

Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems

School of Computing and Information Systems

10-2022

Equivariance and invariance inductive bias for learning from insufficient data

Tan WANG

Nanyang Technological University

Qianru SUN

Singapore Management University, qianrusun@smu.edu.sg

Sugiri PRANATA

Panasonic R &D Center Singapore

Karlekar JAYASHREE

Panasonic R &D Center Singapore

Hanwang ZHANG

Nanyang Technological University

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research



Part of the [Databases and Information Systems Commons](#), [Graphics and Human Computer Interfaces Commons](#), and the [Numerical Analysis and Scientific Computing Commons](#)

Citation

WANG, Tan; SUN, Qianru; PRANATA, Sugiri; JAYASHREE, Karlekar; and ZHANG, Hanwang. Equivariance and invariance inductive bias for learning from insufficient data. (2022). *Computer Vision ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23-27: Proceedings*. 13671, 241-258.

Available at: https://ink.library.smu.edu.sg/sis_research/7513

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylds@smu.edu.sg.

Equivariance and Invariance Inductive Bias for Learning from Insufficient Data

Tan Wang¹ Qianru Sun² Sugiri Pranata³ Karlekar Jayashree³
Hanwang Zhang¹

¹Nanyang Technological University ²Singapore Management University

³Panasonic R&D Center Singapore

{tan317,hanwangzhang}@ntu.edu.sg qianrusun@smu.edu.sg

{sugiri.pranata,karlekar.jayashree}@sg.panasonic.com

Abstract. We are interested in learning robust models from insufficient data, without the need for any externally pre-trained checkpoints. First, compared to sufficient data, we show why insufficient data renders the model more easily biased to the limited training environments that are usually different from testing. For example, if all the training **swan** samples are “white”, the model may wrongly use the “white” environment to represent the intrinsic class **swan**. Then, we justify that **equivariance** inductive bias can retain the class feature while **invariance** inductive bias can remove the environmental feature, leaving the class feature that generalizes to any environmental changes in testing. To impose them on learning, for equivariance, we demonstrate that any off-the-shelf contrastive-based self-supervised feature learning method can be deployed; for invariance, we propose a class-wise invariant risk minimization (IRM) that efficiently tackles the challenge of missing environmental annotation in conventional IRM. State-of-the-art experimental results on real-world benchmarks (VIPriors, ImageNet100 and NICO) validate the great potential of **equivariance** and **invariance** in data-efficient learning. The code is available at <https://github.com/Wangt-CN/EqInv>.

Keywords: Inductive Bias, Equivariance, Invariant Risk Minimization

1 Introduction

Data is never too big. As illustrated in Fig. 1 (a), if we have sufficiently large training sample size of **swan** and **dog**, *e.g.*, dogs and cats in any environment such as different colors, shapes, poses, and backgrounds, by using a conventional softmax cross-entropy based “**swan vs. dog**” classifier, we can obtain a “perfect” model that discards the *shared environmental* features but retains the *discriminative class* features. The underlying common sense is that if the model has seen any “case” in training, the testing data is merely a seen IID subset of the training data, yielding testing accuracy as good as training [76].

In this paper, we are interested in learning from insufficient data. Besides the common motivation that collecting data is expensive, we believe that how

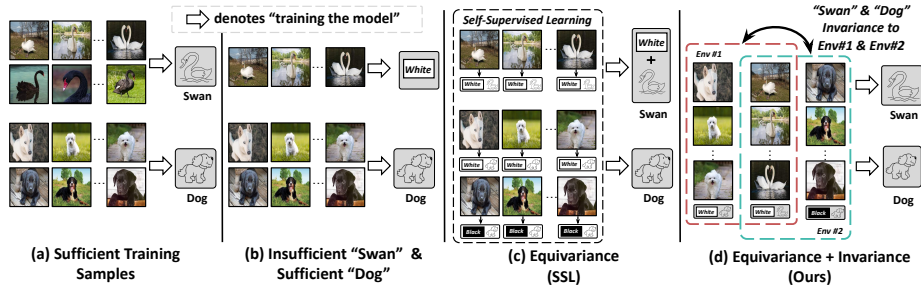



Fig. 1: Illustration of how the proposed equivariance and invariance inductive biases help learning from insufficient data. Cartoon figures such as  denote the class feature. Boxed words such as `White` denote environmental features. Grey-boxed figures denote the learned model. For simple illustration, we omit the environment as background.

to narrow the performance gap between insufficient and sufficient data is the key to tackling the non-IID challenge in machine generalization—even if the training data is sufficient, the testing can still be out of the training distribution (OOD) [32,67,70]. After all, we can always frame up exceptional testing samples that fail the trained model [37,28]. Note that **different from few-shot learning** which widely adopts pre-training on large-scale training set [74,71,73], our task does not allow using any externally pre-trained checkpoint and backbone¹.

Fig. 1 (b) illustrates why insufficient data hurts generalization. Without loss of generality, we conduct a thought experiment that we have limited `swan` only in “white” color environment while sufficient `dog` in diverse environments. So, we can expect that the “dog” feature will still be extracted to represent `dog` model, but the “white” feature will be recklessly learned to represent `swan`. This is because training `swan` model by using either “swan” or “white” feature yields the similar training risk: 1) if the former, the training loss is minimized as in the perfect case of Fig. 1 (a); 2) if the latter, the only training error possible would be misclassifying “white dog” as `swan`. However, it can be easily corrected in practice, *e.g.*, by discriminatively training a sample-to-model distance prior that $\|\mathbf{z}_{\text{dog}}\| > \|\mathbf{z}_{\text{white}}\|$, where \mathbf{z} denotes the feature vector².

Why, under the same training risk, does the `swan` model prefer “white” but not “swan” feature? First, feature extraction in deep network follows a bottom-up, low-level to high-level fashion [49]—“simple” features such as color can be easily learned at lower layers, while “complex” features such as object parts will be only emerged in higher layers [69,86,82]. Second, the commonly used cross-entropy loss encourages models to stop learning once “simple” features suffice to minimize the loss [25,26]. As a result, “swan” features like “feather”, “beak”, and “wing” will be lost after training. Such mechanism is also well-known as the

¹ See <https://vipriors.github.io/> for details.

² The distance between the “white dog” sample vector ($\mathbf{z}_{\text{white}}, \mathbf{z}_{\text{dog}}$) and the `swan` model vector ($\mathbf{z}_{\text{white}}, \mathbf{0}$) is: $\|(\mathbf{z}_{\text{white}}, \mathbf{z}_{\text{dog}}) - (\mathbf{z}_{\text{white}}, \mathbf{0})\| = \|(\mathbf{0}, \mathbf{z}_{\text{dog}})\| = \|\mathbf{z}_{\text{dog}}\|$; similarly, we have the distance between “white dog” and `dog` model as $\|\mathbf{z}_{\text{white}}\|$.

shortcut bias [25] or spurious correlation in causality literature [65,78]. We will provide formal justifications in Section 3.1.

By comparing the difference between Fig. 1 (a) and Fig. 1 (b), we can see that the crux of improving the generalization of insufficient data is to recover the missing “swan” class feature while removing the “white” environmental feature. To this end, we propose two inductive biases to guide the learning: *equivariance* for class preservation and *invariance* for environment removal.

