Class Distribution Shifts in Zero-Shot Learning: Learning Robust Representations

Yuli Slavutsky
Department of Statistics and Data Science
The Hebrew University of Jerusalem
Jerusalem, Israel
yuli.slavutsky@mail.huji.ac.il

Yuval Benjamini
Department of Statistics and Data Science
The Hebrew University of Jerusalem
Jerusalem, Israel
yuval.benjamini@mail.huji.ac.il

Abstract

Zero-shot learning methods typically assume that the new, unseen classes encountered during deployment come from the same distribution as the the classes in the training set. However, real-world scenarios often involve class distribution shifts (e.g., in age or gender for person identification), posing challenges for zero-shot classifiers that rely on learned representations from training classes. In this work, we propose and analyze a model that assumes that the attribute responsible for the shift is unknown in advance. We show that in this setting, standard training may lead to non-robust representations. To mitigate this, we develop an algorithm for learning robust representations in which (a) synthetic data environments are constructed via hierarchical sampling, and (b) environment balancing penalization, inspired by out-of-distribution problems, is applied. We show that our algorithm improves generalization to diverse class distributions in both simulations and experiments on real-world datasets.

1 Introduction

Zero-shot learning systems [14, 27] are designed to classify instances of new, previously unseen classes at deployment, a scenario known as *open-world* classification. These systems are widely applied in extreme multi-class applications, such as face or voice recognition [19] for matching observations of the same individual, and more generally, for learning data representations [2].

Class distribution shifts typically refer to changes in the prevalence of a fixed set of classes between training and testing. In zero-shot learning, however, a different challenge arises: the appearance of entirely new classes at test time. This raises a critical question – are these new classes drawn from the same distribution as the training classes? Most zero-shot methods assume that they are, an assumption that not only shapes the design of test sets [57, 16] but also plays an explicit role in assessing the generalization capabilities of zero-shot classifiers [59, 48].

In practice, training classes are often chosen based on convenience and accessibility during data collection. Even when data is carefully collected, the distribution of classes may shift over time, leading to a different distribution. For instance, this could occur when a face recognition system is deployed in a building located in a neighborhood undergoing demographic changes.

Class distribution shifts pose significant challenges to zero-shot classifiers, since they rely on learning data representations from the training classes to distinguish new, unseen ones. Typically, these classifiers are trained by minimizing the loss on the training set to effectively separate the training classes. However, this approach may result in poor performance when confronted with data from distributions that differ significantly from the class distribution in the training data. Notably, in person re-identification, this concern gained attention from a fairness perspective with respect to gender

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

[15, 23], age [5, 33, 50], and racial [39, 54] bias. In all these studies the variable (i.e., gender, age, race) expected to cause the distribution shift was known in advance.

In contrast, in real-world scenarios, the attribute responsible for a future distribution shift is usually unknown during training. In such cases, existing approaches based on collecting balanced datasets or re-weighting training examples [54, 41, 53] are inapplicable. Furthermore, while class distribution shifts have been extensively studied in the standard setting of supervised learning (see Appendix A), previous research assumed a *closed-world* setting that does not account for new classes at test time. Instead, it only addressed changes in the prevalence of fixed classes between training and testing. Consequently, class distribution shifts in zero-shot learning remain largely unaddressed.

In this paper we first address these limitations by examining the effects of class distribution shifts on constrastive zero-shot learning, by proposing and analyzing a parametric model (§3). We identify conditions where minimizing loss in this model leads to representations that perform poorly when a distribution shift has occurred.

We then use the insights gained from this model to present our second contribution (§4): an algorithm for learning representations that are robust against class distribution shifts in zero-shot classification. In our proposed approach, *artificial data environments* with diverse attribute distributions are constructed using hierarchical subsampling, and an *environment balancing* criterion inspired by out-of-distribution (OOD) methods is applied. We assess our method's effectiveness in both simulations and experiments on real-world datasets, demonstrating its enhanced robustness in §5.

1.1 Problem Setup

Let $\{z_i, c_i\}_{i=1}^{N_z}$ be a labeled set of training data points $z \in \mathcal{Z}$ and classes $c \in \mathcal{C}$, such that c_i is the class of z_i.

In this work, we focus on verification algorithms that enable *open-world* classification by determining whether two data points $x_{ij} := (z_i, z_j)$ belong to the same class. For instance, in person re-identification the task is to identify whether two data points (e.g., face images or voice recordings) belong to the same person. We denote this by y_{ij}, where $y_{ij} = 1$ if $c_i = c_j$ and $y_{ij} = 0$ otherwise. When the identity of each data point in the pair is not important, a single index is used for simplicity, namely (x_k, y_k).

We assume that each class c is characterized by some attribute A. We further assume that the training classes are sampled from $P_C(c)$, the test classes are sampled according to $Q_C(c)$, and the two distributions differ solely due to a shift in the distribution of an attribute A:

$$P_C(c) = \int P_{C|A}(c|a)\boldsymbol{P}_A(a)\, da, \quad Q_C(c) = \int P_{C|A}(c|a)\boldsymbol{Q}_A(a)\, da. \tag{1}$$

Importantly, we assume that the attribute A is unknown, and that both during training and testing, data points $z \in \mathcal{Z}$ for each class are sampled according to $P_{Z|C}(z|c)$. For instance, revisit the person identification example where each person is a class. If the attribute A is binary (e.g., a_1 is blond and a_2 is dark-haired), then $P(C|A = a_1)$ represents the distribution of people with blond hair, and $P(C|A = a_2)$ of individuals with other hair colors. The training classes might be predominantly sampled from the blond population $P(A = a_1) = \rho_{\text{tr}} = 0.8$, while test classes are predominantly sampled from $Q(A = a_1) = \rho_{\text{te}} = 0.1$.

We focus on verification techniques based on *deep metric learning* methods (for surveys see [43, 34]) such as contrastive-learning [17], Siamese neural networks [24], triplet networks [20], and other more recent variations [35, 49, 56, 58]. These methods learn a representation function that maps data points to a representation space $g : \mathcal{Z} \to \hat{\mathcal{Z}}$, so that examples from the same class are close (in a predefined distance function $d(\cdot, \cdot)$), while those from different classes are farther apart.

We assume that g is a neural network trained by optimizing a deep-metric-learning loss, such as the contrastive loss [17]:

$$\ell\left(z_i, z_j, y_{ij}; d_g\right) := y_{ij} d_g^2\left(z_i, z_j\right) + \left(1 - y_{ij}\right) \max\left\{0, m - d_g\left(z_i, z_j\right)\right\}^2 \tag{2}$$

where $m \geq 0$ is a predefined margin, and $d_g(z_i, z_j) := d(g(z_i), g(z_j))$ is the distance between the representations of the datapoints z_i, z_j. In our theoretical analysis, we examine the no-hinge

contrastive loss (see Appendix B for additional details):

$$\widetilde{\ell}\left(z_i, z_j, y_{ij}; d_g\right) := y_{ij} d_g^2\left(z_i, z_j\right) + \left(1 - y_{ij}\right)\left(m - d_g\left(z_i, z_j\right)\right)^2. \tag{3}$$

To evaluate the class separation capability of a representation, we treat the distances between representations, $d_g(z_i, z_j)$, as classification scores. Following common practice in the field (e.g., [47, 22]), we use the area under the receiver operating characteristic curve (AUC) to evaluate the representation, enabling threshold-agnostic assessment:

$$AUC(g) \coloneqq P(d_g(z_i, z_j) < d_g(z_u, z_v) | y_{ij} = 1, y_{uv} = 0). \tag{4}$$

Our goal is to learn a representation g that is robust to class attribute shifts. That is, such that for an unknown shifted distribution Q_A, the performance $\mathbb{E}_{Q_A}\left[\text{AUC}(g)\right]$ does not significantly deteriorate compared to the performance obtained on the training distribution P_A.

2 Background on Environment Balancing Methods in OOD Generalization

The field of OOD generalization gained attention since the work of Peters et al. [36], [37], which deals with closed-world classification where training data is gathered from multiple environments E_{train}. In this setting it is assumed that in each environment $e \in E_{\text{train}}$ examples share the same joint distribution $P^e_{C,Z}(c, z)$, but across environments the joint distribution changes, often due to variations in $P^e_{Z|C}(z|c)$. A well-known example [1] involving the classification of images of cows and camels demonstrates how an algorithm relying on background cues during training (e.g., cows in green pastures, camels in deserts) performs poorly on new images of cows with sandy backgrounds.

Several approaches that rely on access to diverse training environments were proposed to identify stable relations between the data point z and its class c. Examples of such stable relations include choosing causal variables using statistical tests [42], leveraging conditional independence induced by the common causal mechanism [9], and using multi-environment calibration as a surrogate for OOD performance [52].

Most relevant to our work are methods that aim to balance the loss over multiple environments. These methods consider a representation $g = g_\theta$ that is a neural network parameterized by θ trained to optimize an objective of the form

$$\min_\theta \sum_{e \in E_{\text{train}}} \ell^e(g_\theta) + \lambda R(g_\theta, E_{\text{train}}) \tag{5}$$

where $\ell^e(g_\theta)$ is the empirical loss obtained on the environment e, E_{train} is the set of all training environments, R is a regularization term designed to balance performance over multiple environments, and λ is a regularization factor balancing the tradeoff between the empirical risk minimization (ERM) term and the balance penalty. Below, we describe three such methods, which we will refer to later in the paper.

Invariant risk minimization (IRM) presented in [1], aims to find data representations g_θ such that the optimal classifier w on top of the data representation $w \circ g_\theta$ is shared across all environments. Therefore, the authors proposed minimizing the sum of environment losses $\ell^e(w \circ g_\theta)$ over all training environments such that $w \in \arg\min_{w'} \ell^e(w' \circ g_\theta)$ for all $e \in E_{\text{train}}$. However, since this objective is too difficult to optimize, a relaxed version was also proposed, taking the form of Equation 5 with a penalty that measures how close w is to minimizing $\ell^e(w \circ g_\theta)$: $R^e_{\text{IRMv1}}(g_\theta) = \left\|\nabla_{w|w=1} \ell^e\left(w \cdot g_\theta\right)\right\|^2$.

Note that for loss functions for which optimal classifiers can be expressed as conditional expectations, the original IRM objective is equivalent to the requirement that for all environments $e, e' \in E_{\text{train}}$, $\mathbb{E}_{P^e_{C,Z}}\left[c | g(z) = h\right] = \mathbb{E}_{P^{e'}_{C,Z}}\left[c | g(z) = h\right]$, where $P^e_{C,Z}$ and $P^{e'}_{C,Z}$ are the joint data distributions in the respective environments.

Calibration Loss Over Environments (CLOvE) presented in [52], leverages the equivalence above to establish a link between multi-environment calibration and invariance for binary predictors ($c \in \{0, 1\}$). The proposed regularizer is based on the *maximum mean calibration error* (MMCE) [26]. Let $s : \hat{\mathcal{Z}} \to [0, 1]$ be a classification score function applied on the representation $s \circ g$, and

$s_i = \max\{s \circ g(z_i), 1 - s \circ g(z_i)\}$ be the *confidence* on the i-th data point. Denote the *correctness* as $b_i = \mathbb{1}\{|c_i - s_i| < \frac{1}{2}\}$, and let $K : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ be a universal kernel. Let Z^e denote the training data in the environment e. The authors proposed using the MMCE as a penalty in an objective that takes the form of Equation 5 with $R^e_{\text{MMCE}}(s, g_\theta) = \frac{1}{m^2}\sum_{z_i, z_j \in Z^e}(b_i - s_i)(b_j - s_j) K(s_i, s_j)$.

Variance Risk Extrapolation (VarREx) proposed by Krueger et al. [25], is based on the observation that reducing differences in loss (risk) across training domains can reduce a model's sensitivity to a wide range of distribution shifts. The authors found that using the variance of losses as a regularizer is stabler and more effective compared to other penalties. Therefore, they propose the following regularization term for n training environments: $R_{\text{VarREx}}(g_\theta, E_{\text{train}}) = \text{Var}\left(\ell^{e_1}(g_\theta), \ldots, \ell^{e_n}(g_\theta)\right)$.

While simple and intuitive, this approach assumes that losses across different environments accurately reflect the classifier's performance. However, as discussed in §4, this is often not true for deep metric learning losses, where significant changes in loss may correspond to only minor variations in performance.

3 Parametric Model of Class Distribution Shifts in Zero-Shot Learning

In this section, we introduce a parametric model of class distribution shifts. Our model shows that in zero-shot learning, even if the conditional distribution of data given the class $P(z|c)$ remains the same between training and testing, a shift in the class distribution from $P(c)$ to $Q(c)$ can cause poor performance on newly encountered classes sampled from the shifted distribution $Q(c)$.

Assume that for all classes, the data points $z_i \in \mathbb{R}^d$ are sampled from $z_i|c_i \sim \mathcal{N}(c_i, \Sigma_z)$, where $\Sigma_z = \nu_z \cdot I_d$, and I_d is the identity matrix. Let the attribute A indicate the type of a class c, with two possible types: a_1 and a_2. Assume that the classes c_i are drawn according to $c_i \sim \mathcal{N}(0, \Sigma_a)$ for $a \in \{a_1, a_2\}$. Finally, assume that in training, a_1 is the majority type with $P(a_1) = \rho_{\text{tr}} \gg 0.5$, whereas at test time, a_2 is the majority type with $Q(a_1) = \rho_{\text{te}} \ll 0.5$.

