2017) and TerraIncognita (Beery et al., 2018). Each experiment is repeated with 3 different random seeds.

We perform our experiments on the DomainBed codebase[2] and follow its default settings and hyperparameters. For all datasets, we set the number of matched examples to be $a = 1$.

Table 2. Out-of-domain test accuracy on ColoredMNIST dataset. Numbers are averaged over 3 runs with standard deviation.

Sampling	Alg	0.1	0.2	0.9	Avg
Random	ERM	71.8 ± 0.2	72.5 ± 0.1	10.1 ± 0.1	51.5 ± 0.1
	IRM	59.8 ± 1.5	58.1 ± 1.8	9.7 ± 0.0	42.5 ± 1.0
	DRO	72.7 ± 0.2	73.0 ± 0.2	10.0 ± 0.2	51.9 ± 0.0
	CORAL	71.4 ± 0.2	72.8 ± 0.1	9.9 ± 0.0	51.4 ± 0.1
Ours-Z	ERM	63.5 ± 2.2	66.0 ± 0.8	10.0 ± 0.1	46.5 ± 0.9
	IRM	54.4 ± 8.4	69.4 ± 7.5	9.9 ± 0.2	44.6 ± 4.7
	DRO	65.7 ± 1.5	67.1 ± 1.8	14.9 ± 4.0	49.2 ± 0.7
	CORAL	64.7 ± 0.8	65.5 ± 0.9	14.9 ± 4.1	48.4 ± 1.7
Ours-$s^e(Z)$	ERM	72.1 ± 0.1	71.2 ± 0.2	33.1 ± 1.3	58.8 ± 0.5
	IRM	69.8 ± 0.6	62.2 ± 5.6	10.9 ± 0.5	47.6 ± 1.6
	DRO	72.3 ± 0.2	71.4 ± 0.6	25.9 ± 6.7	56.6 ± 2.2
	CORAL	72.0 ± 0.6	71.8 ± 0.6	32.1 ± 0.7	58.6 ± 0.2

ColoredMNIST: The ColoredMNIST (Arjovsky et al., 2020) dataset is generated from the regular MNIST hand-written digits dataset (Lecun et al., 1998) by first assigning a binary label to indicate whether the digit is smaller than 5 to each MNIST image with some random noise, and then assigning a binary color (green or red) based on different correlations with the label across environments. Because of the injected noise in label assignment, the maximum accuracy a classifier can achieve is 75%. As shown in Table 2, there are in total 3 environments in the dataset, each of which has $p(\text{Color=Red}|Y = 1) = 0.1, 0.2, 0.9$, with 70,000 examples in each environment. In this case we have $|\mathcal{E}_{\text{train}}| = 2$ and $m = 2$. We also set $k = 1$ by fixing the variance of the conditional prior $(Z|Y, E = e)$ to be 1. Then we take the maximum possible dimension of the latent Z, $n = 3$, according to the identifiability condition in Theorem 3.8. Note that as $m = 2$, the balanced mini-batches can be regarded as sampling from a balanced distribution with perfect match in balancing score and $a = 1$ as stated in Theorem 3.6.

[2] https://github.com/facebookresearch/DomainBed

Figure 3 shows three sets of reconstructed images with the same latent variable Z and different label Y using our VAE model. We can see that Z keeps the color feature and some style features, while the digit shape is changed with corresponding label Y.

As shown in Table 2, our proposed balanced mini-batch sampling method significantly outperforms random mini-batch sampling in the worst test environment (0.9) by 20% (absolute), while achieving comparable performance on the other two environments. Note that 0.9 is the hardest test environment as the train environments 0.1 and 0.2 are significantly different from it. While the accuracy on 0.9 is low, we are able to get around 75% (maximum) accuracy on the train domain validation sets.

Table 3. Out-of-domain test accuracy on each domain of the PACS dataset. Numbers are averaged over 3 runs. We use Gaussian distribution with $k = 2$ for latent covariate learning and $n = 64$.