Equivariance. This prior requires that the feature representation of a sample should be equivariant to its semantic changes, *e.g.*, any change applied to the sample should be faithfully reflected in the feature change (see Appendix for the mathematical definition). Therefore, if we impose a contrastive loss for each sample feature learning, we can encourage that different samples are mapped into different features (see Section 3.2 for a detailed analysis and our choice). As illustrated in Fig. 1 (c), equivariance avoids the degenerated case in Fig 1 (b), where all “white swan” samples collapse to the same “white” feature. Thus, for a testing “black swan”, the retained “swan” feature can win back some **swan** scores despite losing the similarity between “black” and “white”. It is worth noting that the equivariance prior may theoretically shed light on the recent findings that self-supervised learning features can improve model robustness [36,68,35,79]. We will leave it as future work.

Invariance. Although equivariance preserves all the features, due to the limited environments, the **swan** model may still be confounded by the “white” environment, that is, a testing “black swan” may still be misclassified as **dog**, *e.g.*, when $\|(\mathbf{z}_{\text{black}} - \mathbf{z}_{\text{white}}, \mathbf{z}_{\text{swan}} - \mathbf{z}_{\text{swan}})\| > \|(\mathbf{z}_{\text{black}} - \mathbf{0}, \mathbf{z}_{\text{swan}} - \mathbf{z}_{\text{dog}})\|$. Inspired by invariant risk minimization [4] (IRM) that removes the environmental bias by imposing the environmental invariance prior (Section 3.3), as shown in Fig. 1 (d), if we split the training data into two environments: “white swan” vs. “white dog” and “white swan” vs. “black dog”, we can learn a common classifier (*i.e.*, a feature selector) that focuses on the “swan” and “dog” features, which are the *only invariance* across the two kinds of color environments—one is identical as “white” and the other contains two colors. Yet, conventional IRM requires environment annotation, which is however impractical. To this end, in Section 4, we propose **class-wise IRM** based on contrastive objective that works efficiently without the need for the annotation. We term the overall algorithm of using the two inductive biases, *i.e.*, **equivariance** and **invariance**, as EQINV.

We validate the effectiveness of EQINV on three real-world visual classification benchmarks: 1) VIPriors ImageNet classification [12], where we evaluate 10/20/50 samples per class; 2) NICO [32], where the training and testing environmental distributions are severely different; and 3) ImageNet100 [75] which denotes the case of sufficient training data. On all datasets, we observe significant improvements over baseline learners. Our EQINV achieves a new single-model state-of-the-art on test split: 52.27% on VIPriors-50 and 64.14% on NICO.

2 Related Work

Visual Inductive Bias. For a learning problem with many possible solutions, inductive bias is a heuristic prior knowledge that regularizes the learning behavior to find a better solution [57]. It is ubiquitous in any modern deep learning models: from the shallow MLP [55] to the complex deep ResNet [10,3] and Transformers [80,18]. Inductive biases can be generally grouped into two camps: 1) Equivariance: the feature representation should faithfully preserve all the data semantics [19,20,52]. 2) Invariance: generalization is about learning to be invariant to the diverse environments [77,7]. Common practical examples are the pooling/striding in CNN [44], dropout [33], denoising autoencoder [42], batch normalization [23], and data augmentations [11,14].

Data-Efficient Learning. Most existing works re-use existing datasets [87,13] and synthesize artificial training data [48,22]. We work is more related to those that overcome the data dependency by adding prior knowledge to deep nets [9,27]. Note that data-efficient learning is more general than the popular setting of few-shot learning [74,71,73] which still requires external large pre-training data as initialization or meta-learning. In this work, we offer a theoretical analysis for the difference between learning from insufficient and sufficient data, by posing it in an OOD generalization problem.

OOD Generalization. Conventional machine generalization is based on the Independent and Identically Distributed (IID) assumption for training and testing data [76]. However, this assumption is often challenged in practice—the Out-of-Distribution (OOD) problem degrades the generalization significantly [34,84,67]. Most existing works can be framed into the stable causal effect pursuit [65,78,43] or finding an invariant feature subspace [62,81]. Recently, Invariant Risk Minimization (IRM) takes a different optimization strategy such as convergence speed regularization [4,47] and game theory [1]. Our proposed class-wise IRM makes it more practical by relaxing the restrictions on needing environment annotation.

3 Justifications of the Two Inductive Biases

As we discussed in Section 1, given an image $X = x$ with label $Y = y$, our goal is to extract the intrinsic class feature $\phi(x)$ invariant to the environmental changes $z \in Z$. Specifically, Z is defined as all the class-agnostic items in the task of interest. For example, spatial location is the intrinsic class feature in object detection task, but an environmental feature in image classification. This goal can be achieved by using the interventional Empirical Risk Minimization (ERM) [43]. It replaces the observational distribution $P(Y|X)$ with the interventional distribution $P(Y|do(X))$, which removes the environmental effects from the prediction of Y , making $Y = y$ only affected by $X = x$ [63]. The interventional empirical risk \mathcal{R} with classifier f can be written as (See Appendix for the detailed derivation):

$$\begin{aligned} \mathcal{R} &= \mathbb{E}_{x \sim P(X), y \sim P(Y|do(X))} \mathcal{L}(f(\phi(x)), y) \\ &= \sum_x \sum_y \sum_z \mathcal{L}(f(\phi(x)), y) P(y|x, z) P(z) P(x), \end{aligned} \tag{1}$$

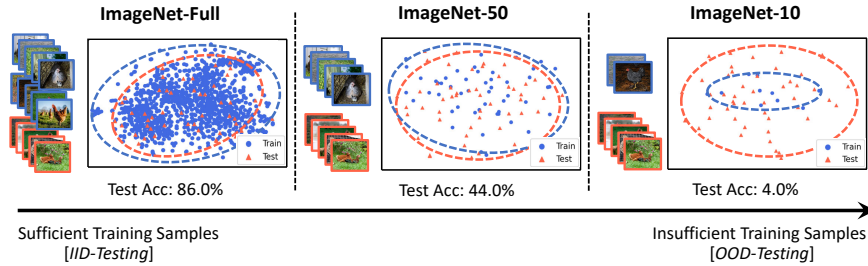


Fig. 2: The t-SNE [56] data visualization of class “hen” on different-scale ImageNet dataset using CLIP [66] pretrained feature extractor. Blue dot and orange triangle represent training and testing samples, respectively. The testing accuracy is evaluated by ResNet-50 [31] trained from scratch on each dataset. See Appendix for details.

where $\mathcal{L}(f(\phi(x)), y)$ is the standard cross-entropy loss. Note that Eq. (1) is hard to implement since the environment Z is unobserved in general.