We construct the model such that differences between the training class distribution $P(c)$ and the test distribution $Q(c)$ stem solely from a shift in the mixing probabilities of an unknown attribute A (see Equation 1). Therefore, we define Σ_a as a diagonal matrix with replicates of three distinct values on its diagonal: ν_0, ν^+, ν^-. Let $0 < \nu^- < \nu_z \leq \nu_0 < \nu^+$. Then, in the coordinates corresponding to ν_0 and ν^+ data points from different classes are well separated, whereas in the coordinates corresponding to ν^- they are not. Assume the coordinates corresponding to ν_0 are shared by both types, but ν^+ and ν^- are swapped:

$$\Sigma_{a_1} = \text{diag}\big(\overbrace{\nu_0, \ldots, \nu_0}^{d_0}, \overbrace{\nu^+, \ldots, \nu^+}^{d_1}, \overbrace{\nu^-, \ldots, \nu^-}^{d_2}\big),$$
$$\Sigma_{a_2} = \text{diag}\big(\nu_0, \ldots, \nu_0, \nu^-, \ldots, \nu^-, \nu^+, \ldots, \nu^+\big).$$

An illustration with one replicate of each value is shown in Figure 1.

The following proposition shows that if the number of dimensions d_1 that allow good separation for classes of type a_1 is relatively similar to the number of dimensions d_2 that enable good separation for classes of type a_2, specifically if $h_l(\rho, \nu_z, \nu_0, \nu_1, \nu_2) < \frac{d_2+2}{d_1+2} < h_u(\rho, \nu_z, \nu_0, \nu_1, \nu_2)$, then the optimal solution for the training distribution prioritizes the components (features) corresponding to ν^+ for classes of type a_1. Thus, the prioritized features allow good separation for classes from the majority type in training, but offer poor separation for the shifted test distribution, where most classes are of type a_2. Note that if d_2 is large, when combined, the corresponding components may still provide reasonable separation. We define h_l and h_u in Equation 35 and provide the proof of Proposition 1 in Appendix B.2.

Proposition 1. *Consider a weight representation $g(z) = Wz$, where $W \in \mathbb{R}^{d \times d}$ is a diagonal matrix, and the squared Euclidean distance $d_g(z_i, z_j) = \|W(z_i - z_j)\|^2$. Let $W^* = diag(w^*) \in \arg\min_W \mathbb{E}\left[\widetilde{\ell}(\cdot, \cdot, \cdot; d_g)\right]$. Denote $\overline{w}_1^{*2} = \frac{1}{d_1}\sum_{k=d_0+1}^{d_1} w_k^*$ and $\overline{w}_2^{*2} = \frac{1}{d_2}\sum_{k=d_1+1}^{d} w_k^{*2}$. Then, for all $\rho > \frac{1}{2}$ and $\nu_z, \nu_0, \nu_1, \nu_2, d_1, d_2$ satisfying $h_l(\rho, \nu_z, \nu_0, \nu_1, \nu_2) < \frac{d_2+2}{d_1+2} < h_u(\rho, \nu_z, \nu_0, \nu_1, \nu_2)$ it holds that $d_2\overline{w}_2^{*2} \leq d_1\overline{w}_1^{*2}$.*

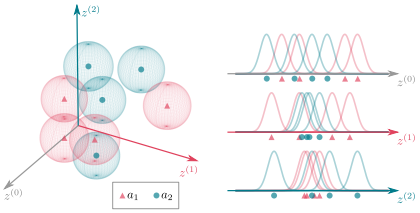

Figure 1: Illustration of the parametric model. Classes of each type are best separated along specific axes: classes of type a_1 along the red axis ($z^{(1)}$) and classes of type a_2 along the green axis ($z^{(2)}$). On axis $z^{(0)}$ both types can be separated but not as effectively as on their respective optimal axes.

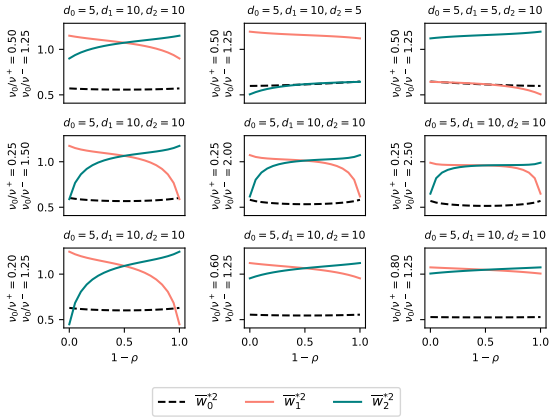

Figure 2: Optimal weights. Top row: d_0 is fixed, d_1 and d_2 vary. Middle and bottom rows: d_0, d_1, d_2 are fixed. Middle: ν_0/ν^- varies. Bottom: ν_0/ν^+ varies.

Note that the conditions outlined in Proposition 1 are sufficient but not necessary. Accordingly, in Appendix B.3, we provide the complete analytical solution for w^* that minimizes the expected loss $\mathbb{E}\left[\widetilde{\ell}(\cdot,\cdot,\cdot;d_g)\right]$ for the weight representation $g(z) = Wz$, using the squared Euclidean distance. According to Proposition 1, larger d_2 values favor ν^- for better aggregated separation. Increasing ν_0/ν^+ leads to increased differences between w_1^{*2} and w_2^{*2}, and vice versa for ν_0/ν^-.

These relationships in the optimal solution are illustrated in Figure 2, showcasing different scenarios. The top row shows that when $d_1 = d_2 = 10$ dimensions favoring classes of type a_1 are prioritized for $\rho > 0.5$, while those favoring type a_2 are prioritized for $\rho < 0.5$. When $d_1 = 10$ while $d_2 = 5$, dimensions favoring type a_1 are prioritized for all values of ρ, and vice versa when d_2 is significantly larger than d_1. The middle and the bottom row further explore the $d_1 = d_2$ case, showing how differences in separability between shared dimensions (ν_0) and type-favoring dimensions impact weight allocation.

Since components corresponding to ν^+ for classes of type a_1 align with ν^- for classes of type a_2, the optimal representation for the training distribution results in poor separation for the shifted test distribution. Therefore, a robust representation should prioritize dimensions that provide effective separation for both class types, corresponding to ν_0.

This aligns with a common principle in the OOD generalization field, where robust representations are those that rely on features shared across environments (see §2). This principle is often referred to as *invariance*.

4 Proposed Approach

Motivated by our analysis of the parametric model, we propose a new approach for tackling class distribution shifts in zero-shot learning. Our approach revolves around two key ideas: (i) during training, different mixtures of the attribute A can be produced by sampling small subsets of the classes, forming artificial environments, and (ii) penalizing for differences in performance across these environments is likely to increase robustness to the class mixture encountered at test time.

4.1 Synthetic Environments

Standard ERM training involves sampling pairs of data points (z_i, z_j) uniformly at random from all N_c classes available during training. However, as discussed in §3, this is prone to overfitting to the attribute distribution of the training data. Since the identity of the attribute is unknown, weighted sampling (and similar approaches) cannot be used to create environments with different attribute mixtures.

Yet, our goal is to design artificial environments with diverse compositions of the (unknown) attribute of interest. To do so, we leverage the variability in small samples: while class subsets of similar size to N_c maintain attribute mixtures similar to the overall training set, smaller subsets with $k \ll N_c$ classes are likely to exhibit distinct attribute mixtures. Therefore, we propose creating multiple environments, composed of examples from few sampled classes.

This results in a hierarchical sampling scheme for the data pairs: first, sample a subset of k classes, $S = \{c_1, \ldots, c_k\}$. Then, for each $c \in S$ sample $2r$ pairs of data points as follows: r pairs from within the class c, $\{z_i; c_i = c\}$, uniformly at random (positive pairs); and r negative pairs, where one point is sampled uniformly at random from c, and the other from all other data points in S, $\{z_i; c_i \neq c, c_i \in S\}$.[1]

Across multiple class subsets $S = \{S_1, \ldots, S_n\}$, this hierarchical sampling results in diverse mixtures of any unknown attribute (see Figure 3). In particular, in some of the class subsets, classes from the overall minority type constitute the majority in the environment.

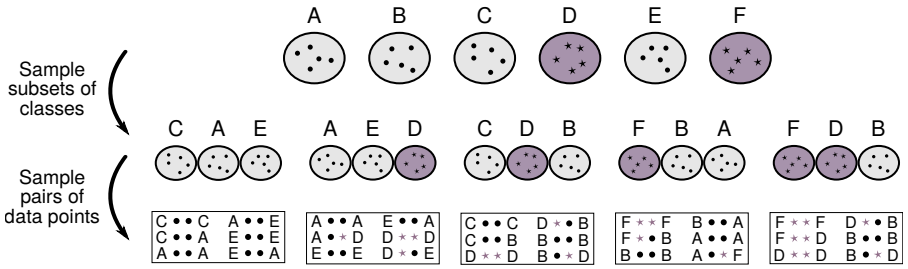

Figure 3: Illustration of the proposed hierarchical sampling. Top: $N_c = 6$ classes, with 2 minority-type classes D, F (in purple). Middle: synthetic environments formed by sampling small ($k = 3$) class subsets; in $1/5$ of the environments, minority-type classes become the majority constituting $2/3$ of the classes. Bottom: sampling $r = 1$ positive and $r = 1$ negative pairs for each class in the environment.

4.2 Environment Balancing Algorithm for Class Distribution Shifts

Our goal is to learn data representations that will allow separation between classes without knowing which attribute is expected to change and how significantly. Therefore, we require the learned data representation to perform similarly well on all mixtures obtained on the synthetic environments.

To achieve this, inspired by OOD performance balancing methods (see §2), we optimize a penalized objective:

$$\min_{\theta} \quad \sum_{l=1}^{n} \ell^{S_l}(g_\theta) + \lambda R\left(S_1, \ldots, S_n\right) \tag{6}$$

where $R\left(S_1, \ldots, S_n\right)$ is any balancing term between the constructed synthetic environments.

Note that computing $R\left(S_1, \ldots, S_n\right)$ often involves evaluating some value on each environment separately. For a general balancing term, we denote the value in the l-th environment as $f(S_l)$ and accordingly express $R\left(S_1, \ldots, S_n\right) = \mathring{f}\left(f(S_1), \ldots, f(S_n)\right)$, where $\mathring{f}$ represents the corresponding aggregation function. Our approach[2] for balancing performance across synthetic environments of class subsets, is outlined in Algorithm 1.

4.3 Balancing Performance Instead of Loss

In multiple OOD penalties (e.g., IRM and VarREx), f represents the loss in each environment, which, in deep metric learning algorithms, is based on distance. This presents a challenge in zero-

[1] Here, for simplicity we create balanced environments, but different proportions of positive examples can be considered instead.

[2] For notation simplicity we assume that the unpenalized training loss is applied to pairs of data points $(x_{ij}, y_{ij}) = ((z_i, z_j), \mathbb{1}_{c_i = c_j})$, but it can easily be adapted for any tuple size (e.g., triplets).

Algorithm 1 Robust Zero-Shot Representation

Input: Labeled data $D = \{z_i, c_i\}_{i=1}^{N_z}$, number of synthetic environments n, number of classes within subset k, number of pairs per class $2r$, neural network $g(\cdot;\theta)$, loss ℓ, distance function d, regularization functions f, $\mathring{f}$, initial weights θ_0, number of training iterations T, learning rate η
Output: Learned representation $g(\cdot;\theta_T)$
Compute unique classes $C^* = \{c^{(1)}, \ldots, c^{(N_c)}\}$
for $t = 1$ **to** T **do**
  for $l = 1$ **to** n **do**
    Sample k classes from C^* without replacement: $S_l^{(t)} = \{c_l^{(1)}, \ldots, c_l^{(k)}\}$.
    From each class in $S_l^{(t)}$ sample r positive and r negative data pairs. Denote the set by $D_l^{(t)}$.
    Compute $f(S_l^{(t)})$.
    Compute average unpenalized loss over $(x_m, y_m) \in D_l^{(t)}$: $\quad \bar{\ell}_l^{(t)} = \frac{1}{2rk} \sum_{m=1}^{2rk} \ell(x_m, y_m)$.
  end for
  Compute $R^{(t)} \coloneqq R\left(S_1^{(t)}, \ldots, S_n^{(t)}\right) = \mathring{f}\left(f(S_1^{(t)}), \ldots, f(S_n^{(t)})\right)$.
  Update network parameters performing a gradient descent step:
  $\theta^{(t)} \leftarrow \theta^{(t-1)} - \eta \nabla_\theta \left(\frac{1}{n} \sum_{l=1}^{n} \bar{\ell}_l^{(t)} + R^{(t)}\right)$
end for
Return: $g(\cdot;\theta_T)$

shot verification, where sampled tuples often include numerous easy negative examples, leading to performance plateau early in the learning process, although the distances themselves still exhibit considerable variations. Strategies like selecting the most difficult tuples [18] were proposed to address this issue, however these methods have been found to generate noisy gradients and loss values [34].

We therefore propose to balance performance directly instead of relying on the losses in the training environments. Denote the set of negative pairs in a synthetic environment by $D_l^0 = \{x_{ij} = (z_i, z_j) : c_i, c_j \in S_l, y_{ij} = 0\}$ and the set of positive pairs by $D_l^1 = \{x_{ij} = (z_i, z_j) : c_i, c_j \in S_l, y_{ij} = 1\}$. An unbiased estimator of the AUC on a given synthetic environment S_l is given by

$$\widetilde{\mathrm{AUC}}\left(S_l; d_g\right) = \frac{1}{|D_l^0|\,|D_l^1|} \sum_{x_{ij}} \sum_{x_{uv}} \mathbb{1}\left[d_g(x_{ij}) < d_g(x_{uv})\right] \tag{7}$$

for $x_{ij} \in D_l^1$ and $x_{uv} \in D_l^0$. Since this estimator is non-differentiable and therefore cannot be used in gradient-descent-based optimization, we use *soft-AUC* as an approximation [7]

$$\widehat{\mathrm{AUC}}\left(S_l; d_g\right) \frac{1}{|D_l^0|\,|D_l^1|} \sum_{x_{ij}} \sum_{x_{uv}} \sigma_\beta\left(d_g(x_{uv}) - d_g(x_{ij})\right) \tag{8}$$

where a sigmoid $\sigma_\beta(t) = \frac{1}{1+e^{-\beta t}}$ approximates the step function. Note that when $\beta \to \infty$, σ_β converges pointwise to the step function. Consequently, we propose the penalty:

$$R_{\text{VarAUC}}\left(S_1, \ldots, S_n; g_d\right) = \widehat{\mathrm{Var}}\left(\widehat{\mathrm{AUC}}\left(S_1; g, d\right), \ldots, \widehat{\mathrm{AUC}}\left(S_n; g, d\right)\right). \tag{9}$$

4.4 How Many Environments Are Needed?

The proposed hierarchical sampling scheme allows for the construction of many synthetic environments with various attribute mixtures, influenced by the number of classes in each environment. As shown in the analysis below, this ensures that with high probability there will be at least one environment with a pair of minority type classes, thereby supporting learning to separate negative pairs within the minority type.