Sampling	Alg	A	C	P	S	Avg
Random	ERM	85.0 ± 1.6	77.9 ± 2.2	95.4 ± 0.3	76.4 ± 1.1	83.7 ± 0.1
	IRM	81.4 ± 0.7	76.4 ± 1.3	96.6 ± 0.5	70.5 ± 2.0	81.2 ± 0.4
	DRO	83.6 ± 0.8	79.9 ± 1.2	96.4 ± 0.4	74.9 ± 0.3	83.7 ± 0.4
	CORAL	86.0 ± 0.6	77.6 ± 0.3	95.7 ± 0.1	80.1 ± 0.1	84.8 ± 0.2
Ours-Z	ERM	85.7 ± 0.7	80.2 ± 0.3	96.4 ± 0.2	78.5 ± 2.2	85.2 ± 0.3
	IRM	83.6 ± 1.2	74.7 ± 2.0	96.6 ± 0.3	75.6 ± 1.6	82.6 ± 0.3
	DRO	83.6 ± 1.2	80.1 ± 1.2	95.1 ± 0.5	80.2 ± 1.3	84.7 ± 0.6
	CORAL	82.2 ± 1.9	79.0 ± 1.1	96.2 ± 0.1	79.7 ± 0.8	84.3 ± 0.5
Ours-$s^e(Z)$	ERM	84.3 ± 0.5	79.8 ± 1.0	95.7 ± 0.4	80.9 ± 1.2	85.2 ± 0.4
	IRM	84.2 ± 1.7	73.8 ± 0.7	97.1 ± 0.4	73.4 ± 1.2	82.1 ± 0.6
	DRO	82.7 ± 0.9	79.0 ± 0.9	95.5 ± 0.3	80.0 ± 1.3	84.3 ± 0.4
	CORAL	84.0 ± 1.8	81.6 ± 0.9	94.2 ± 0.9	80.4 ± 1.1	85.1 ± 0.4

Table 4. Out-of-domain test accuracy on on each domain of the TerraIncognita dataset. Numbers are averaged over 3 runs. We use Gaussian distribution with $k = 2$ for latent covariate learning and $n = 29$.

Sampling	Alg	L100	L38	L43	L46	Avg
Random	ERM	50.6 ± 4.1	42.9 ± 2.6	55.8 ± 0.1	35.1 ± 2.0	46.1 ± 1.2
	IRM	34.0 ± 4.3	33.2 ± 9.1	50.4 ± 0.8	9.6 ± 1.1	39.3 ± 1.8
	DRO	50.0 ± 1.9	36.5 ± 4.8	56.3 ± 0.8	31.7 ± 3.1	43.6 ± 1.2
	CORAL	48.5 ± 2.9	37.9 ± 4.0	55.5 ± 2.1	37.3 ± 0.9	44.8 ± 0.3
Ours-Z	ERM	50.5 ± 2.9	45.1 ± 1.1	55.6 ± 0.9	37.4 ± 1.0	47.1 ± 0.6
	IRM	32.0 ± 3.9	40.8 ± 2.9	49.0 ± 1.5	38.7 ± 1.3	40.1 ± 1.4
	DRO	48.7 ± 3.1	23.3 ± 4.2	55.7 ± 1.0	38.5 ± 1.4	41.6 ± 1.5
	CORAL	54.9 ± 2.8	41.3 ± 0.9	55.2 ± 0.7	37.7 ± 1.7	47.3 ± 0.4
Ours-$s^e(Z)$	ERM	51.9 ± 0.4	45.8 ± 2.4	55.0 ± 1.0	39.8 ± 1.3	48.1 ± 0.3
	IRM	37.6 ± 7.2	43.0 ± 1.8	43.5 ± 3.8	36.9 ± 1.8	40.2 ± 2.9
	DRO	49.7 ± 2.7	34.6 ± 2.5	54.5 ± 1.2	33.4 ± 2.7	43.1 ± 0.6
	CORAL	50.1 ± 2.1	42.6 ± 2.1	54.7 ± 0.8	37.3 ± 2.5	46.2 ± 0.4

PACS and TerraIncognita: We also conduct experiments on two more realistic datasets, PACS (Li et al., 2017) and TerraIncognita (Beery et al., 2018). The PACS dataset has four environments, each indicates an image style: art, cartoons, photos and sketches, with 9,991 examples in total. There are 7 classes in the dataset, indicating the object in the image. The TerraIncognita dataset contains photographs of wild animals taken by camera traps at four different locations: L100, L38, L43 and L46, with 24,788 examples in total. There are 10 classes in the dataset, indicating the animal appears in the image.

In these two datasets, we do not require $|\mathcal{E}_{\text{train}}|$, m, n and k to explicitly satisfy the identifiability condition in Theorem 3.8, as the latent variable could require a larger dimensionality n to capture than the dataset allows. Also, as $m > 2$, $a = 1$ means we cannot completely eliminate the spurious correlation in the training dataset in the ideal scenario.

All experiments were conducted on NVidia A-100 and Titan X GPUs. The DomainBed codebase is released under an MIT license at https://github.com/facebookresearch/DomainBed. We utilize versions of the datasets ColoredMNIST, PACS, and

TerraIncognita (Caltech Camera Traps) that are distributed within DomainBed, under the Creative Commons Attribution-NonCommercial 4.0, CopyLeft/No Rights Reserved, and Community Data License Agreement (CDLA) licenses respectively. Our code and corresponding instructions are included in the supplementary materials.