When the training data is sufficient, X can be almost observed in any environment Z , leading to the approximate independence of Z and X , *i.e.*, $P(Z|X) \approx P(Z)$. Then \mathcal{R} in Eq. (1) approaches to the conventional ERM $\hat{\mathcal{R}}$:

$$\mathcal{R} \approx \hat{\mathcal{R}} = \sum_x \sum_y \mathcal{L}(f(\phi(x)), y) P(y|x) P(x) = \mathbb{E}_{(x,y) \sim P(X,Y)} \mathcal{L}(f(\phi(x)), y), \quad (2)$$

3.1 Model Deficiency in Data Insufficiency

However, when the training data is insufficient, $P(Z|X)$ is no longer approximate to $P(Z)$ and thus $\hat{\mathcal{R}} \not\approx \mathcal{R}$. For example, $P(Z = \text{White} | \text{🐔}) > P(Z = \text{Black} | \text{🐔})$. Then, as we discussed in Section 1, some simple environmental semantics Z , *e.g.*, $Z = \text{White}$, are more likely dominant in minimizing $\hat{\mathcal{R}}$ due to $P(y|x) = P(y|x,z)P(z|x)$ in Eq. (2), resulting the learned ϕ that mainly captures the dominant environment but missing the intrinsic class feature. Empirical results in Fig. 2 also support such analysis. We show the ImageNet classification results of class **hen** using various training sizes. We can observe that with the decreasing of training samples, the accuracy degrades significantly, from 86.0% to 4.0%. After all, when the training size is infinite, any testing data is a subset of training.

3.2 Inductive Bias I: Equivariant Feature Learning

To win back the missing intrinsic class feature, we impose the contrastive-based self-supervised learning (SSL) techniques [15,30,61], without the need for any external data, to achieve the equivariance. In this paper, we follow the definition and implementation in [77] to achieve sample-equivariant by using contrastive learning, *i.e.*, different samples should be respectively mapped to different features. Given an image x , the data augmentation of x constitute the positive example x^+ , whereas augmentations of other images constitute N negatives x^- .

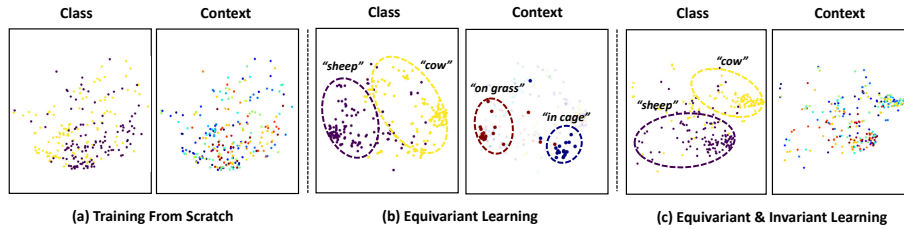


Fig. 3: The t-SNE [56] visualization of learned features *w.r.t* both class and context annotations on NICO dataset with (a) training from scratch; (b) equivariant learning; and (c) equivariant & invariant learning.

The key of contrastive loss is to map positive samples closer, while pushing apart negative ones in the feature space:

$$\mathbb{E}_{x, x^+, \{x_i^-\}_{i=1}^N} \left[-\log \frac{\exp(\phi(x)^T \phi(x^+))}{\exp(\phi(x)^T \phi(x^+)) + \sum_{i=1}^N \exp(\phi(x)^T \phi(x_i^-))} \right]. \quad (3)$$

Note that we are open to any SSL choice, which is investigated in Section 5.

We visualize the features learned by training from scratch and utilizing the equivariance inductive bias on NICO [32] dataset with both class and context annotations. In Fig. 3 (a), it is obvious that there is no clear boundary to distinguish the semantics of class and context in the feature space, while in Fig. 3 (b), the features are well clustered corresponding to both class and context.

3.3 Inductive Bias II: Invariant Risk Minimization

Although the equivariance inductive bias preserves all the features, the **swan** model may still be confounded by the “white” feature during the downstream fine-tuning, causing $\mathcal{R} \neq \mathcal{R}$. To mitigate such shortcut bias, a straightforward solution is to use Inverse Probability Weighting (IPW) [5,41,53] (also known as reweighting [6,60,51]) to down weight the overwhelmed “white” feature in **swan**. However, they must follow the positivity assumption [39], *i.e.*, all the environmental semantics Z should exist in each class. However, when the training data is insufficient, such assumption no longer holds. For example, how do you down weight “white” over “black” if there is even no “black swan” sample?

Recently, Invariant Risk Minimization (IRM) [4,47] resolves the non-positivity issue by imposing the invariance inductive bias to directly remove the effect of environmental semantics Z . Specifically, IRM first splits the training data into multiple environments $e \in \mathcal{E}$. Then, it regularizes ϕ to be *equally* optimal in different splits, *i.e.*, invariant across environments:

$$\sum_e \mathcal{L}_e(w^T \phi(x), y) + \lambda \|\nabla_{w=1} \mathcal{L}_e(w^T \phi(x), y)\|_2^2, \quad (4)$$

where λ is trade-off hyper-parameter, w stands for a dummy classifier [4] to calculate the gradient penalty across splits—though different environments may induce different losses, the feature ϕ must regularize them optimal at the same

time (the lower gradient the better) in the same way (by using the common dummy classifier). Note that each environment should contain a unique mode of environmental feature distribution [4,21,2]: suppose that we have k environmental features that are distributed as $\{p_1, p_2, \dots, p_k\}$. If $p_i^{e_1} \neq p_i^{e_2}$, $i = 1$ to k , IRM under the two environments will remove all the k features—the keeping of any one will be penalized by the second term of Eq. (4).

Conventional IRM requires the environment annotations, which are generally impossible in practice. To this end, we propose a novel class-wise IRM to regularize the invariance within each class, without the need for environment supervision. We show the qualitative results of imposing such invariance inductive bias in Fig. 3 (c). Compared to Fig. 3 (b), we can observe that after applying our proposed class-wise IRM, the equivariance of intrinsic class features are reserved with well-clustered data points while the context labels are no longer responsive—the environment features are removed.

4 Our EqInv Algorithm

Fig. 4 depicts the computing flow of EQINV. In the following, we elaborate each of its components.

Input: Insufficient training samples denoted as the pairs $\{(x, y)\}$ of an image x and its label y .

Output: Robust classification model $f \cdot \phi$ with intrinsic class feature $\phi(x)$ and unbiased classifier $f(\phi(x))$.

Step 1: Equivariant Learning via SSL. As introduced in Section 3, a wide range of SSL pretext tasks are sufficient for encoding the sample-equivariance. For fair comparison with other methods in VIPriors challenge dataset [12], we use MoCo-v2 [30,16], SimSiam [17], and IP-IRM [77] to learn ϕ in Fig 4 (a). We leave the results based on the most recent MAE [29] in Appendix.