In each training iteration, we consider n class subsets (environments) of size k. Our goal is to achieve robustness to all attribute values a that are associated with at least $\rho_{\text{min}} \in (0, 1)$ of the training classes. Note that ρ_{min} is specified by the practitioner without knowledge of the true attribute that may cause the shift or its true prevalence ρ in the training set.

We compute the number of synthetic environments n, such that with high probability of $(1-\alpha)$, $S_1, \ldots, S_n$ will include at least one subset with at least two classes associated with a (otherwise none of the subsets would contain negative pairs with the attribute a). Denote the probability of a given subset not to contain any class associated with a by $\phi_0 = \binom{\lceil (1-\rho_{\min})N_c \rceil}{k} / \binom{N_c}{k}$ and the probability of a given subset to contain exactly one such class by $\phi_1 = \rho_{\min} N_c \binom{\lceil (1-\rho_{\min})N_c \rceil}{k-1} / \binom{N_c}{k}$. Therefore, the required number of environments needed to ensure that at least two minority-type classes appear together in the same environment is

$$n \approx \frac{\log(\alpha)}{\log(\phi_0 + \phi_1)}. \tag{10}$$

Note that that n is typically much smaller than $\binom{N_c}{k}$.

5 Empirical Results

Our method enhances standard training with two components: a hierarchical sampling scheme and a balancing term for synthetic environments. To the best of our knowledge, this is the first work addressing OOD generalization for class distribution in zero-shot learning. We therefore benchmark our algorithm against the ERM baseline (uniform random sampling with an unpenalized score) and a hierarchical sampling baseline (hierarchical sampling with unpenalized score). Additionally, we tested standard regularization techniques including dropout and the L_2 norm, which did not yield notable improvements in the distribution shift scenario, and are therefore not shown.

To ensure a comprehensive comparison, in addition to the proposed VarAUC penalty, we evaluate variants of our algorithm in which the IRM, CLOvE, and VarREx penalties are used instead. While we show that VarAUC consistently outperforms other penalties, the crucial improvement lies in its performance compared to the ERM baseline: application of existing OOD penalties is enabled by the construction of synthetic environments in our algorithm. As discussed in Appendix C, this construction facilitates the formulation of class distribution shifts in zero-shot learning within the OOD setting.

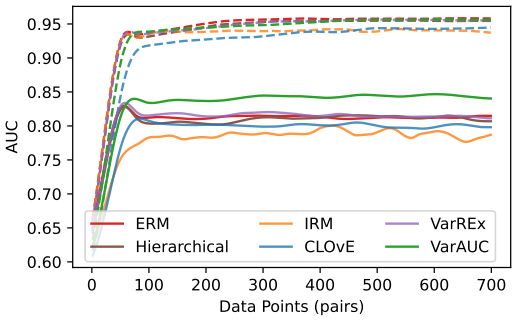

Figure 4: Average AUC over 10 simulation repetitions for majority attribute proportion $\rho = 0.9$ in training (and 0.1 in test). Solid lines: distribution-shift. Dashed lines: in-distribution. Our method improves robustness for shifts, without compromising training distribution results.

In all of the experiments performed, we trained the network with contrastive loss (Equation 2) and the normalized cosine distance: $d_g(z_1, z_2) = \frac{1}{2}\left(1 - \frac{g(z_1)\cdot g(z_2)}{\|g(z_1)\|\|g(z_2)\|}\right)$. The specific setups are detailed below (additional details can be found in Appendix F), and code to reproduce our results is available at https://github.com/YuliSl/Zero_Shot_Robust_Representations .

5.1 Simulations: Revisiting the Parametric Model

We now revisit the parametric model presented in §3. To increase the complexity of the problem, we add dimensions where classes from both types are not well separated. That is, Σ_a includes additional dimensions set to zero.

Setup We used 68 subsets in each training iteration, each consisting of two classes. This corresponds to choosing $\rho_{\min} = 0.1$ (desired sensitivity, regardless of the true unknown parameter $\rho \in \{0.05, 0.1, 0.3\}$), with a low α value of 0.5, resulting in the construction of fewer environments according to Equation 10. For each class, we sampled $2r = 10$ pairs of data points. The representation was defined as $g(z) = wz$ for $w \in \mathbb{R}^{d\times p}$ [3]. Here we focus on the case of $p = 16$, $\nu_z = \nu_0 = 1$,

[3] A linear representation is chosen to facilitate an analysis of the learned representation space.

$\nu^- = 0.1$, $\nu_+ = 2$, $d_0 = 5$, $d_1 = d_2 = 10$. The results for additional representation sizes p, noise ratios $\frac{\nu^+}{\nu^-}$ and varying proportions of positive and negative examples are presented in Appendix D.1.

To assess the importance assigned to each dimension, we examine weight values relative to other weights: $$\text{Importance}_i = \left| \frac{\sum_{j=1}^{p} w_{ij}}{\sum_{i'=1}^{d} \sum_{j'=1}^{p} w_{i'j'}} \right| . \quad (11)$$

Results In Figure 5 we examine the learned representation. The analysis indicates that ERM prioritizes dimensions 5-15, providing good separation for a_1, the dominant type in training, but leading to poor separation after the shift. ERM assigns low weights to dimensions beneficial for both types (0-5) and those suitable for a_2 (15-25). In contrast, our algorithm, particularly with the two variance-based penalties, assigns the lowest weights to dimensions corresponding to a_1 and higher weights to shared dimensions and those that effectively separate a_2 classes.

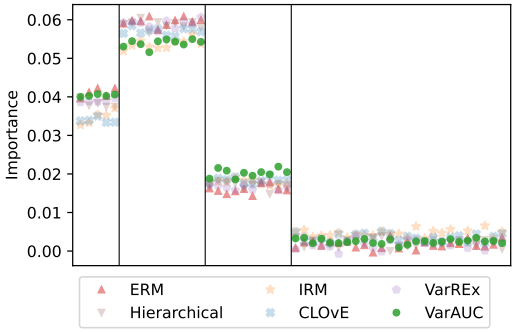

Figure 5: Average feature importance for $\rho = 0.9$, 10 repetitions. Our VarAUC penalty favors shared features (blocks 1 and 3), while deprioritizing majority features (block 2). All methods assign low weight to noise features (block 4).

In Figure 4, the learning progress is depicted for $\rho = 0.9$ (a similar analysis for $\rho = 0.95$ and $\rho = 0.7$ can be found in Appendix D). Performance on the same distribution as the training data is similar for ERM and our algorithm, suggesting that applying our algorithm does not negatively impact performance when no distribution shift occurs. However, when there is a distribution shift our algorithm achieves much better results. The VarREx penalty achieves high AUC values more quickly than the VarAUC penalty, but the VarAUC penalty attains higher overall accuracy. IRM shows noisier convergence, since it is applied directly on the gradients, which have been shown to be noisy in contrastive learning due to high variance in data-pair samples [34]. Means and standard deviations are reported in Appendix D.1, as well as the results for additional data dimensions, positive proportions, and variance ratios.

5.2 Experiments on Real Data

Experiment 1 - Species Recognition We used the ETHEC dataset [11] which contains 47,978 butterfly images from six families and 561 species (example of the images are provided in Appendix D). We filtered out species with less than five images and focused on images of butterflies from the Lycaenidae and Nymphalidae families. In the training set, 10% of the species were from the Nymphalidae family, while at test time, 90% of the species were from the Nymphalidae family. For each class we sampled $2r = 20$ pairs.

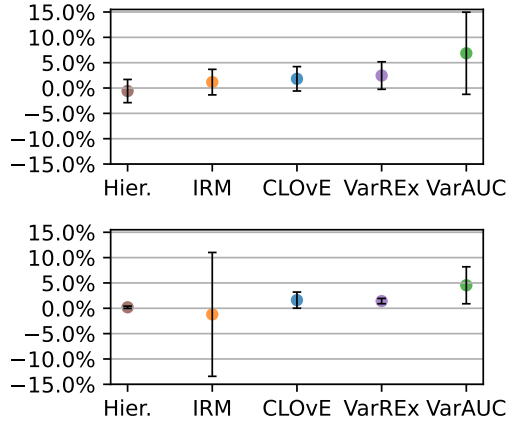

Figure 6: Average percentage changes of our method compared to ERM across 10 repetitions are shown for the ETHEC (top) and CelebA (bottom) datasets. Error bars represent $\pm$ one std-dev.

Experiment 2 - Face Recognition We used the *CelebA* dataset [30] which contains 202,599 images of 10,177 celebrities. We filtered out people for which the dataset contains less than three images. Following Vinyals et al. [51], we implemented g as a convolutional neural network which has four modules with 3×3 convolutions and 64 filters, followed by batch normalization, a ReLU activation, 2×2 max-pooling, and a fully connected layer of size 32. We used the attribute *blond hair* for the class distribution shift: for training, we mainly sampled people without blond hair (95%), while at test time, most people (95%)

had blond hair. Each training iteration had 150 synthetic environments of two classes and $2r = 20$ data points per class.

We trained the models on 200 synthetic environments at a time, each of two classes. We implemented g as a fully connected neural network with layers of sizes 128, 64, 32 and 16, and ReLU activations between them.

Experimental Results As can be seen in Figure 6, while all versions of our algorithm show some improvement over ERM, the best results are achieved with the VarAUC penalty (exact means and standard deviations are reported in Table 3 in Appendix D). One-sided paired t-tests show that the improvement over ERM achieved by our algorithm with the VarAUC penalty is statistically significant, with p-values of < 0.04 on both datasets; p-values for other penalties are reported in Table 4. All p-values were adjusted with FDR [4] correction.

In Appendix D we also provide additional analysis confirming that the main improvement of our algorithm over the ERM baseline stems from improved performance on negative minority pairs.

6 Discussion

In this study, we examined class distribution shifts in zero-shot learning, with a focus on shifts induced by unknown attributes. Such shifts pose significant challenges in zero-shot learning where new classes emerge in testing, causing standard techniques trained via ERM to fail on shifted class distributions, even when the conditional distribution of the data given class remains the same.

Previous research (see Appendix A) assumes closed-world classification or a known cause, making these methods unsuitable for zero-shot learning or shifts caused by unknown attributes. In response, we introduced a framework and the first algorithm to address class distribution shifts in zero-shot learning using OOD environment balancing methods.

In the causal terminology of closed-world OOD generalization, our framework employs synthetic environments to intervene on attribute mixtures by sampling small class subsets, thereby manipulating the class distribution. This facilitates the creation of diverse environments with varied attribute mixtures, enhancing the distinction between negative examples. A further comparison of our framework with OOD environment balancing methods is provided in Appendix C. Additionally, our proposed VarAUC penalty, designed for metric losses, enhances the separation of negative examples.

Our results demonstrate improvements compared to the ERM baseline on shifted distributions, without compromising performance on unshifted distributions, enabling the learning of more robust representations for zero-shot tasks and ensuring reliable performance.

While the proposed framework is general, our current experiments address shifts in a binary attribute. We defer exploration of additional scenarios, such as those involving shifts in multiple correlated attributes, to future work. An additional promising direction for future work is the consideration of shifts where the responsible attribute is strongly correlated with additional attributes or covariates. This opens up possibilities to explore structured constructions of synthetic environments that leverage such correlations.

References

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

[2] Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. Openface 2.0: Facial behavior analysis toolkit. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 59–66. IEEE, 2018.

[3] Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

[4] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

[5] Lacey Best-Rowden and Anil K Jain. Longitudinal study of automatic face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(1):148–162, 2017.

[6] Jonathon Byrd and Zachary Lipton. What is the effect of importance weighting in deep learning? In *International conference on machine learning*, pages 872–881. PMLR, 2019.

[7] Toon Calders and Szymon Jaroszewicz. Efficient auc optimization for classification. In *European conference on principles of data mining and knowledge discovery*, pages 42–53. Springer, 2007.

[8] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019.

[9] Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. Invariant rationalization. In *International Conference on Machine Learning*, pages 1448–1458. PMLR, 2020.

[10] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019.

[11] Ankit Dhall. Eth entomological collection (ethec) dataset [palearctic macrolepidoptera, spring 2019]. 2019.

[12] John C Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021.

[13] John C Duchi, Peter W Glynn, and Hongseok Namkoong. Statistics of robust optimization: A generalized empirical likelihood approach. *Mathematics of Operations Research*, 46(3): 946–969, 2021.

[14] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006.

[15] Patrick Grother, Mei Ngan, Kayee Hanaoka, et al. Ongoing face recognition vendor test (frvt) part 3: Demographic effects. *Nat. Inst. Stand. Technol., Gaithersburg, MA, USA, Rep. NISTIR*, 8280, 2019.

[16] Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. Context-aware feature generation for zero-shot semantic segmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1921–1929, 2020.

[17] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[18] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[19] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[20] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. In *International workshop on similarity-based pattern recognition*, pages 84–92. Springer, 2015.

[21] Badr Youbi Idrissi, Martin Arjovsky, Mohammad Pezeshki, and David Lopez-Paz. Simple data balancing achieves competitive worst-group-accuracy. In *Conference on Causal Learning and Reasoning*, pages 336–351. PMLR, 2022.