Step 2: Environment Construction based on Adjusted Similarity. Now we are ready to use IRM to remove the environmental features in ϕ . Yet, conventional IRM does not apply as we do not have environment annotations. So, this step aims to automatically construct environments \mathcal{E} . However, it is extremely challenging to identify the combinatorial number of unique environmental modes—improper environmental split may contain shared modes, which cannot be removed. To this end, we propose an efficient *class-wise* approximation that seeks *two* environments *w.r.t.* each class. Our key motivation is that, for insufficient training data, the environmental variance within each class is relatively simple and thus we assume that it is single-modal. Therefore, as shown in Fig. 4 (b), we propose to use each class (we call **anchor** class) as an anchor environmental mode to split the samples of the rest of the classes (we call **other** classes) into two groups: similar to the **anchor** or not. As a result, for C classes, we will have totally $2C$ approximately unique environments. Intuitively, this class-wise strategy can effectively remove the severely dominant context bias in a class. For example, if all **swan** samples are “white”, the “white” feature can

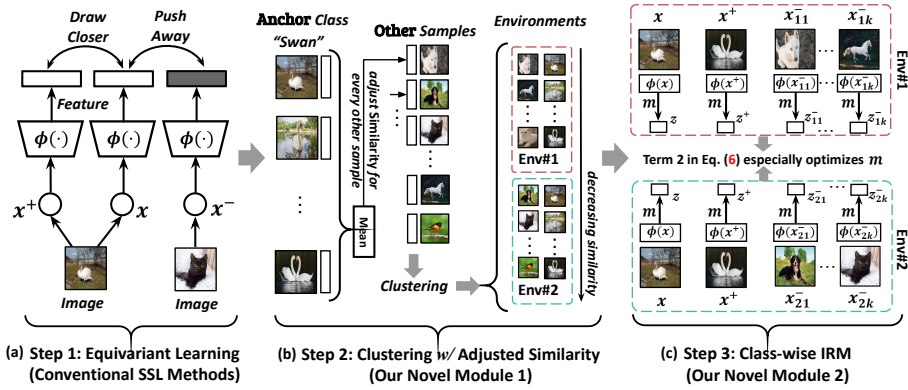


Fig. 4: The flowchart of our proposed EQINV with 3 steps. Rectangle with shading denotes the feature and \mathcal{E}_j represents the generated environment of class j . x_{1k}^- and x_{2k}^- in (c) are the k -th negative samples of subset e_1 and e_2 , respectively. We highlight that class-wise IRM optimizes the mask layer m (and an extra MLP g) without gradients flowing back to the feature extract ϕ .

still be identified as a non-discriminative color feature, thanks to the “black” and “white” samples of dog class.

For each **anchor** class containing l images, environment Env#1 contains these l samples as positive and the “similar” samples from **other** classes as negative; environment Env#2 contains the same positive samples while the “dissimilar” samples from **other** classes as negative. A straightforward way to define the “similarity” between two samples is to use cosine similarity. We compute the cosine similarity between the pair images sampled from **anchor** class and **other** classes, respectively. We get the matrix $\mathbf{S} \in \mathbb{R}^{l \times n}$, where n is the number of images in **other** classes. Then, we average this matrix along the axis of **anchor** class, as in the pseudocode: $\mathbf{s}^+ = \text{mean}(\mathbf{S}, \text{dim} = 0)$. After ranking \mathbf{s}^+ , it is easy to get “similar” samples (corresponding to higher half values in \mathbf{s}^+) grouped in Env#1 and “dissimilar” samples (corresponding to lower half values in \mathbf{s}^+) grouped in Env#2. It is an even split. Fig. 5 (a) shows the resultant environments for **anchor** class 0 on the Colored MNIST³ [60]. using the above straightforward cosine similarity. We can see that the digit classes distribute differently in Env#1 and Env#2, indicating that the difference of the two environments also include class information, which will be disastrously removed after applying IRM.

To this end, we propose a similarity adjustment method. It is to adjust every sample-to-class similarity by subtracting a class-to-class similarity, where the sample belongs to the class. First, we calculate the class-to-class similarity \bar{s}_i between the i -th ($i = 1, \dots, C - 1$) **other** class and the **anchor** class: $\bar{s}_i = \text{mean}(\mathbf{s}^+[a_i : b_i])$, where we assume that the image index range of the i -th **other** class is $[a_i : 1 : b_i]$. Such similarity can be viewed as a purer “class effect” to be

³ It is modified from MNIST dataset [50] by injecting *color* bias on each digit (class). The non-bias ratio is 0.5%, e.g., 99.5% samples of 0 are in red and only 0.5% in uniform colors.

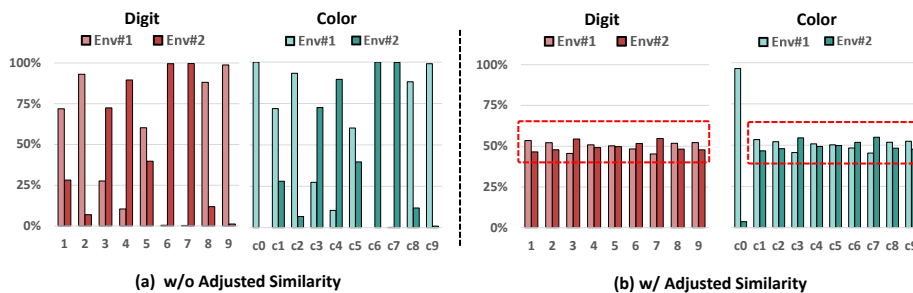


Fig. 5: The obtained environments \mathcal{E}_0 for an example **anchor** class 0 on the Colored MNIST [60], by using (a) the vanilla cosine similarity and (b) our adjusted similarity. On X-axis, 1-9 are **other** digit classes, and c0-c9 denote 10 colors used to create this color-bias dataset. On Y-axis, the percentage point denotes the proportion of a digit (or a color) grouped into a specific environment.

removed from the total effect of both class and environment—only “environment effect” is then left. Therefore, for any sample x^j from the i -th **other** class, its adjusted similarity to the **anchor** class is: $s = \mathbf{s}^+[j] - \bar{s}_i$. Using this similarity, we obtain new environments and show statistics in Fig. 5 (b). It is impressive that the biased color of **anchor** class 0 (*i.e.*, the 0-th color c0 or red) varies between Env#1 and Env#2, but the classes and other colors (red dashed boxes) distribute almost uniformly in these two environments. It means the effects of class and environment are disentangled.

Step 3: Class-wise Invariant Risk Minimization. With the automatically constructed environments, we are ready to remove the environmental feature from ϕ . In particular, we propose a class-wise IRM based on the contrastive objective, which is defined as follows. As shown in Fig. 4 (c), given a training image x in environment e of class i , we use a learnable vector mask layer m multiplied on $\phi(x)$ to select the invariant feature. Then, we follow [15] to build a projection layer $g(\cdot)$ to obtain $\mathbf{z} = g(m \circ \phi(x))$ for contrastive supervision, where g is a one-hidden-layer MLP with ReLU activation and \circ denotes element-wise production. For each **anchor** class k , we define an environment-based supervised contrastive loss [45]. It is different from the traditional self-supervised contrastive loss. Specifically, our loss is computed within each environment $e \in \mathcal{E}_k$. We take the representations of **anchor** class samples (in e) as positives \mathbf{z}^+ , and the representations of **other** class samples (in e) as negatives \mathbf{z}^- , and we have:

$$\ell(e \in \mathcal{E}_k, w = 1) = \sum_{\mathbf{z}^+ \in e} \frac{1}{N^+} \sum_{\mathbf{z}^- \in e} \left[-\log \frac{\exp(\mathbf{z}^{\text{T}} \mathbf{z}^+ \cdot w)}{\exp(\mathbf{z}^{\text{T}} \mathbf{z}^+ \cdot w) + \sum_{\mathbf{z}^- \in e} \exp(\mathbf{z}^{\text{T}} \mathbf{z}^- \cdot w)} \right], \quad (5)$$

where N^+ denotes the number of the positive samples in the current minibatch and $w = 1$ is a “dummy” classifier to calculate the gradient penalty term [4]. Therefore, the proposed class-wise IRM loss⁴ is:

$$\mathcal{L}_k = \sum_{e \in \mathcal{E}_k} \ell(e, w = 1) + \lambda \|\nabla_{w=1} \ell(e, w = 1)\|_2^2, \quad (6)$$