[22] Aparna R Joshi, Xavier Suau Cuadros, Nivedha Sivakumar, Luca Zappella, and Nicholas Apostoloff. Fair sa: Sensitivity analysis for fairness in face recognition. In *Algorithmic fairness through the lens of causality and robustness workshop*, pages 40–58. PMLR, 2022.

[23] Brendan F Klare, Mark J Burge, Joshua C Klontz, Richard W Vorder Bruegge, and Anil K Jain. Face recognition performance: Role of demographic information. *IEEE Transactions on information forensics and security*, 7(6):1789–1801, 2012.

[24] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. Siamese neural networks for one-shot image recognition. In *International Conference on Machine Learning (ICML) deep learning workshop*, volume 2, 2015.

[25] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pages 5815–5826. PMLR, 2021.

[26] Aviral Kumar, Sunita Sarawagi, and Ujjwal Jain. Trainable calibration measures for neural networks from kernel mean embeddings. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2805–2814. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/kumar18a.html`.

[27] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *AAAI*, volume 1, page 3, 2008.

[28] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[29] Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*, pages 6781–6792. PMLR, 2021.

[30] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[31] Arakaparampil M Mathai and Serge B Provost. Quadratic forms in random variables: theory and applications. *(No Title)*, 1992.

[32] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *International Conference on Learning Representations*, 2020.

[33] Dana Michalski, Sau Yee Yiu, and Chris Malec. The impact of age and threshold variation on facial recognition algorithm performance using images of children. In *2018 International Conference on Biometrics (ICB)*, pages 217–224. IEEE, 2018.

[34] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pages 681–699. Springer, 2020.

[35] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4004–4012, 2016.

[36] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(5):947–1012, 2016.

[37] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

[38] Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. Focus on the common good: Group distributional robustness follows. In *International Conference on Learning Representations*, 2021.

[39] Inioluwa Deborah Raji and Joy Buolamwini. Actionable auditing: Investigating the impact of publicly naming biased performance results of commercial ai products. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 429–435, 2019.

[40] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in neural information processing systems*, 33:4175–4186, 2020.

[41] Joseph P Robinson, Gennady Livitz, Yann Henon, Can Qin, Yun Fu, and Samson Timoner. Face recognition: too bias, or not too bias? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–1, 2020.

[42] Mateo Rojas-Carulla, Bernhard Schölkopf, Richard Turner, and Jonas Peters. Invariant models for causal transfer learning. *The Journal of Machine Learning Research*, 19(1):1309–1342, 2018.

[43] Karsten Roth, Timo Milbich, Samarth Sinha, Prateek Gupta, Bjorn Ommer, and Joseph Paul Cohen. Revisiting training strategies and generalization performance in deep metric learning. In *International Conference on Machine Learning*, pages 8242–8252. PMLR, 2020.

[44] Marco Saerens, Patrice Latinne, and Christine Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure. *Neural computation*, 14(1):21–41, 2002.

[45] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2019.

[46] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

[47] Tomáš Sixta, Julio CS Jacques Junior, Pau Buch-Cardona, Eduard Vazquez, and Sergio Escalera. Fairface challenge at eccv 2020: Analyzing bias in face recognition. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 463–481. Springer, 2020.

[48] Yuli Slavutsky and Yuval Benjamini. Predicting classification accuracy when adding new unobserved classes. In *International Conference on Learning Representations, ICLR, Conference Track Proceedings*, 2021.

[49] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29, 2016.

[50] Nisha Srinivas, Karl Ricanek, Dana Michalski, David S Bolme, and Michael King. Face recognition algorithm bias: Performance differences on images of children and adults. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019.

[51] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

[52] Yoav Wald, Amir Feder, Daniel Greenfeld, and Uri Shalit. On calibration and out-of-domain generalization. *Advances in neural information processing systems*, 34:2215–2227, 2021.

[53] Mei Wang and Weihong Deng. Mitigating bias in face recognition using skewness-aware reinforcement learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9322–9331, 2020.

[54] Mei Wang, Weihong Deng, Jiani Hu, Xunqiang Tao, and Yaohai Huang. Racial faces in the wild: Reducing racial bias by information maximization adaptation network. In *Proceedings of the ieee/cvf international conference on computer vision*, pages 692–702, 2019.

[55] Jiaheng Wei, Harikrishna Narasimhan, Ehsan Amid, Wen-Sheng Chu, Yang Liu, and Abhishek Kumar. Distributionally robust post-hoc classifiers under prior shifts. In *International Conference on Learning Representations (ICLR)*, 2023.

[56] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *Proceedings of the IEEE international conference on computer vision*, pages 2840–2848, 2017.

[57] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8256–8265, 2019.

[58] Tongtong Yuan, Weihong Deng, Jian Tang, Yinan Tang, and Binghui Chen. Signal-to-noise ratio: A robust distance metric for deep metric learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4815–4824, 2019.

[59] Charles Zheng, Rakesh Achanta, and Yuval Benjamini. Extrapolating expected accuracies for large multi-class problems. *The Journal of Machine Learning Research*, 19(1):2609–2638, 2018.

[60] Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16489–16498, 2021.

A Related Work on Class Distribution Shifts in Closed-World Settings

In *class-imbalanced learning* [28, 10, 8] it is assumed that some classes are more dominant in training, while in deployment this is no longer the case. Therefore, solutions classically include data or loss re-weighting [46, 6, 40, 32] and calibration of the classification score [44, 60]. A popular framework for addressing class distribut ion shifts is *distributionally robust optimization* (DRO) [3, 13, 12, 55], where instead of assuming a specific probability distribution, a set or range of possible distributions is considered, and optimization is performed to achieve the best results on the worst-case distribution. A special case known as *group DRO* [45, 38], involves a group variable that introduces discriminatory patterns among classes within specific groups. The framework to address this includes methods that assume that the classifier does not have access to the group information, and therefore propose re-weighting high loss examples [29], and data sub-sampling to balance classes and groups [21]. Nevertheless, the methods mentioned above rely on the training and test class sets being identical, making them unsuitable for direct application in zero-shot learning scenarios.

B Analysis of the Parametric Model

B.1 Derivation of the Loss

We begin by revisiting the parametric model introduced in §3. Let $z_i|c_i \sim \mathcal{N}(c_i, \Sigma_z)$, where $\Sigma_z = \nu_z I_d$, $0 < \nu_z \in \mathbb{R}$, and I_d is the d dimensional identity matrix. Classes c_i are drawn according to a Gaussian distribution $c_i \sim \mathcal{N}(0, \Sigma_a)$ corresponding to their type $a \in \{a_1, a_2\}$. Here, we use a simpler (although less intuitive) notation for the values of the diagonal matrices Σ_a:

$$\Sigma_{a_1} = \operatorname{diag}\left(\overbrace{\nu_0, \ldots, \nu_0}^{d_0}, \overbrace{\nu_1, \ldots, \nu_1}^{d_1}, \overbrace{\nu_2, \ldots, \nu_2}^{d_2}\right),$$
$$\Sigma_{a_2} = \operatorname{diag}\left(\nu_0, \ldots, \nu_0\,, \nu_2, \ldots, \nu_2, \nu_1, \ldots, \nu_1\right),$$

where $0 < \nu_2 < \nu_z < \nu_0 < \nu_1$.

We consider a weight representations $g(z) = Wz$, where W is a diagonal matrix with diagonal $w \in \mathbb{R}^d$.

Since Σ_z is of full rank, it suffices to consider the no-hinge version of the contrastive loss, that is

$$\widetilde{\ell}(z_i, z_j, y_{ij}; d_g) := y_{ij} \left\|W(z_i - z_j)\right\|^4 + (1 - y_{ij}) \left(m - \left\|W(z_i - z_j)\right\|^2\right)^2, \tag{12}$$

where $d_g(z_i, z_j) \coloneqq \left\|g(z_i - z_j)\right\|^2 = \left\|W(z_i - z_j)\right\|^2$ ($\|\cdot\|$ denotes the Euclidean norm[4]).

For a balanced sample of positive and negative examples, the expected loss is given by

$$\begin{aligned}\mathbb{E}\left[\widetilde{\ell}(z_i, z_j, y_{ij}; d_g)\right] = &\frac{1}{2}\mathbb{E}_{y_{ij}=1}\left[\left\|W(z_i - z_j)\right\|^4\right] \\ &+ \frac{1}{2}\mathbb{E}_{y_{ij}=0}\left[m^2 - 2m\left\|W(z_i - z_j)\right\|^2 + \left\|W(z_i - z_j)\right\|^4\right].\end{aligned} \tag{13}$$

To calculate the expression above, we begin by proving the following lemma:

Lemma 1. *Let $\mu \in \mathbb{R}^d$ be a random variable and let $t|\mu \sim \mathcal{N}(\mu, \Sigma)$. If $\mu \equiv 0$ (constant), then*

1. $\mathbb{E}\left\|t\right\|^4 = 2\operatorname{tr}(\Sigma^2) + \operatorname{tr}^2(\Sigma)$.

If $\mu \sim \mathcal{N}(0, \Sigma_\mu)$, then

2. $\mathbb{E}\left\|t\right\|^2 = \operatorname{tr}(\Sigma) + \operatorname{tr}(\Sigma_\mu)$,
3. $\mathbb{E}\left\|t\right\|^4 = 2\operatorname{tr}(\Sigma^2) + 4\operatorname{tr}(\Sigma\Sigma_\mu) + \operatorname{tr}^2(\Sigma) + 2\operatorname{tr}(\Sigma)\operatorname{tr}(\Sigma_\mu) + 2\operatorname{tr}(\Sigma_\mu^2) + \operatorname{tr}^2(\Sigma_\mu)$.

[4]Squared distance is selected for its simplicity in computing the expected value of even powers of the Euclidean norm of Gaussian variables.

Proof. For any random variable $u \in \mathbb{R}^d$, such that $u \sim \mathcal{N}(\mu_u, \Sigma_u)$, and any symmetric matrix A, we have

$$\mathbb{E}_u[u^T A u] = \operatorname{tr}(A\Sigma_u) + \mu_u^T A \mu_u, \tag{14}$$

$$\mathbb{E}_u[u^T A u]^2 = 2\operatorname{tr}\left((A\Sigma_u)^2\right) + 4\mu_u^T A \Sigma_u A \mu_u + \left(\operatorname{tr}(A\Sigma_u) + \mu_u^T A \mu_u\right)^2 \tag{15}$$

(see, for example, Thm. 3.2b.2 in [31]).

First, letting $\mu_u = 0$, $\Sigma_u = \Sigma$ and $A = I_d$ in (15) we get

$$\mathbb{E}\left\|t\right\|^4 = \mathbb{E}[t^T t]^2 = 2\operatorname{tr}(\Sigma^2) + \operatorname{tr}^2(\Sigma). \tag{16}$$

Now, assume that $\mu \sim \mathcal{N}(0, \Sigma_\mu)$. From (14) we get $\mathbb{E}_\mu \left\|\mu\right\|^2 = \mathbb{E}_\mu[\mu^T \mu] = \operatorname{tr}(\Sigma_\mu)$, and thus

$$\mathbb{E}\left\|t\right\|^2 = \mathbb{E}_\mu\left[\mathbb{E}_{t|\mu}[t^T t \mid \mu]\right] = \mathbb{E}_\mu\left[\operatorname{tr}(\Sigma) + \mu^T \mu\right] = \operatorname{tr}(\Sigma) + \operatorname{tr}(\Sigma_\mu). \tag{17}$$

Similarly, from (15) we have

$$\mathbb{E}\left\|t\right\|^4 = \mathbb{E}_\mu\left[\mathbb{E}_{t|\mu}[[t^T t]^2 \mid \mu]\right] = 2\operatorname{tr}(\Sigma^2) + 4\,\mathbb{E}_\mu[\mu^T \Sigma \mu] + \operatorname{tr}^2(\Sigma) + 2\operatorname{tr}(\Sigma)\,\mathbb{E}_\mu\left\|\mu\right\|^2 + \mathbb{E}_\mu\left\|\mu\right\|^4. \tag{18}$$

By substituting $A = \Sigma$ in (14) we get $\mathbb{E}_\mu[\mu^T \Sigma \mu] = \operatorname{tr}(\Sigma\Sigma_\mu)$, and from (15) we have

$$\mathbb{E}_\mu\left\|\mu\right\|^4 = 2\operatorname{tr}(\Sigma_\mu^2) + \operatorname{tr}^2(\Sigma_\mu). \tag{19}$$

Therefore,

$$\mathbb{E}\left\|t\right\|^4 = 2\operatorname{tr}(\Sigma^2) + 4\operatorname{tr}(\Sigma\Sigma_\mu) + \operatorname{tr}^2(\Sigma) + 2\operatorname{tr}(\Sigma)\operatorname{tr}(\Sigma_\mu) + 2\operatorname{tr}(\Sigma_\mu^2) + \operatorname{tr}^2(\Sigma_\mu). \tag{20}$$

□

Note that $W(z_i - z_j) \sim \mathcal{N}(\mu, \Sigma)$, with $\mu = W(c_i - c_j)$ and $\Sigma = 2\nu_z W^T W$.