⁴ Please note that in implementation, we adopt an advanced version [47] of IRM. Please check appendix for details.

where λ is the trade-off hyper-parameter. The overall training objective is the combination of minimizing a conventional cross entropy \mathcal{L}_{ce} and the class-wise IRM regularization \mathcal{L}_k :

$$\min_{f,g,m,\phi} \mathcal{L}_{ce}(f, m, \phi) + \sum_{k=1}^C \mathcal{L}_k(g, m), \quad (7)$$

where we use $f(m \circ \phi(x))$ for inference. It is worth noting that each loss trains a different set of parameters— ϕ is frozen during the class-wise IRM penalty update. As the equivariance of ϕ is only guaranteed by SSL pretraining, compared to the expensive SSL equivariance regularization in training [77], our frozen strategy is more efficient to mitigate the adversary effect introduced by the invariance bias, which may however discard equivariant features to achieve invariance. We investigate this phenomenon empirically in Section 5.4.

5 Experiments

5.1 Datasets and Settings

VIPriors [12] dataset is proposed in VIPrior challenge [12] for data-efficient learning. It contains the same 1,000 classes as in ImageNet [24], and also follows the same splits of **train**, **val** and **test** data. In all splits, each class contains only 50 images, so the total number of samples in the dataset is $150k$. Some related works [8,54] used the merged set (of **train** and **val**) to train the model. We argue that this to some extent violates the protocol of data-efficient learning—using insufficient training data. In this work, our EQINV models as well as comparing models are trained on the standard **train** set and evaluated on **val** and **test** sets. In addition, we propose two more challenging settings to evaluate the models: **VIPriors-20** and **VIPriors-10**. The only difference with VIPriors is they have 20 and 10 images per class in their **train** sets, respectively. There is no change on **val** and **test** sets. We thus call the original **VIPriors-50**. **NICO** [32] is a real-world image dataset proposed for evaluating OOD methods. The key insight of NICO is to provide image labels as well as context labels (annotated by human). On this dataset, it is convenient to “shift” the distribution of the class by “adjusting” the proportions of specific contexts. In our experiments, we follow the “adjusting” settings in the related work [78]. Specifically, this is a challenging OOD setting using the NICO animal set. It mixes three difficulties: 1) Long-Tailed; 2) Zero-Shot and 3) Orthogonal. See Appendix for more details. **ImageNet100** [75] is a subset of original ImageNet [24] with 100 classes and $1k$ images per class. Different with previous OOD datasets, ImageNet100 is to evaluate the performances of our EQINV and comparison methods in sufficient training data settings.

5.2 Implementation Details

We adopted ResNet-50/-18 as model backbones for VIPriors/ImageNet100 and NICO datasets, respectively. We trained the model with 100 epochs for “training

Table 1: Recognition accuracies (%) on the VIPriors-50, -20, -10, NICO and ImageNet-100 (IN-100) datasets. ‘‘Aug.’’ represents augmentation. Note that due to the effectiveness of ‘‘Random Aug.’’, we set it as a default configuration for the methods trained from SSL. Our results are highlighted.

Model	VIPriors-50 [12]		VIPriors-20		VIPriors-10		NICO [32]		IN-100 [75]
	Val	Test	Val	Test	Val	Test	Val	Test	Val
Baseline	32.30	30.60	13.13	12.39	5.02	4.59	43.08	40.77	83.56
<i>Augmentation</i>									
Stronger Aug. [15]	36.60	34.72	16.17	15.21	3.49	3.26	42.31	43.31	83.72
Random Aug. [22]	41.09	39.18	16.71	16.03	3.88	4.01	45.15	44.92	85.12
Mixup [83]	34.66	32.75	13.35	12.69	2.47	2.31	40.54	38.77	84.52
Label smoothing [58]	33.77	31.87	12.71	12.05	4.76	4.43	39.77	38.15	85.22
<i>Debias Learning</i>									
Lff [60]	35.04	33.29	13.26	12.58	5.20	4.79	41.62	42.54	83.74
Augment Feat. [51]	35.41	33.63	13.62	12.97	3.43	3.12	42.31	43.27	83.88
CaaM [78]	36.13	34.24	14.68	13.99	4.88	4.63	46.38	46.62	84.56
<i>Train from Scratch</i>									
MoCo-v2 [16]	49.47	46.98	30.76	28.83	18.40	16.97	46.45	45.70	86.30
+EqINV (Ours)	54.21	52.09	38.30	36.66	26.70	25.20	52.55	51.51	88.38
SimSiam [17]	42.69	40.75	22.09	21.15	6.84	6.68	41.27	42.68	85.28
+EqINV (Ours)	52.55	50.36	37.29	35.65	24.74	23.33	45.67	44.77	86.80
<i>Train from SSL</i>									
IP-IRM [77]	51.45	48.90	38.91	36.26	29.94	27.88	63.60	60.26	86.94
+EqINV (Ours)	54.58	52.27	41.53	39.21	32.70	30.36	66.07	64.14	87.78

from scratch’’ methods. We initialized the learning rate as 0.1 and decreased it by 10 times at the 60-th and 80-th epochs. We used SGD optimizer and the batch size was set as 256. For equivariant learning (*i.e.*, SSL), we utilized MoCo-v2 [16], SimSiam [17] and IP-IRM [77] to train the model for 800 epochs without using external data, using their default hyper-parameters. We pretrain the model for 200 epochs on ImageNet100 dataset. Then for downstream fine-tuning, We used SGD optimizer and set batch size as 128. We set epochs as 50, initialized learning rate as 0.05, and decreased it at the 30-th and 40-th epochs. Please check appendix for more implementation details. Below we introduce our baselines including augmentation-based methods, debiased learning methods and domain generalization (DG) methods.

Augmentation-based Methods are quite simple yet effective techniques in the VIPriors challenges as well as for the task of data-efficient learning. We chose four top-performing methods in this category to compare with: stronger augmentation [15], random augmentation[22], mixup [83] and label smoothing [58].

Debias Learning Methods. Data-efficient learning can be regarded as a task for OOD. We thus compared our EqINV with three state-of-the-art (SOTA) debiased learning methods: Lff [60], Augment Feat. [51] and CaaM [78].

Domain Generalization Methods. Domain Generalization (DG) task also tackles the OOD generalization problem, but requires sufficient domain samples and full ImageNet pretraining. In this paper, we select three SOTA DG approaches (SD [64], SelfReg [46] and SagNet [59]) for comparison. These methods do not require domain labels which share the same setting as ours.