If $y_{ij} = 1$, then z_i and z_j are from the same class, meaning that $c_i = c_j$ and thus $\mu = 0$. Therefore, by Lemma 1.(1) we have

$$\begin{aligned}
\mathbb{E}_{y_{ij}=1}\left\|W(z_i - z_j)\right\|^4 &= 2\operatorname{tr}\left(\Sigma^2\right) + \operatorname{tr}^2(\Sigma) \\
&= 2 \cdot 4\nu_z^2 \operatorname{tr}\left(\left[W^T W\right]^2\right) + 4\nu_z^2 \operatorname{tr}^2\left(W^T W\right) \\
&= 8\nu_z^2 \sum_{i=1}^{d} w_i^4 + 4\nu_z^2 \left(\sum_{i=1}^{d} w_i^2\right)^2.
\end{aligned} \tag{21}$$

However, for pairs from different classes, that is, when $y_{ij} = 0$, the mean μ is itself a Gaussian random variable distributed according to $\mathcal{N}(0, \Sigma_\mu)$, where

$$\Sigma_\mu = \begin{cases} W^T(2\Sigma_{a_1})W & c_i, c_j \text{ are both of type } a_1 \\ W^T(2\Sigma_{a_2})W & c_i, c_j \text{ are both of type } a_2 \\ W^T(\Sigma_{a_1} + \Sigma_{a_2})W & c_i, c_j \text{ are of different types .} \end{cases} \tag{22}$$

Therefore, by Lemma 1.(2) we have

$$\begin{aligned}
\mathbb{E}_{y_{ij}=0}\left\|W(z_i - z_j)\right\|^2 &= \mathbb{E}_{y_{ij}=0}\left[\operatorname{tr}(\Sigma_\mu) + \operatorname{tr}(\Sigma)\right] = \mathbb{E}_{y_{ij}=0}\left[\operatorname{tr}(\Sigma_\mu)\right] + \operatorname{tr}(\Sigma) \\
&= \rho^2 \operatorname{tr}\left(2W^T \Sigma_{a_1} W\right) + (1-\rho)^2 \operatorname{tr}\left(2W^T \Sigma_{a_2} W\right) \\
&\quad + 2\rho(1-\rho)\operatorname{tr}\left(W^T(\Sigma_{a_1} + \Sigma_{a_2})W\right) + \operatorname{tr}(\Sigma) \\
&= 2\left[(\nu_0 + \nu_z)\sum_{i=1}^{d_0} w_i^2 + (\alpha_1 + \nu_z)\sum_{i=d_0+1}^{d_0+d_1} w_i^2 + (\alpha_2 + \nu_z)\sum_{i=d_0+d_1+1}^{d} w_i^2\right],
\end{aligned} \tag{23}$$

where

$$\begin{aligned}
\alpha_1 &\coloneqq \rho^2\nu_1 + (1-\rho)^2\nu_2 + \rho(1-\rho)\left(\nu_1+\nu_2\right) = \rho\nu_1 + (1-\rho)\,\nu_2,\\
\alpha_2 &\coloneqq \rho^2\nu_2 + (1-\rho)^2\nu_1 + \rho(1-\rho)\left(\nu_1+\nu_2\right) = \rho\nu_2 + (1-\rho)\,\nu_1,\\
\beta_1 &\coloneqq 2\rho^2\nu_1^2 + 2(1-\rho)^2\nu_2^2 + \rho(1-\rho)\left(\nu_1+\nu_2\right)^2,\\
\beta_2 &\coloneqq 2\rho^2\nu_2^2 + 2(1-\rho)^2\nu_1^2 + \rho(1-\rho)\left(\nu_1+\nu_2\right)^2.
\end{aligned} \tag{24}$$

By Lemma 1.(3) we have

$$\begin{aligned}
\mathbb{E}_{y_{ij}=0}\left\|w\left(z_i - z_j\right)\right\|^4 =& \mathbb{E}_{y_{ij}=0}\Big[2\operatorname{tr}\left(\Sigma^2\right) + 4\operatorname{tr}\left(\Sigma\Sigma_\mu\right) + \operatorname{tr}^2(\Sigma) + 2\operatorname{tr}\left(\Sigma\right)\operatorname{tr}\left(\Sigma_\mu\right)\\
&+2\operatorname{tr}\left(\Sigma_\mu^2\right) + \left(\operatorname{tr}\left(\Sigma_\mu\right)\right)^2\Big]\\
=&2\operatorname{tr}(\Sigma^2) + 4\,\mathbb{E}_{y_{ij}=0}[\operatorname{tr}(\Sigma\Sigma_\mu)] + \operatorname{tr}^2(\Sigma) + 2\operatorname{tr}(\Sigma)\,\mathbb{E}_{y_{ij}=0}[\operatorname{tr}(\Sigma_\mu)]\\
&+ 2\,\mathbb{E}_{y_{ij}=0}[\operatorname{tr}(\Sigma_\mu^2)] + \mathbb{E}_{y_{ij}=0}[\operatorname{tr}^2(\Sigma_\mu)],
\end{aligned} \tag{25}$$

where

$$\begin{aligned}
\mathbb{E}_{y_{ij}=0}\left[\operatorname{tr}\left(\Sigma\Sigma_\mu\right)\right] =& 2\nu_z\operatorname{tr}\Big(W^TW\left[2\rho^2W^T\Sigma_{a_1}W + 2(1-\rho)^2W^T\Sigma_{a_2}W\right.\\
&\left.+\ 2\rho(1-\rho)W^T\left(\Sigma_{a_1}+\Sigma_{a_2}\right)W\right]\Big)\\
=&4\nu_z\left[\nu_0\sum_{i=1}^{d_0}w_i^4 + \alpha_1\sum_{i=d_0+1}^{d_0+d_1}w_i^4 + \alpha_2\sum_{i=d_0+d_1+1}^{d}w_i^4\right];
\end{aligned} \tag{26}$$

$$\begin{aligned}
\mathbb{E}_{y_{ij}=0}\left[\operatorname{tr}\left(\Sigma_\mu\right)\right] =& \rho^2\operatorname{tr}\left(2W^T\Sigma_{a_1}W\right) + (1-\rho)^2\operatorname{tr}\left(2W^T\Sigma_{a_1}W\right)\\
&+ 2\rho(1-\rho)\operatorname{tr}\left(W^T\left(\Sigma_{a_1}+\Sigma_{a_2}\right)W\right)\\
=&2\left[\nu_0\sum_{i=1}^{d_0}w_i^2 + \alpha_1\sum_{i=d_0+1}^{d_0+d_1}w_i^2 + \alpha_2\sum_{i=d_0+d_1+1}^{d}w_i^2\right], r
\end{aligned} \tag{27}$$

and so

$$\operatorname{tr}\left(\Sigma\right)\mathbb{E}_{y_{ij}=0}\left[\operatorname{tr}\left(\Sigma_\mu\right)\right] = 4\nu_z\left(\sum_{i=1}^{d}w_i^2\right)\left[\nu_0\sum_{i=1}^{d_0}w_i^2 + \alpha_1\sum_{i=d_0+1}^{d_0+d_1}w_i^2 + \alpha_2\sum_{i=d_0+d_1+1}^{d}w_i^2\right]; \tag{28}$$

$$\begin{aligned}
\mathbb{E}_{y_{ij}=0}\left[\operatorname{tr}\left(\Sigma_\mu^2\right)\right] =& \rho^2\operatorname{tr}\left((2W^T\Sigma_{a_1}W)^2\right) + (1-\rho)^2\operatorname{tr}\left((2W^T\Sigma_{a_2}W)^2\right)\\
&+ 2\rho(1-\rho)\operatorname{tr}\left((W^T\left(\Sigma_{a_1}+\Sigma_{a_2}\right)W)^2\right)\\
=&2\left[2\nu_0^2\sum_{i=1}^{d_0}w_i^4 + \beta_1\sum_{i=d_0+1}^{d_0+d_1}w_i^4 + \beta_2\sum_{i=d_0+d_1+1}^{d}w_i^4\right];
\end{aligned} \tag{29}$$

and similarly

$$\begin{aligned}
\mathbb{E}_{y_{ij}=0}\left[\operatorname{tr}^2\left(\Sigma_\mu\right)\right] =& 2\Bigg[2\nu_0^2\left(\sum_{i=1}^{d_0}w_i^2\right)^2 + \beta_1\left(\sum_{i=d_0+1}^{d_0+d_1}w_i^2\right)^2 + \beta_2\left(\sum_{i=d_0+d_1+1}^{d}w_i^2\right)^2\\
&+ 4\gamma_{0,1}\sum_{i=1}^{d_0}w_i^2\sum_{i=d_0+1}^{d_0+d_1}w_i^2 + 4\gamma_{0,2}\sum_{i=1}^{d_0}w_i^2\sum_{i=d_0+d_1+1}^{d}w_i^2\\
&+ 4\gamma_{1,2}\sum_{i=d_0+1}^{d_0+d_1}w_i^2\sum_{i=d_0+d_1+1}^{d}w_i^2\Bigg],
\end{aligned} \tag{30}$$

where we denote for short

$$\begin{aligned}
\gamma_{0,1} &\coloneqq \rho^2\nu_0\nu_1 + (1-\rho)^2\nu_0\nu_2 + \rho(1-\rho)\nu_0\left(\nu_1+\nu_2\right), \\
\gamma_{0,2} &\coloneqq \rho^2\nu_0\nu_2 + (1-\rho)^2\nu_0\nu_1 + \rho(1-\rho)\nu_0\left(\nu_1+\nu_2\right), \\
\gamma_{1,2} &\coloneqq \rho^2\nu_1\nu_2 + (1-\rho)^2\nu_1\nu_2 + \frac{1}{2}\rho(1-\rho)\left(\nu_1+\nu_2\right)^2.
\end{aligned} \tag{31}$$

Finally, due to symmetry, at the optimal solution we have

$$w_i = \begin{cases} u_0 & 0 \le i \le d_0 \\ u_1 & d_0+1 \le i \le d_0+d_1 \\ u_2 & d_0+d_1+1 \le i \le d, \end{cases} \tag{32}$$

and by combining these results, we get

$$\begin{aligned}
\mathbb{E}\left[\widetilde{\ell}\left(z_i, z_j, y_{ij}; d_g\right)\right] &= d_0u_0^4\left(8\nu_z^2 + 8\nu_z\nu_0 + 4\nu_0^2 + 2\nu_0^2 d_0\right) \\
&+ d_1u_1^4\left(8\nu_z^2 + 8\nu_z\alpha_1 + 2\beta_1 + \beta_1 d_1\right) \\
&+ d_2u_2^4\left(8\nu_z^2 + 8\nu_z\alpha_2 + 2\beta_2 + \beta_2 d_2\right) \\
&- 2d_0u_0^2\left(\nu_0+\nu_z\right) - 2d_1u_1^2\left(\alpha_1+\nu_z\right) - 2d_2u_2^2\left(\alpha_2+\nu_z\right) \\
&+ 4\nu_z^2\left(d_0u_0^2 + d_1u_1^2 + d_2u_2^2\right)^2 + \frac{1}{2}m \\
&+ 4\nu_z\left(d_0u_0^2 + d_1u_1^2 + d_2u_2^2\right)\left[\nu_0 d_0u_0^2 + \alpha_1 d_1u_1^2 + \alpha_2 d_2u_2^2\right] \\
&+ 4\gamma_{0,1}d_0d_1u_0^2u_1^2 + 4\gamma_{0,2}d_0d_2u_0^2u_2^2 + 4\gamma_{1,2}d_1d_2u_1^2u_2^2.
\end{aligned} \tag{33}$$

B.2 Analysis of the Optimal Solution (Proof of Proposition 1)

Proposition 1 shows that when d_1 and d_2 are relatively similar, the optimal solution on the training distribution, assigns more weight to components with high variance in the training data than to those with high variance in the shifted test distribution.

We begin by defining the required condition on d_1 and d_2. Denote

$$\begin{aligned}
\psi_1 &\coloneqq 2\nu_z^2 + 2\nu_z\alpha_1 + \beta_1 \\
\psi_2 &\coloneqq 2\nu_z^2 + 2\nu_z\alpha_2 + \beta_2 \\
\eta_{01} &\coloneqq 4\nu_z^2 + 2\nu_z\left(\alpha_1+\nu_0\right) + 2\gamma_{0,1} \\
\eta_{02} &\coloneqq 4\nu_z^2 + 2\nu_z\left(\alpha_2+\nu_0\right) + 2\gamma_{0,2} \\
\eta_{12} &\coloneqq 4\nu_z^2 + 2\nu_z\left(\alpha_1+\alpha_2\right) + 2\gamma_{1,2}.
\end{aligned} \tag{34}$$

Then for α, β, γ values as in equations 24 and 31, we define

$$h_l\left(\rho, \nu_z, \nu_0, \nu_1, \nu_2\right) \coloneqq \frac{\psi_1}{\psi_2}\frac{\left(\alpha_2+\nu_z\right)}{\left(\alpha_1+\nu_z\right)}, \quad h_u\left(\rho, \nu_z, \nu_0, \nu_1, \nu_2\right) \coloneqq \frac{\psi_1}{\psi_2}\frac{\eta_{02}}{\eta_{01}}, \tag{35}$$

and the corresponding condition

$$h_l\left(\rho, \nu_z, \nu_0, \nu_1, \nu_2\right) < \frac{d_2+2}{d_1+2} < h_u\left(\rho, \nu_z, \nu_0, \nu_1, \nu_2\right). \tag{36}$$

While this condition is sufficient, it is not necessary.Values of $\rho, \nu_z, \nu_0, \nu_1, \nu_2$ and d_1, d_2 that satisfy 35 provide an example requiring only a simple analysis, without a full characterization of the optimal solution, for the failure of optimization over the training distribution. However, such failures can occur for additional parameter values, and the full characterization is provided in Appendix B.3.

Proof. Without loss of generality assume m=1. Then,

$$\begin{aligned}\frac{\partial \mathbb{E}\left[\widetilde{\ell}\left(z_i, z_j, y_{ij}; d_g\right)\right]}{\partial u_0^2} &= 2d_0u_0^2\left(8\nu_z^2 + 8\nu_z\nu_0 + 4\nu_0^2 + 2\nu_0^2 d_0\right) \\ &\quad - 2d_0\left(\nu_0 + \nu_z\right) \\ &\quad + 8d_0\nu_z^2\left(d_0u_0^2 + d_1u_1^2 + d_2u_2^2\right) \\ &\quad + 4d_0\nu_z\left(\nu_0 d_0u_0^2 + \alpha_1 d_1u_1^2 + \alpha_2 d_2u_2^2\right) \\ &\quad + 4d_0\nu_z\nu_0\left(d_0u_0^2 + d_1u_1^2 + d_2u_2^2\right) \\ &\quad + 4\gamma_{0,1}d_0d_1u_1^2 + 4\gamma_{0,2}d_0d_2u_2^2 \end{aligned} \tag{37}$$

and by setting the partial derivative to zero we get

$$\begin{aligned}2u_0^2\left(2+d_0\right)\left(2\nu_z^2 + 2\nu_z\nu_0 + \nu_0^2\right) =& 2d_1u_1^2\left(2\nu_z^2 + \nu_z\left(\alpha_1 + \nu_0\right) + \gamma_{0,1}\right) \\ &+2d_2u_2^2\left(2\nu_z^2 + \nu_z\left(\alpha_2 + \nu_0\right) + \gamma_{0,2}\right) - \left(\nu_0 + \nu_z\right). \end{aligned} \tag{38}$$

Therefore,

$$u_0^2 = \frac{\nu_0 + \nu_z - \eta_{01}d_1u_1^2 - \eta_{02}d_2u_2^2}{2\left(2+d_0\right)\left(2\nu_z^2 + 2\nu_z\nu_0 + \nu_0^2\right)}. \tag{39}$$

and similarly

$$u_1^2 = \frac{\left(\alpha_1 + \nu_z\right) - \eta_{01}d_0u_0^2 - \eta_{12}d_2u_2^2}{2\left(2+d_1\right)\left(2\nu_z^2 + 2\nu_z\alpha_1 + \beta_1\right)} \tag{40}$$

$$u_2^2 = \frac{\left(\alpha_2 + \nu_z\right) - \eta_{02}d_0u_0^2 - \eta_{12}d_1u_1^2}{2\left(2+d_2\right)\left(2\nu_z^2 + 2\nu_z\alpha_2 + \beta_2\right)}. \tag{41}$$

Hence,

$$\begin{aligned}d_1u_1^2 - d_2u_2^2 =& \frac{\left(2+d_2\right)\left(2\nu_z^2 + 2\nu_z\alpha_2 + \beta_2\right)\left[d_1\left(\alpha_1 + \nu_z\right) - \eta_{01}d_1d_0u_0^2 - \eta_{12}d_1d_2u_2^2\right]}{2\left(2+d_1\right)\left(2+d_2\right)\left(2\nu_z^2 + 2\nu_z\alpha_1 + \beta_1\right)\left(2\nu_z^2 + 2\nu_z\alpha_2 + \beta_2\right)} \\ &- \frac{\left(2+d_1\right)\left(2\nu_z^2 + 2\nu_z\alpha_1 + \beta_1\right)\left[d_2\left(\alpha_2 + \nu_z\right) - \eta_{02}d_2d_0u_0^2 - \eta_{12}d_1d_2u_1^2\right]}{2\left(2+d_1\right)\left(2+d_2\right)\left(2\nu_z^2 + 2\nu_z\alpha_1 + \beta_1\right)\left(2\nu_z^2 + 2\nu_z\alpha_2 + \beta_2\right)}. \end{aligned} \tag{42}$$

Denoting

$$\begin{aligned}\xi &\coloneqq 2\left(2+d_1\right)\left(2+d_2\right)\left(2\nu_z^2 + 2\nu_z\alpha_1 + \beta_1\right)\left(2\nu_z^2 + 2\nu_z\alpha_2 + \beta_2\right) \\ &= 2\left(2+d_1\right)\left(2+d_2\right)\psi_1\psi_2 \end{aligned}$$

we have

$$\begin{aligned}d_1u_1^2\left[1 - \frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}\right] =& d_2u_2^2\left[1 - \frac{1}{\xi}\left(2+d_2\right)\psi_2\eta_{12}\right] \\ &+\frac{1}{\xi}\left(2+d_2\right)\psi_2\left(\alpha_1 + \nu_z\right) - \frac{1}{\xi}\left(2+d_1\right)\psi_1\left(\alpha_2 + \nu_z\right) \\ &+d_0u_0^2\left[\frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{02} - \frac{1}{\xi}\left(2+d_2\right)\psi_2\eta_{01}\right] \end{aligned}$$

and therefore

$$\begin{aligned}d_1u_1^2 - d_2u_2^2 =& d_2u_2^2\left(\frac{1 - \frac{1}{\xi}\left(2+d_2\right)\psi_2\eta_{12}}{1 - \frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}} - 1\right) \\ &+\frac{1}{2\left(2+d_1\right)\left(2+d_2\right)\psi_1\psi_2}\frac{\left(2+d_2\right)\psi_2\left(\alpha_1 + \nu_z\right) - \left(2+d_1\right)\psi_1\left(\alpha_2 + \nu_z\right)}{1 - \frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}} \\ &+d_0u_0^2\frac{1}{2\left(2+d_1\right)\left(2+d_2\right)\psi_1\psi_2}\left[\frac{\left(2+d_1\right)\psi_1\eta_{02}}{1 - \frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}} - \frac{\left(2+d_2\right)\psi_2\eta_{01}}{1 - \frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}}\right]. \end{aligned} \tag{43}$$

Denote

$$\begin{aligned}\Delta =&\left(2+d_1\right)\left[d_2u_2^2\eta_{12}\psi_1-\left(\alpha_2+\nu_z\right)\psi_1+d_0u_0^2\psi_1\eta_{02}\right]\\&-\left(2+d_2\right)\left[d_2u_2^2\eta_{12}\psi_2-\left(\alpha_1+\nu_z\right)\psi_2+d_0u_0^2\psi_2\eta_{01}\right],\end{aligned}\tag{44}$$

and thus

$$d_1u_1^2-d_2u_2^2=\frac{1}{2\left(2+d_1\right)\left(2+d_2\right)\psi_1\psi_2}\frac{1}{1-\frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}}\Delta.\tag{45}$$

Note that for d_1, d_2 such that

$$\begin{cases}\left(2+d_1\right)\psi_1-\left(2+d_2\right)\psi_2>0 & \Rightarrow \frac{d_2+2}{d_1+2}<\frac{\psi_1}{\psi_2}\\ \left(2+d_2\right)\left(\alpha_1+\nu_z\right)\psi_2-\left(2+d_1\right)\left(\alpha_2+\nu_z\right)\psi_1>0 & \Rightarrow \frac{d_2+2}{d_1+2}>\frac{\psi_1}{\psi_2}\frac{\left(\alpha_2+\nu_z\right)}{\left(\alpha_1+\nu_z\right)}\\ \left(2+d_1\right)\psi_1\eta_{02}-\left(2+d_2\right)\psi_2\eta_{01}>0 & \Rightarrow \frac{d_2+2}{d_1+2}<\frac{\psi_1}{\psi_2}\frac{\eta_{02}}{\eta_{01}}\end{cases}$$

we have $\Delta>0$. Since $\frac{\eta_{02}}{\eta_{01}}<1$, this reduces to the last two conditions and therefore, in particular for

$$\frac{\psi_1}{\psi_2}\frac{\left(\alpha_2+\nu_z\right)}{\left(\alpha_1+\nu_z\right)}<\frac{d_2+2}{d_1+2}<\frac{\psi_1}{\psi_2}\frac{\eta_{02}}{\eta_{01}}\tag{46}$$

we have $\Delta>0$. Additionally, note that

$$1-\frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}=1-\frac{\eta_{12}}{2\left(2+d_1\right)\left(2+d_2\right)\psi_1\psi_2}\left(2+d_1\right)\psi_1=\frac{2\left(2+d_2\right)\psi_2-\eta_{12}}{2\left(2+d_2\right)\psi_2}\tag{47}$$

and thus $1-\frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}>0$ iff

$$d_2+2>\frac{1}{2}\frac{\eta_{12}}{\psi_2}.\tag{48}$$

Combining these conditions reduces to

$$\frac{\psi_1}{\psi_2}\frac{\left(\alpha_2+\nu_z\right)}{\left(\alpha_1+\nu_z\right)}<\frac{d_2+2}{d_1+2}<\frac{\psi_1}{\psi_2}\frac{\eta_{02}}{\eta_{01}},\tag{49}$$

and therefore, for $\nu_z, \nu_0, \nu_1, \nu_2, d_1, d_2$ satisfying

$$\frac{\psi_1}{\psi_2}\frac{\left(\alpha_2+\nu_z\right)}{\left(\alpha_1+\nu_z\right)}<\frac{d_2+2}{d_1+2}<\frac{\psi_1}{\psi_2}\frac{\eta_{02}}{\eta_{01}}.\tag{50}$$

we have $d_1u_1^2-d_2u_2^2>0$.[5] □

[5]Similarly, the condition obtained for $1-\frac{1}{\xi}\left(2+d_1\right)\psi_1\eta_{12}<0$ and $\Delta<0$ is $\frac{\psi_1}{\psi_2}\left(2+d_1\right)<2+d_2<\frac{\psi_1}{\psi_2}\frac{\left(\alpha_2+\nu_z\right)}{\left(\alpha_1+\nu_z\right)}\left(2+d_1\right)$, which cannot be achieved since $\alpha_2<\alpha_1$.

B.3 Explicit Expression for the Optimal Representation

In order to derive the optimal representation, we differentiate the expected loss with respect to the squared values in the diagonal of W, that is, w_i^2:

$$\frac{\partial}{\partial\left(w_i^2\right)}\operatorname{tr}\left(\Sigma^2\right)=8\nu_z^2 w_i^2 \tag{51}$$

$$\frac{\partial}{\partial\left(w_i^2\right)}\operatorname{tr}^2\left(\Sigma\right)=8\nu_z^2\sum_{j=1}^{d}w_j^2 \tag{52}$$

$$\frac{\partial}{\partial\left(w_i^2\right)}\mathbb{E}_{y=0}\left[\operatorname{tr}\left(\Sigma\Sigma_\mu\right)\right]=\begin{cases}8\nu_z\nu_0 w_i^2, & 1\le i\le d_0\\ 8\nu_z\alpha_1 w_i^2, & d_0+1\le i\le d_0+d_1\\ 8\nu_z\alpha_2 w_i^2, & d_0+d_1+1\le i\le d\end{cases} \tag{53}$$

$$\frac{\partial\left[\operatorname{tr}(\Sigma)\,\mathbb{E}_{y=0}[\operatorname{tr}\Sigma_\mu]\right]}{\partial\left(w_i^2\right)}= \tag{54}$$

$$\begin{cases}4\nu_z\left[2\nu_0\sum\limits_{j=1}^{d_0}w_j^2+(\alpha_1+\nu_0)\sum\limits_{j=d_0+1}^{d_0+d_1}w_j^2+(\alpha_2+\nu_0)\sum\limits_{j=d_0+d_1+1}^{d}w_j^2\right] & 1\le i\le d_0\\ 4\nu_z\left[(\nu_0+\alpha_1)\sum\limits_{j=1}^{d_0}w_j^2+2\alpha_1\sum\limits_{j=d_0+1}^{d_0+d_1}w_j^2+(\alpha_2+\alpha_1)\sum\limits_{j=d_0+d_1+1}^{d}w_j^2\right] & d_0+1\le i\le d_0+d_1\\ 4\nu_z\left[(\nu_0+\alpha_2)\sum\limits_{j=1}^{d_0}w_j^2+(\alpha_1+\alpha_2)\sum\limits_{j=d_0+1}^{d_0+d_1}w_j^2+2\alpha_2\sum\limits_{j=d_0+d_1+1}^{d}w_j^2\right] & d_0+d_1+1\le i\le d\end{cases} \tag{55}$$

$$\frac{\partial}{\partial\left(w_i^2\right)}\mathbb{E}_{y=0}\left[\operatorname{tr}\left(\Sigma_\mu^2\right)\right]=\begin{cases}8\nu_0^2 w_i^2 & 1\le i\le d_0\\ 4\beta_1 w_i^2 & d_0+1\le i\le d_0+d_1\\ 4\beta_2 w_i^2 & d_0+d_1+1\le i\le d\end{cases} \tag{56}$$

$$\frac{\partial}{\partial\left(w_i^2\right)}\mathbb{E}_{y=0}\left[\operatorname{tr}^2\left(\Sigma_\mu\right)\right]= \tag{57}$$

$$\begin{cases}8\nu_0^2\sum\limits_{j=1}^{d_0}w_j^2+8\gamma_{0,1}\sum\limits_{j=d_0+1}^{d_0+d_1}w_j^2+8\gamma_{0,2}\sum\limits_{j=d_0+d_1+1}^{d}w_j^2 & 1\le i\le d_0\\ 8\gamma_{0,1}\sum\limits_{j=1}^{d_0}w_j^2+4\beta_1\sum\limits_{j=d_0+1}^{d_0+d_1}w_j^2+8\gamma_{1,2}\sum\limits_{j=d_0+d_1+1}^{d}w_j^2 & d_0+1\le i\le d_0+d_1\\ 8\gamma_{0,2}\sum\limits_{j=1}^{d_0}w_j^2+8\gamma_{1,2}\sum\limits_{j=d_0+1}^{d_0+d_1}w_j^2+4\beta_2\sum\limits_{j=d_0+d_1+1}^{d}w_j^2 & d_0+d_1+1\le i\le d\end{cases} \tag{58}$$