5.3 Comparing to SOTAs

Table 1 shows the overall results comparing to baselines on VIPriors-50, -20, -10, NICO and ImageNet100 datasets. Our EQINV achieves the best performance across all settings. In addition, we have another four observations. 1) Incorporating SSL pretraining, vanilla fine-tuning can achieve much higher accuracy than all the methods of “training from scratch”. This validates the efficiency of the equivariance inductive bias (learned by SSL) for Etackling the challenge of lacking training data. 2) When decreasing the training size of VIPriors from 50 to 10 images per class, the comparison methods of training from scratch cannot bring performance boosting even hurt the performance. This is because the extremely insufficient data cannot support to establish an equivariant representation, not mention to process samples with harder augmentations. 3) Interestingly, compared to SSL methods, we can see that the improvement margins by our method are larger in the more challenging VIPriors-10, *e.g.*, 8.2% on MoCo-v2 and 16.7% on SimSiam. It validates the invariance inductive bias learned by the class-wise IRM (in our EQINV) helps to disentangle and alleviate the OOD bias effectively. 4) Results on ImageNet100 dataset show the consistent improvements of EQINV due to the additional supervised contrastive loss, indicating the generalizability of our EQINV in a wide range of cases from insufficient to sufficient data.

Table 2: Test accuracy (%) of DG SOTA methods. V-50/-10 denote VIPriors-50/-10.

		Methods	V-50	V-10	Methods	V-50	V-10
Train from Scratch	Boardline		30.60	4.59	IP-IRM	48.90	26.88
	SD [64]		33.91	4.85	+SD [64]	49.91	28.01
	SelfReg [46]		23.85	3.64	+SelfReg [46]	36.48	22.75
	SagNet [59]		34.92	5.62	+SagNet [59]	47.82	26.17
					+EQINV	52.27	30.36
	Train from SSL						

In Table 2, we compare our EQINV with DG methods. We try both “train from scratch” and “train from SSL” to meet the pre-training requirement of DG. We can find that our EQINV outperforms DG methods with large margins, showing the weaknesses of existing OOD methods for handling *insufficient* data.

In Table 3, we compare our EQINV with the solutions from other competition teams in the challenge with the same comparable setting: no val is used for training, single model w/o ensemble, similar ResNet50/ResNext50 backbones. We can observe the best performance is by our method. It is worth noting that the competitors Zhao *et al.* also used SSL techniques for pretraining. They took the knowledge distillation [38,40] as their downstream learning method. Our EQINV outperforms their model with a large margin.

5.4 Ablation Study

Q1: *What are the effects of different components of EQINV?*

A1: We traversed different combinations of our proposed three steps to evaluate their effectiveness. The results are shown in Table 4. We can draw the following observations: 1) By focusing on the first three rows, we can find that the improvements are relatively marginal without the SSL equivariance pretraining. This is reasonable as the feature similarity cannot reflect the semantics change

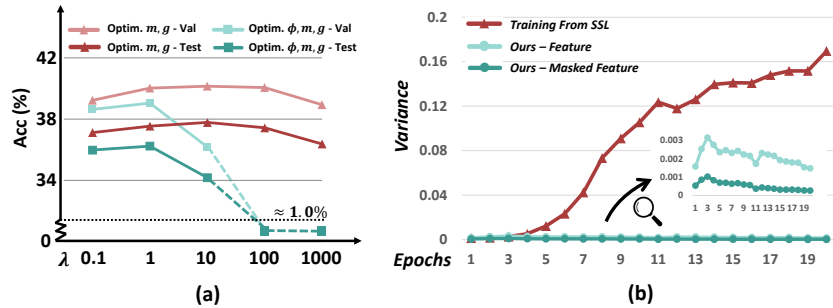


Fig. 6: (a) Accuracies (%) with different optimization schedules and values of λ on the VIPriors-20 `val` and `test` sets. (b) The intra-class feature variance of training from SSL and our EQINV on VIPriors-10 dataset during training process. “Feature” and “Masked Feature” represent $\phi(x)$ and $m \circ \phi(x)$, respectively.

exactly without the equivariance property, thus affects the environments construction (Step 2) and class-wise IRM (Step 3); 2) The comparison between row 4 to 6 indicates the significance of our proposed similarity adjustment (Step 2). It is clear that the vanilla cosine similarity results in clear performance drops due to the inaccurate environment construction.

Table 3: Accuracy (%) comparison with other competition teams (single model w/o ensemble) on the `val` set of VIPriors-50.

Team	Backbone	Val Acc
Official Baseline	ResNet50	33.16
Zhao <i>et al.</i> [85]	ResNet50	44.60
Wang <i>et al.</i> [12]	ResNet50	50.70
Sun <i>et al.</i> [72]	ResNext50	51.82
EQINV (Ours)	ResNet50	54.58

Table 4: Evaluation of the effectiveness of our three steps in EQINV on VIPriors-20.

Components			Val	Test
Step 1	Step 2	Step 3		
✗	✗	✗	13.13	12.39
✗	✗	✓	13.01	12.41
✗	✓	✓	15.69	14.17
✓	✗	✗	37.61	34.88
✓	✗	✓	38.87	36.34
✓	✓	✓	40.15	37.78

Q2: What is the optimal λ for EQINV? Why does not the class-wise IRM penalty term update feature backbone ϕ ?

A2: Recall that we highlight such elaborate design in Section 4 Step 3. In Fig. 6 (a), we evaluate the effect of freezing ϕ for Eq. (6) on VIPriors dataset. First, we can see that setting $\lambda = 10$ with freezing ϕ can achieve the best validation and test results. Second, when increasing λ over 10, we can observe a sharp performance drops for updating ϕ , even down to the random guess ($\approx 1\%$). In contrast, the performances are much more robust with freezing ϕ while varying λ , indicating the non-sensitivity of our EQINV. This validates the adversary effect of the equivariance and invariance. Updating ϕ with large λ would destroy the previously learnt equivariance inductive bias.

Q3: Does our EQINV achieve invariance with the learned environments \mathcal{E} and the proposed class-wise IRM (i.e., Step 3)?

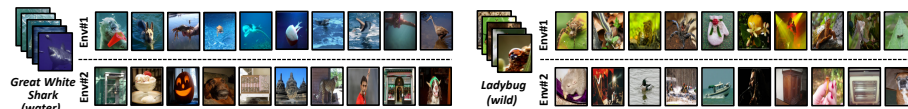


Fig. 7: Visualizations of the top-10 images of generated environments for two classes (*i.e.*, **great white shark** and **ladybug**) on VIPriors-10 dataset. We manually label their main context label (*i.e.*, **water** and **wild**).

A3: In Fig. 6 (b), we calculate the variance of intra-class feature with training from SSL and our EQINV on VIPriors-10 data. It represents the feature divergence within the class. We can find that: 1) Compared to our EQINV, the variance of training from SSL increases dramatically, indicating that equivariant features are still easily biased to environments without invariance regularizations. 2) The masked feature $m \circ \phi(x)$ of our EQINV achieves continuously lower variance than $\phi(x)$, validates the effectiveness of our learnt mask. See Appendix for more visual attention visualizations.

Q4: *What does the cluster look like for real data with the proposed similarity adjustment (*i.e.*, Step 2)?*

A4: Recall that we have displayed the cluster results on a toy Colored MNIST data in Fig. 5 and validated the superiority of our similarity adjustment. Here we wonder how does it perform on real-world data with much comprehensive semantics? We visualize the top-10 images of Env#1 and Env#2 for two random selected classes in Fig. 7. Interestingly, we can find that images of Env#1 mainly share the context (*e.g.*, **water**) with the **anchor** class (*e.g.*, **Great White Shark**). In contrast, images of Env#2 have totally different context. More importantly, the classes distribute almost uniformly in both Env#1 and #2, indicating that our adjusted similarity isolate the effect of the class feature.