Combining these results, we get for $1\le i\le d_0$

$$\begin{aligned}\partial_0 \coloneqq \frac{\partial}{\partial\left(w_i^2\right)}\widetilde{\ell}\left(z_i,z_j,y_{ij};d_g\right)=\frac{1}{2}&\left[2\cdot 8\nu_z^2 w_i^2+8\nu_z^2\sum_{j=1}^{d}w_j^2\right]-m\left[2\left(\nu_0+\nu_z\right)\right]\\ &+8\nu_z^2 w_i^2+4\nu_z^2\sum_{j=1}^{d}w_j^2+2\cdot 8\nu_z\nu_0 w_i^2\\ &+4\nu_z\left[2\nu_0\sum_{j=1}^{d_0}w_j^2+(\alpha_1+\nu_0)\sum_{j=d_0+1}^{d_0+d_1}w_j^2+(\alpha_2+\nu_0)\sum_{j=d_0+d_1+1}^{d}w_j^2\right]\\ &+8\nu_0^2 w_i^2+\frac{1}{2}\left[8\nu_0^2\sum_{j=1}^{d_0}w_j^2+8\gamma_{0,1}\sum_{j=d_0+1}^{d_0+d_1}w_j^2+8\gamma_{0,2}\sum_{j=d_0+d_1+1}^{d}w_j^2\right],\end{aligned}$$

for $d_0+1 \leq i \leq d_0+d_1$

$$\begin{aligned}\partial_1 \coloneqq \frac{\partial}{\partial\left(w_i^2\right)}\widetilde{\ell}\left(z_i,z_j,y_{ij};d_g\right) = \frac{1}{2}\left[2\cdot 8\nu_z^2 w_i^2 + 8\nu_z^2\sum_{j=1}^{d} w_j^2\right] - m\left[2\left(\alpha_1+\nu_z\right)\right] \\ + 8\nu_z^2 w_i^2 + 4\nu_z^2\sum_{j=1}^{d} w_j^2 + 2\cdot 8\nu_z\alpha_1 w_i^2 \\ + 4\nu_z\left[\left(\nu_0+\alpha_1\right)\sum_{j=1}^{d_0} w_j^2 + 2\alpha_1\sum_{j=d_0+1}^{d_0+d_1} w_j^2 + \left(\alpha_2+\alpha_1\right)\sum_{j=d_0+d_1+1}^{d} w_j^2\right] \\ + 4\beta_1 w_i^2 + \frac{1}{2}\left[8\gamma_{0,1}\sum_{j=1}^{d_0} w_j^2 + 4\beta_1\sum_{j=d_0+1}^{d_0+d_1} w_j^2 + 8\gamma_{1,2}\sum_{j=d_0+d_1+1}^{d} w_j^2\right],\end{aligned}$$

and similarly for $d_0+d_1+1 \leq i \leq d$

$$\begin{aligned}\partial_2 \coloneqq \frac{\partial}{\partial\left(w_i^2\right)}\widetilde{\ell}\left(z_i,z_j,y_{ij};d_g\right) = \frac{1}{2}\left[2\cdot 8\nu_z^2 w_i^2 + 8\nu_z^2\sum_{j=1}^{d} w_j^2\right] - m\left[2\left(\alpha_2+\nu_z\right)\right] \\ + 8\nu_z^2 w_i^2 + 4\nu_z^2\sum_{j=1}^{d} w_j^2 + 2\cdot 8\nu_z\alpha_2 w_i^2 \\ + 4\nu_z\left[\left(\nu_0+\alpha_2\right)\sum_{j=1}^{d_0} w_j^2 + \left(\alpha_1+\alpha_2\right)\sum_{j=d_0+1}^{d_0+d_1} w_j^2 + 2\alpha_2\sum_{j=d_0+d_1+1}^{d} w_j^2\right] \\ + 4\beta_2 w_i^2 + \frac{1}{2}\left[8\gamma_{0,2}\sum_{j=1}^{d_0} w_j^2 + 8\gamma_{1,2}\sum_{j=d_0+1}^{d_0+d_1} w_j^2 + 4\beta_2\sum_{j=d_0+d_1+1}^{d} w_j^2\right].\end{aligned}$$

Thus, we can write for the symmetric solution

$$\partial_0 = -2m\left(\nu_0+\nu_z\right) + u_0^2 G_{0,0} + u_1^2 G_{0,1} + u_2^2 G_{0,2}, \tag{59}$$

$$\partial_1 = -2m\left(\alpha_1+\nu_z\right) + u_0^2 G_{1,0} + u_1^2 G_{1,1} + u_2^2 G_{1,2}, \tag{60}$$

$$\partial_2 = -2m\left(\alpha_2+\nu_z\right) + u_0^2 G_{2,0} + u_1^2 G_{2,1} + u_2^2 G_{2,2}, \tag{61}$$

where

$$\begin{aligned}G_{0,0} &= 16\nu_z^2 + 8\nu_z^2 d_0 + 16\nu_z\nu_0 + 8\nu_z\nu_0 d_0 + 8\nu_0^2 + 4v_0^2 d_0 \\ G_{0,1} &= 8\nu_z^2 d_1 + 4\nu_z\left(\alpha_1+\nu_0\right)d_1 + 4\gamma_{0,1}d_1 \\ G_{0,2} &= 8\nu_z^2 d_2 + 4\nu_z\left(\alpha_2+\nu_0\right)d_2 + 4\gamma_{0,2}d_2 \\ G_{1,0} &= 8\nu_z^2 d_0 + 4\nu_z\left(\nu_0+\alpha_1\right)d_0 + 4\gamma_{0,1}d_0 \\ G_{1,1} &= 16\nu_z^2 + 8\nu_z^2 d_1 + 16\nu_z\alpha_1 + 8\nu_z\alpha_1 d_1 + 4\beta_1 + 4\beta_1 d_1 \\ G_{1,2} &= 8\nu_z^2 d_2 + 4\nu_z\left(\alpha_2+\alpha_1\right)d_2 + 4\gamma_{1,2}d_2 \\ G_{2,0} &= 8\nu_z^2 d_0 + 4\nu_z\left(\nu_0+\alpha_2\right)d_0 + 4\gamma_{0,2}d_0 \\ G_{2,1} &= 8\nu_z^2 d_1 + 4\nu_z\left(\alpha_1+\alpha_2\right)d_1 + 4\gamma_{1,2}d_1 \\ G_{2,2} &= 16\nu_z^2 + 8\nu_z^2 d_2 + 16\nu_z\alpha_2 + 8\nu_z\alpha_2 d_2 + 4\beta_2 + 4\beta_2 d_2.\end{aligned}$$

Therefore, the optimal representation is given by the solution to the following set of linear equations:

$$\begin{pmatrix} u_0^2 \\ u_1^2 \\ u_2^2 \end{pmatrix} = 2m\,G^{-1}\begin{pmatrix} \nu_0+\nu_z \\ \alpha_1+\nu_z \\ \alpha_2+\nu_z \end{pmatrix}, \tag{62}$$

Table 1: Simulation results. For each mixture ratio we report the mean AUC and the standard deviation across 10 repetitions of the experiment. Results are reported for in-distribution scenario (P_C), and class distribution shift (Q_C). Best result is marked in bold.

		$\rho = 0.05$	$\rho = 0.1$	$\rho = 0.3$
In Distribution	ERM	0.948±0.013	0.948±0.007	0.913±0.017
	Hier	0.945±0.013	**0.949**±0.010	**0.917**±0.016
	IRM	0.945±0.013	0.947±0.009	0.909±0.018
	CLOvE	0.944±0.008	0.949±0.011	0.911±0.020
	VarREx	0.949±0.013	0.948±0.009	0.910±0.022
	VarAUC	**0.950**±0.017	0.947±0.008	0.912±0.022
Distribution Shift	ERM	0.731±0.007	0.808±0.015	**0.883**±0.014
	Hier	0.727 ± 0.009	0.810 ± 0.014	0.882 ± 0.020
	IRM	0.724±0.017	0.806±0.019	0.880±0.023
	CLOvE	0.745±0.020	0.807±0.018	0.878±0.020
	VarREx	0.729±0.005	0.811±0.018	0.880±0.026
	VarAUC	**0.767**±0.008	**0.838**±0.019	0.881 ±0.024

where

$$G = \begin{pmatrix} G_{0,0} & G_{0,1} & G_{0,2} \\ G_{1,0} & G_{1,1} & G_{1,2} \\ G_{2,0} & G_{2,1} & G_{2,2} \end{pmatrix}. \tag{63}$$

C Comparison to OOD Environment Balancing Methods

Previous methods in the field of OOD generalization (see §2) exhibit several key differences compared to our setting: (i) They address closed-world classification, whereas in zero-shot learning, new classes are encountered. (ii) The presumed shift is typically in the conditional distribution of the data given the class (e.g., the background given the class being a cow or a camel), whereas we consider shifts in the class distribution $P(c)$. (iii) Existing methods often assume that training data comes from various data environments, providing explicit information about how the distribution might shift, while we assume the attribute A causing the shift is unknown.

Despite these differences, in this work we recast class distribution shifts in zero-shot learning into environment balancing OOD setting, by making the following observations. First, when posed as verification methods, zero-shot classifiers in fact perform a binary (closed-world) classification task, predicting whether a pair of data points $x_{ij} \coloneqq (z_i, z_j)$ belong to the same class $y_{ij} = \mathbb{1}_{c_i=c_j}$.

Note that the distribution of possible pairs $x_{ij} = (z_i, z_j)$ given the label y_{ij} changes with variations in class attribute probabilities, and therefore across synthetic environments S. Thus, in this formulation the shift occurs in the conditional distribution of the data given the class $p(x_{ij}|y_{ij})$.

Another distinction lies in data availability: in the setting of closed-world OOD environment balancing methods, a main drawback is the challenge of securing a sufficient number of diverse training environments. This is essential to ensure that a representation performing well on observed environments, will likely perform similarly on unobserved ones. In contrast, our framework allows for the construction of many synthetic environments via sampling.

D Additional Empirical Results

D.1 Additional Simulation Results

Simulations Exact mean and standard deviations matching Figure 4 are provided in table 1.

AUC progress during training iterations and feature importance results for the majority class proportion of $\rho = 0.1$ were shown in the main text. Here, we provide analogous results for $\rho = 0.05$ and $\rho = 0.3$. These are summarized in Figure 8.

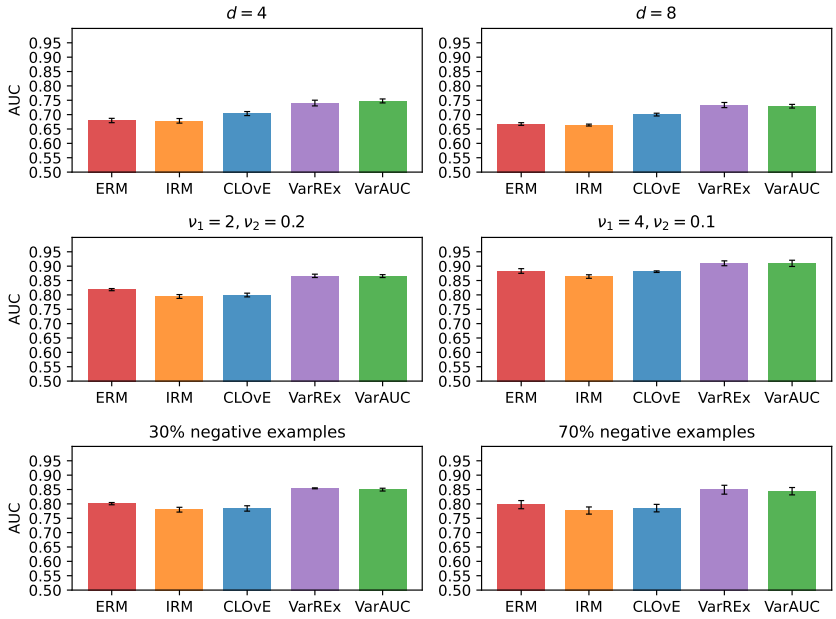

Figure 7: Additional simulation results. Top row: Additional dimensions of the representation. Middle row: additional rations of the attribute variances. Bottom row: unbalanced sets of positive and negative examples. Bars show mean AUC values on the test set across 5 repetitions of the experiment, whiskers show $\pm$ standard deviation.

For $\rho = 0.05$ the convergence results are similar to those obtained for $\rho = 0.1$ – under distribution-shift the two variance based methods show significantly better results compared to other approaches. Our algorithm with the VarREx penalty achieves high AUC values more quickly than the VarAUC penalty, but the VarAUC penalty attains higher accuracy overall. The CLOvE penalty achieves improvement over ERM, but smaller compared to the variance based methods. IRM converges to the same AUC as ERM. In contrast, on in-distribution data all methods perform well.

For $\rho = 0.3$ the distribution shift is milder and therefore ERM performs very well (0.902 AUC is achieved on distribution shift scenario compared to 0.932 on in-distribution setting). Therefore encouragement of similar performance across different data subsets does not benefit the learning process. Slightly better result is achieved with VarREx penalty (0.911).

The analysis of feature importance for $\rho = 0.05$ yields results similar to those for $\rho = 0.1$. At $\rho = 0.3$ the analysis remains mostly unchanged, except that VarREx assigns higher importance to features corresponding to ν_0 (0-5) compared to VarAUC, while in more extreme distribution shifts VarAUC assigns higher importance to the shared features.

D.2 Additional Representation Sizes, Noise Ratios and Positive Proportions

In §5.1 we explored varying values of ρ in a setting where $\nu^+ = 2$, $\nu^- = 0.1$ ($\frac{\nu^+}{\nu^-} = 20$). We now focus on the case of $\rho = 0.1$ and examine additional representation sizes p, and noise ratios ($\frac{\nu^+}{\nu^-} \in \{10, 40\}$). Additionally, we examine the original setting where $p = 16$ and $\nu^+ = 2$, $\nu^- = 0.1$, with varying proportions of positive and negative examples.

The results in Figure 7 show that in all the additional settings our methods provides statistically significant improvement over the baseline. FDR adjusted p-values for multiple comparisons are provided in Table 2.

Table 2: FDR adjusted p-values for the results reported in Figure 7

Experiment	IRM	CLOvE	VarREx	VarAUC
$p = 4$	0.7339	0.0112	0.0003	0.0001
$p = 8$	0.8552	0.0005	0.0003	0.0001
$\nu_1 = 2, \nu_2 = 0.2$	0.9995	0.9995	0.0002	<0.0001
$\nu_1 = 4, \nu_2 = 0.1$	0.9989	0.9971	0.0041	0.0041
30% negative	0.9939	0.9939	<0.0001	<0.0001
70% negative	1.0	1.0	<0.0001	0.0002

Experiments In Table 3, we provide the means and standard deviations for the experiments detailed in §5.2. Additionally, Table 4 presents the adjusted p-values for assessing the performance increase over the ERM baseline achieved by our algorithm with the explored penalties.