6 Conclusion

We pointed out the theoretical reasons why learning from insufficient data is inherently more challenging than sufficient data—the latter will be inevitably biased to the limited environmental diversity. To counter such “bad” bias, we proposed to use two “good” inductive biases: equivariance and invariance, which are implemented as the proposed EQINV algorithm. In particular, we used SSL to achieve the equivariant feature learning that wins back the class feature lost by the “bad” bias, and then proposed a class-wise IRM to remove the “bad” bias. For future work, we plan to further narrow down the performance gap by improving the class-muted clustering to construct more unique environments.

Acknowledgement. The authors would like to thank all reviewers for their constructive suggestions. This research is partly supported by the Alibaba-NTU Joint Research Institute, AISG, A*STAR under its AME YIRG Grant (Project No.A20E6c0101).

References

1. Ahuja, K., Shanmugam, K., Varshney, K., Dhurandhar, A.: Invariant risk minimization games. In: ICML. pp. 145–155. PMLR (2020) [4](#)
2. Ahuja, K., Wang, J., Dhurandhar, A., Shanmugam, K., Varshney, K.R.: Empirical or invariant risk minimization? a sample complexity perspective. arXiv preprint (2020) [7](#)
3. Allen-Zhu, Z., Li, Y.: What can resnet learn efficiently, going beyond kernels? NeurIPS **32** (2019) [4](#)
4. Arjovsky, M., Bottou, L., Gulrajani, I., Lopez-Paz, D.: Invariant risk minimization. arXiv preprint arXiv:1907.02893 (2019) [3](#), [4](#), [6](#), [7](#), [9](#)
5. Austin, P.C.: An introduction to propensity score methods for reducing the effects of confounding in observational studies. Multivariate behavioral research **46**(3), 399–424 (2011) [6](#)
6. Bahng, H., Chun, S., Yun, S., Choo, J., Oh, S.J.: Learning de-biased representations with biased representations. In: ICML. pp. 528–539. PMLR (2020) [6](#)
7. Bardes, A., Ponce, J., LeCun, Y.: Vicreg: Variance-invariance-covariance regularization for self-supervised learning. arXiv preprint arXiv:2105.04906 (2021) [4](#)
8. Barz, B., Brigato, L., Iocchi, L., Denzler, J.: A strong baseline for the vipriors data-efficient image classification challenge. arXiv preprint arXiv:2109.13561 (2021) [10](#)
9. Battaglia, P.W., Hamrick, J.B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., et al.: Relational inductive biases, deep learning, and graph networks. arXiv preprint arXiv:1806.01261 (2018) [4](#)
10. Bietti, A., Mairal, J.: On the inductive bias of neural tangent kernels. NeurIPS **32** (2019) [4](#)
11. Bouchacourt, D., Ibrahim, M., Morcos, A.: Grounding inductive biases in natural images: invariance stems from variations in data. NeurIPS **34** (2021) [4](#)
12. Bruintjes, R.J., Lengyel, A., Rios, M.B., Kayhan, O.S., van Gemert, J.: Vipriors 1: Visual inductive priors for data-efficient deep learning challenges. arXiv preprint arXiv:2103.03768 (2021) [3](#), [7](#), [10](#), [11](#), [13](#)
13. Castro, F.M., Marín-Jiménez, M.J., Guil, N., Schmid, C., Alahari, K.: End-to-end incremental learning. In: ECCV. pp. 233–248 (2018) [4](#)
14. Chen, S., Dobriban, E., Lee, J.: A group-theoretic framework for data augmentation. NeurIPS **33**, 21321–21333 (2020) [4](#)
15. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: ICML. pp. 1597–1607. PMLR (2020) [5](#), [9](#), [11](#)
16. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020) [7](#), [11](#)
17. Chen, X., He, K.: Exploring simple siamese representation learning. arXiv preprint arXiv:2011.10566 (2020) [7](#), [11](#)
18. Chrupała, G.: Symbolic inductive bias for visually grounded learning of spoken language. arXiv preprint arXiv:1812.09244 (2018) [4](#)
19. Cohen, T., Weiler, M., Kicanaoglu, B., Welling, M.: Gauge equivariant convolutional networks and the icosahedral cnn. In: ICML. pp. 1321–1330. PMLR (2019) [4](#)
20. Cohen, T., Welling, M.: Group equivariant convolutional networks. In: ICML. pp. 2990–2999. PMLR (2016) [4](#)

21. Creager, E., Jacobsen, J.H., Zemel, R.: Environment inference for invariant learning. In: ICML. pp. 2189–2200. PMLR (2021) [7](#)
22. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: CVPR Workshops. pp. 702–703 (2020) [4](#), [11](#)
23. Daneshmand, H., Joudaki, A., Bach, F.: Batch normalization orthogonalizes representations in deep random networks. *NeurIPS* **34** (2021) [4](#)
24. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255. Ieee (2009) [10](#)
25. Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A.: Shortcut learning in deep neural networks. *Nature Machine Intelligence* **2**(11), 665–673 (2020) [2](#), [3](#)
26. Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A., Brendel, W.: Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *ICLR* (2018) [2](#)
27. Gondal, M.W., Wuthrich, M., Miladinovic, D., Locatello, F., Breidt, M., Volchokov, V., Akpo, J., Bachem, O., Schölkopf, B., Bauer, S.: On the transfer of inductive bias from simulation to the real world: a new disentanglement dataset. *NeurIPS* **32** (2019) [4](#)
28. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014) [2](#)
29. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377* (2021) [7](#)
30. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722* (2019) [5](#), [7](#)
31. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016) [5](#)
32. He, Y., Shen, Z., Cui, P.: Towards non-iid image classification: A dataset and baselines. *Pattern Recognition* **110**, 107383 (2021) [2](#), [3](#), [6](#), [10](#), [11](#)
33. Helmbold, D.P., Long, P.M.: On the inductive bias of dropout. *The Journal of Machine Learning Research* **16**(1), 3403–3454 (2015) [4](#)
34. Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. *ICLR* (2019) [4](#)
35. Hendrycks, D., Liu, X., Wallace, E., Dzierdzic, A., Krishnan, R., Song, D.: Pretrained transformers improve out-of-distribution robustness. *arXiv preprint arXiv:2004.06100* (2020) [3](#)
36. Hendrycks, D., Mazeika, M., Kadavath, S., Song, D.: Using self-supervised learning can improve model robustness and uncertainty. *NeurIPS* **32** (2019) [3](#)
37. Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., Song, D.: Natural adversarial examples. In: CVPR. pp. 15262–15271 (2021) [2](#)
38. Heo, B., Kim, J., Yun, S., Park, H., Kwak, N., Choi, J.Y.: A comprehensive overhaul of feature distillation. In: ICCV. pp. 1921–1930 (2019) [12](#)
39. Hernán, M.A., Robins, J.M.: Causal inference (2010) [6](#)
40. Hinton, G., Vinyals, O., Dean, J., et al.: Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* **2**(7) (2015) [12](#)
41. Imbens, G.W., Rubin, D.B.: Causal inference in statistics, social, and biomedical sciences. Cambridge University Press (2015) [6](#)
42. Jo, Y., Chun, S.Y., Choi, J.: Rethinking deep image prior for denoising. In: ICCV. pp. 5087–5096 (2021) [4](#)
43. Jung, Y., Tian, J., Bareinboim, E.: Learning causal effects via weighted empirical risk minimization. *NeurIPS* **33**, 12697–12709 (2020) [4](#)

44. Kayhan, O.S., Gemert, J.C.v.: On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In: CVPR. pp. 14274–14285 (2020) [4](#)
45. Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D.: Supervised contrastive learning. *NeurIPS* **33**, 18661–18673 (2020) [9](#)
46. Kim, D., Yoo, Y., Park, S., Kim, J., Lee, J.: Selfreg: Self-supervised contrastive regularization for domain generalization. In: ICCV. pp. 9619–9628 (2021) [11](#), [12](#)
47. Krueger, D., Caballero, E., Jacobsen, J.H., Zhang, A., Binas, J., Zhang, D., Priol, R.L., Courville, A.: Out-of-distribution generalization via risk extrapolation (rex). *arXiv preprint* (2020) [4](#), [6](#), [9](#)
48. Lahiri, A., Kwatra, V., Frueh, C., Lewis, J., Bregler, C.: Lipsync3d: Data-efficient learning of personalized 3d talking faces from video using pose and lighting normalization. In: CVPR. pp. 2755–2764 (2021) [4](#)
49. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *nature* **521**(7553), 436–444 (2015) [2](#)
50. LeCun, Y., Cortes, C., Burges, C.: Mnist handwritten digit database (2010) [8](#)
51. Lee, J., Kim, E., Lee, J., Lee, J., Choo, J.: Learning debiased representation via disentangled feature augmentation. *NeurIPS* **34** (2021) [6](#), [11](#)
52. Lenssen, J.E., Fey, M., Libuschewski, P.: Group equivariant capsule networks. *NeurIPS* **31** (2018) [4](#)
53. Little, R.J., Rubin, D.B.: Statistical analysis with missing data, vol. 793. John Wiley & Sons (2019) [6](#)
54. Liu, Q., Mohamadabadi, B.B., El-Khamy, M., Lee, J.: Diversification is all you need: Towards data efficient image understanding (2020) [10](#)
55. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017) [4](#)
56. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. *Journal of machine learning research* **9**(11) (2008) [5](#), [6](#)
57. Mitchell, T.M.: The need for biases in learning generalizations. Department of Computer Science, Laboratory for Computer Science Research ... (1980) [4](#)
58. Müller, R., Kornblith, S., Hinton, G.E.: When does label smoothing help? *NeurIPS* **32** (2019) [11](#)
59. Nam, H., Lee, H., Park, J., Yoon, W., Yoo, D.: Reducing domain gap by reducing style bias. In: CVPR. pp. 8690–8699 (2021) [11](#), [12](#)
60. Nam, J., Cha, H., Ahn, S., Lee, J., Shin, J.: Learning from failure: De-biasing classifier from biased classifier. *NeurIPS* **33**, 20673–20684 (2020) [6](#), [8](#), [9](#), [11](#)
61. Van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. *arXiv e-prints* pp. arXiv-1807 (2018) [5](#)
62. Pan, S.J., Tsang, I.W., Kwok, J.T., Yang, Q.: Domain adaptation via transfer component analysis. *IEEE transactions on neural networks* **22**(2), 199–210 (2010) [4](#)
63. Pearl, J.: Causality. Cambridge university press (2009) [4](#)
64. Pezeshki, M., Kaba, O., Bengio, Y., Courville, A.C., Precup, D., Lajoie, G.: Gradient starvation: A learning proclivity in neural networks. *NeurIPS* **34**, 1256–1272 (2021) [11](#), [12](#)
65. Pfister, N., Bühlmann, P., Peters, J.: Invariant causal prediction for sequential data. *Journal of the American Statistical Association* **114**(527), 1264–1276 (2019) [3](#), [4](#)
66. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. *Image* **2**, T2 [5](#)

67. Recht, B., Roelofs, R., Schmidt, L., Shankar, V.: Do imagenet classifiers generalize to imagenet? In: ICML. pp. 5389–5400. PMLR (2019) [2](#), [4](#)
68. Saito, K., Kim, D., Sclaroff, S., Saenko, K.: Universal domain adaptation through self supervision. *NeurIPS* **33**, 16282–16292 (2020) [3](#)
69. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017) [2](#)
70. Shen, Z., Liu, J., He, Y., Zhang, X., Xu, R., Yu, H., Cui, P.: Towards out-of-distribution generalization: A survey. arXiv preprint arXiv:2108.13624 (2021) [2](#)
71. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. *NeurIPS* **30** (2017) [2](#), [4](#)
72. Sun, P., Jin, X., Su, W., He, Y., Xue, H., Lu, Q.: A visual inductive priors framework for data-efficient image classification. In: ECCV. pp. 511–520. Springer (2020) [13](#)
73. Sun, Q., Liu, Y., Chua, T.S., Schiele, B.: Meta-transfer learning for few-shot learning. In: CVPR. pp. 403–412 (2019) [2](#), [4](#)
74. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. In: CVPR. pp. 1199–1208 (2018) [2](#), [4](#)
75. Tian, Y., Krishnan, D., Isola, P.: Contrastive multiview coding. arXiv preprint arXiv:1906.05849 (2019) [3](#), [10](#), [11](#)
76. Vapnik, V.: Principles of risk minimization for learning theory. In: *NeurIPS* (1992) [1](#), [4](#)
77. Wang, T., Yue, Z., Huang, J., Sun, Q., Zhang, H.: Self-supervised learning disentangled group representation as feature. In: Conference and Workshop on Neural Information Processing Systems (*NeurIPS*) (2021) [4](#), [5](#), [7](#), [10](#), [11](#)
78. Wang, T., Zhou, C., Sun, Q., Zhang, H.: Causal attention for unbiased visual recognition. In: ICCV. pp. 3091–3100 (2021) [3](#), [4](#), [10](#), [11](#)
79. Wen, Z., Li, Y.: Toward understanding the feature learning process of self-supervised contrastive learning. In: ICML. pp. 11112–11122. PMLR (2021) [3](#)
80. Xu, Y., Zhang, Q., Zhang, J., Tao, D.: Vitae: Vision transformer advanced by exploring intrinsic inductive bias. *NeurIPS* **34** (2021) [4](#)
81. You, K., Long, M., Cao, Z., Wang, J., Jordan, M.I.: Universal domain adaptation. In: CVPR. pp. 2720–2729 (2019) [4](#)
82. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: ECCV. pp. 818–833. Springer (2014) [2](#)
83. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017) [11](#)
84. Zhang, X., Zhou, L., Xu, R., Cui, P., Shen, Z., Liu, H.: Towards unsupervised domain generalization. In: CVPR. pp. 4910–4920 (2022) [4](#)
85. Zhao, B., Wen, X.: Distilling visual priors from self-supervised learning. In: ECCV. pp. 422–429. Springer (2020) [13](#)
86. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR. pp. 2921–2929 (2016) [2](#)
87. Zhu, F., Cheng, Z., Zhang, X.y., Liu, C.l.: Class-incremental learning via dual augmentation. *NeurIPS* **34** (2021) [4](#)