Table 3: Experimental results. Mean and standard deviation of AUC values over 5 repetitions are reported for in distribution scenario (P_C), and class distribution shift (Q_C). Best result is marked in bold.

		IN DISTRIBUTION	DISTRIBUTION SHIFT
CELEBA	**ERM**	0.826 ± 0.001	0.666 ± 0.001
	IRM	0.843 ± 0.009	0.659 ± 0.087
	CLOVE	$\mathbf{0.853} \pm 0.002$	0.677 ± 0.012
	VARREX	0.834 ± 0.002	0.676 ± 0.004
	VARAUC	0.836 ± 0.002	$\mathbf{0.697} \pm 0.027$
ETHEC	**ERM**	0.869 ± 0.004	0.786 ± 0.030
	IRM	0.879 ± 0.004	0.795 ± 0.034
	CLOVE	$\mathbf{0.888} \pm 0.004$	0.800 ± 0.040
	VARREX	0.877 ± 0.007	0.805 ± 0.033
	VARAUC	0.872 ± 0.004	$\mathbf{0.838} \pm 0.049$

Table 4: Adjusted p-values for one-sided paired t-tests for testing the improvements over the ERM baseline.

	CELEBA	ETHEC
HIERARCHICAL	0.0154	0.7677
IRM	0.6117	0.1290
CLOVE	0.0119	0.0383
VARREX	< 0.0001	0.0383
VARAUC	0.0058	0.0383

D.3 Analysis of Loss Values

Here we present an analysis of the unpenalized loss after convergence in both real-data experiments. We performed separate analyses on pairs of data points from the dominant type during training (majority), and those from the other type (minority). Additionally, we separated positive pairs ($y = 1$) and negative pairs ($y = 0$). Figure 9 displays histograms illustrating the differences between losses on the training set obtained with the representation learned using ERM (g_{ERM}), and those obtained using our algorithm with VarAUC penalty (g_{VarAUC}):

$$\text{Diff}_{ij} = \ell(x_{ij}, y_{ij}; d_{g_{\text{ERM}})} - \ell(x_{ij}, y_{ij}; d_{g_{\text{VarAUC}}}).$$

Positive values of the differences correspond to higher losses for ERM.

In both experiments, when examining negative pairs from the minority group, as shown in the top-left histograms, most of the observed differences are positive. This indicates that the ERM losses for these pairs are higher compared to the losses obtained for the representation trained with the VarAUC

penalty. The disparities are smaller for the other three groups: majority negative pairs, minority negative pairs, and minority positive pairs. Among these groups, ERM performs better on positive pairs.

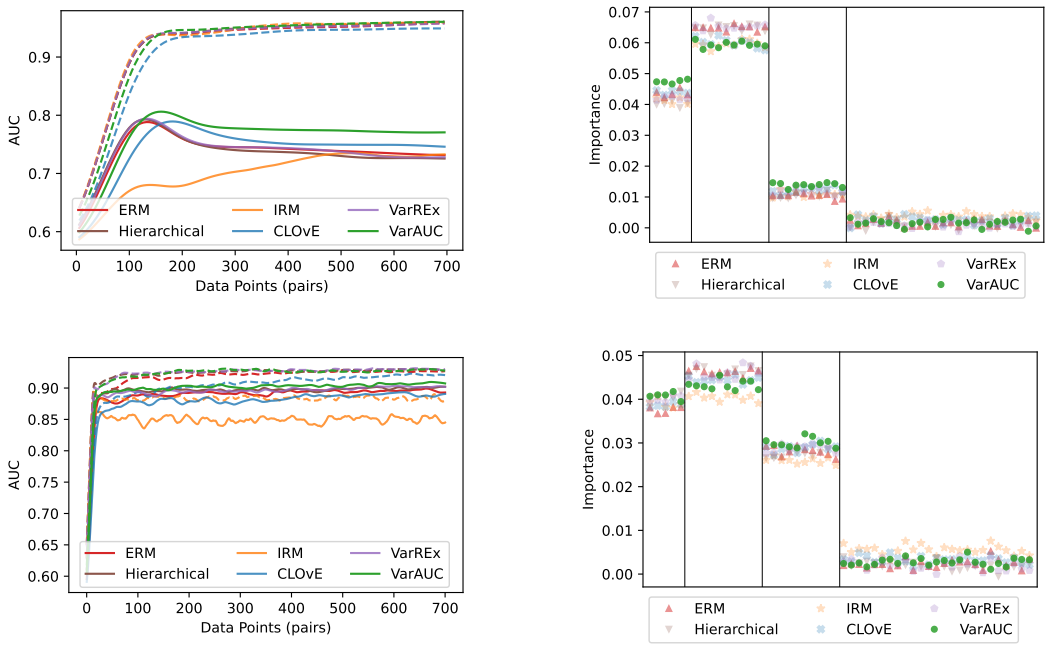

Figure 8: Additional Simulation Results. Top row: $\rho = 0.05$, Bottom row: $\rho = 0.3$. Left: Average AUC progress over 10 repetitions of the simulation. Solid lines correspond to performance on test data (distribution shift scenario), dashed lines show performance on data sampled from the same distribution as training data (in-distribution scenario). Right: Average feature importance results over 10 repetitions.

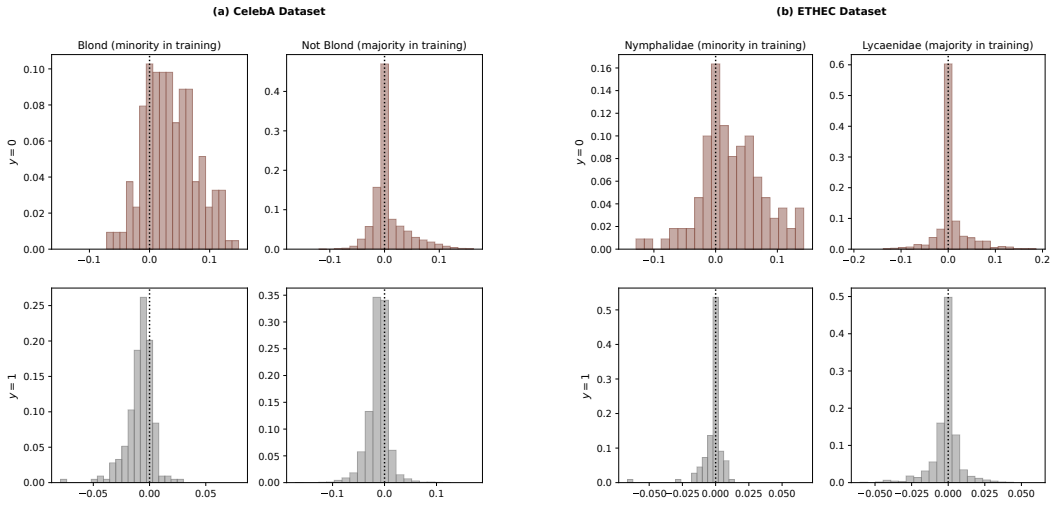

Figure 9: Analysis of Loss Differences. Histograms of differences between ERM and our algorithm with VarAUC penalty are shown for two experiments in separate sub-figures: (a) CelebA dataset, (b) ETHEC dataset. The top rows show differences for negative pairs ($y = 0$), bottom ones show differences for positive pairs ($y = 1$). In each sub-figure the left column corresponds to the minority type and right one to the majority. A dotted black line marks a difference of 0. Positive values correspond to higher losses for ERM.

E Datasets

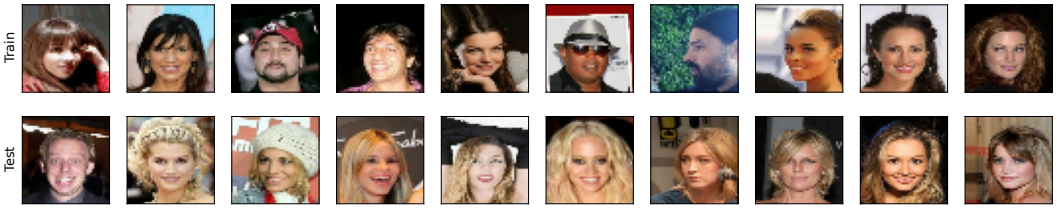

Figure 10: Sample Images from the CelebA Dataset. Top: a random sample of the training data with 95% non-blond people. Bottom: a random sample of the test data with 95% blond people.

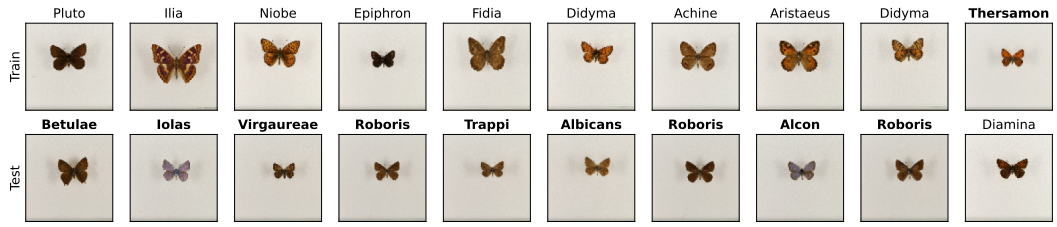

Figure 11: Sample Images from the ETHEC Dataset. Top: a sample of the the training data – 9 species of the Lycaenidae family and 1 from the Nymphalidae family. Bottom: a sample of the test data where the proportion of the families is reversed. Nymphalidae species names are marked in bold.

F Implementation Details

A link to a permanent repository with code to reproduce our results is included in the main text.

The data-related parameters of our experiments are described in the main text. In all our experiments we used margin of $m = 0.5$ for the contrastive loss and Adam (Kingma & Ba, 2014) optimizer to train all models.

For the CLOvE penalty we used a Laplacian kernel $k(r, r') = e^{\frac{1}{\text{width}} - |r - r'|}$ with width of 0.4 as originally suggested by Kumar et al. (2018).

For optimization of the VarAUC objective we disregard the finite sample correction $\frac{N_s - n}{N_s - 1}$ in the implementation since n is very small compared to N_s. In practice, we minimize the standard deviation instead of the variance in both variance based penalties, and the hyperparameters are reported accordingly.

In our scenario where the attribute of interest is unknown, we generated a synthetic attribute for hyperparameter selection using Principle Components (PC). We ranked examples based on their first PC component values, classifying the top 10% as positive and the rest as negative. Hyperparameters for all methods were chosen via grid-search in a single experiment repetition, ensuring robustness against this synthetic attribute. Notably, the experiments themselves did not involve the PC attribute; instead, they focused on dimension swapping in simulations and attributes like hair color or species family in CelebA and ETHEC experiments.

The grid search produced almost identical hyperparameters for all three ρ values. We observed that performance converged to the same value when employing hyperparameters derived from cross-validation for one ρ value, as those selected for another. Therefore, for simplicity we repeated simulations using the same hyperparameters, determined based on the grid search results for $\rho = 0.1$ (the intermediate parameter value). Similarly, minimal differences in optimal learning rates were observed among the methods within an experiment and therefore a shared learning rate was used for each experiment. To emphasize the improvement of OOD methods over the ERM baseline, we used the learning rate optimized for the ERM method. Large differences were observed in optimal

regularization factors, and therefore these parameters (as well as method-specific parameters) were not shared. All hyper-parameters are reported in Table 5.

All models were initialized with identical weights, and trained on identical data splits.

All the code in this work was implemented in Python 3.10. We used the TensorFlow 2.13 and TensorFlow Addons 0.21 packages. For evaluation we used the `auc` function from scikit-learn 1.2. The CelebA dataset was loaded through TensorFlow Datasets 4.9 and pandas 1.5 was used to process the ETHEC dataset. Statistical tests were performed using `ttest_rel` and `false_discovery_control` functions from scipy.stats 1.11.4. All figures were generated using Matplotlib 3.7.

The IRM implementation was adapted from the source code of the paper, available at `https://github.com/facebookresearch/InvariantRiskMinimization`.

We ran all experiments on a single A100 cloud GPU. For simulations, each full repetition of the experiment (comparing all methods) required on average 2.06 hours. Each repetition on the ETHEC dataset took 7.38 hours on average, and on the CelebA dataset 11.52 hours.

Table 5: Hyper Parameters.

		ERM	IRM	CLOvE	VarREx	VarAUC
Simulations	Learning rate η	0.01	0.01	0.01	0.01	0.01
	Regularization factor λ	–	0.01	0.05	3.0	1.5
	Network weight regularizer	–	0.01	–	–	–
CelebA	Learning rate η	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	Regularization factor λ	–	0.1	0.085	0.01	0.2
	Network weight regularizer	–	0.01	–	–	–
ETHEC	Learning rate η	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}
	Regularization factor λ	–	0.02	0.05	0.1	0.2
	Network weight regularizer	–	0.01	–	–	–

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes] ,

   Justification: Both abstract and the introduction (last 2 paragraphs) accurately state the main contributions of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Main limitations are discussed in §6.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All assumptions are clearly stated and full proofs to the theoretical claims appear in Appendix B

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All the details are provided in Section §5 and Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The datasets are publicly available and code implementing all our results is submitted with the paper.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We specify all hyperparameters and training details in Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We perform statistical testing to provide significance of our results and report FDR adjusted p-values. For all reported results we include either error bars in the main text, or when other visualizations are chosen we report means and standard deviations in Appendix D.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All resources including GPU information and run times are provided in Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper conforms with the provided code of ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This paper presents work whose goal is to advance the field of learning robust data representations. It is not tied to any particular applications and therefore we do not see an immediate risk for negative societal impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We do not release any new data or models.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The only asset used is the IRM implementation. The corresponding paper is cited and we explicitly mention this in Appendix F, while providing also reference for the code itself.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve human subjects and did not use crowd sourcing.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve human subjects and did not use crowd sourcing.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.