Learning Debiased Representations via Conditional Attribute Interpolation

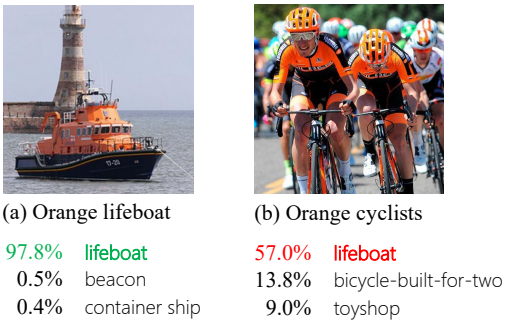
Yi-Kai Zhang, Qi-Wei Wang, De-Chuan Zhan, Han-Jia Ye✉
State Key Laboratory for Novel Software Technology, Nanjing University
{zhangyk,wangqiwei,zhandc,yehj}@lamda.nju.edu.cn

Abstract

An image is usually described by more than one attribute like "shape" and "color". When a dataset is biased, i.e., most samples have attributes spuriously correlated with the target label, a Deep Neural Network (DNN) is prone to make predictions by the "unintended" attribute, especially if it is easier to learn. To improve the generalization ability when training on such a biased dataset, we propose a χ^2-model to learn debiased representations. First, we design a χ-shape pattern to match the training dynamics of a DNN and find Intermediate Attribute Samples (IASs) — samples near the attribute decision boundaries, which indicate how the value of an attribute changes from one extreme to another. Then we rectify the representation with a χ-structured metric learning objective. Conditional interpolation among IASs eliminates the negative effect of peripheral attributes and facilitates retaining the intra-class compactness. Experiments show that χ^2-model learns debiased representation effectively and achieves remarkable improvements on various datasets. Code is available at: https://github.com/ZhangYikaii/chi-square

1. Introduction

Deep neural networks (DNNs) have emerged as an epoch-making technology in various machine learning tasks with impressive performance [5,26]. In some real applications, an object may possess multiple attributes, and some of them are only spuriously correlated to the target label. For example, in Figure 1, the intrinsic attribute of an image annotated by "lifeboats" is its *shape*. Although there are many lifeboats colored orange, a learner can not make predictions through the *color*, *i.e.*, there is a misleading correlation from attribute as *one containing "orange" color is the target "lifeboats"*. When the major training samples can be well discerned by such peripheral attribute, especially learning on it is easier than on the intrinsic one, a DNN is prone to *bias* towards that "unintended" bias attribute [6, 11, 21, 43, 47, 48, 51], like recognizing a "cyclist" wearing orange as a "lifeboat". Similar spurious attribute also exists in various applications



(a) Orange lifeboat (b) Orange cyclists

Figure 1. Classification of a standard ResNet-50 of **(a)** an orange lifeboat in the training set (with both *color* and *shape* attributes), and **(b)** an orange cyclist for the test (aligned with *color* attribute but conflicting with the *shape* one). Most of the lifeboats in the training set are orange. The biased model is prone to predict via the "unintended" *color* attribute rather than the intrinsic *shape*.

such as recommendation system [8, 35, 53, 59] and neural language processing [13, 14, 33, 41, 56].

Given such a biased training dataset, how to get rid of the negative effect of the misleading correlations? One intuitive solution is to perform special operations on those samples highly correlated to the bias attributes, which requires additional supervision, such as the pre-defined bias type [1, 4, 11, 12, 22, 30, 34, 46, 50]. Since prior knowledge of the dataset bias requires expensive manual annotations and is naturally missing in some applications, learning a debiased model without additional supervision about bias is in demand. Nam *et al*. [36] identify samples with intrinsic attributes based on the observation that malignant bias attributes are often easier-to-learn than others. Then the valuable samples for a debiasing scheme could be dynamically reweighted or augmented [11,27,34]. However, the restricted number of such samples implies uncertain representations and limits its ability to assist in debiasing.

To leverage more valuable-for-debiasing knowledge, we take a further step in analyzing the representation space of naïvely-training dynamics, especially focusing on the discrepancies in attributes with a different learning difficulty. As we will later illustrate in Figure 2, an attribute-based

DNN pushes and fits on the easier bias attribute initially. The intrinsic attribute is then forced to shift in a "lazy" manner. The bias attribute that is pushed away first leaves a large margin boundary. Since the space of the other intrinsic attribute is filled with many different samples on bias attribute, it has a large intra-class variance, like a "hollow". The representation is biased toward one side of the "hollow", *i.e.*, those samples aligned with the bias attribute. Without the true intra-class structure, the model becomes biased.

From the above observation, it is crucial to fill intra-class "hollow" and remodel representation compactness. Notice that the samples shifting to the two sides of the "hollow" have different characteristics, as aligned with the bias attribute and conflicting with it, respectively. We can find samples with an intermediate state between the above two. We call this type of sample the *Intermediate* Attribute Samples (IASs) which are near the decision boundary. When we *condition* (fix) on the intrinsic attribute, IASs vary on the other bias attribute and are exactly located in the "hollow" with low-density structural knowledge. Further, we can mine samples, including IASs, based on the distinct training dynamics.

To this end, we propose our two-stage χ^2-model. In the first stage, we train a vanilla model on the biased dataset and record the sample-wise training dynamics w.r.t. both the target and the most obvious non-target classes (as the bias ones) along the epochs. An IAS is often predicted as a non-target class in the beginning and then switched to its target class gradually, making its dynamics plot a χ-shape. Following this observation, we design a χ-shape pattern to match the training samples. The matching score ranks the mined samples according to the bias level, *i.e.*, how much they are biased towards the side of the bias attribute. Benefiting from the IASs, we conduct conditional attribute interpolation, *i.e.*, fixing the value of the target attribute. We interpolate the class-specific prototypes around IASs with various bias ratios. These conditional interpolated prototypes precisely "average out" on the bias attribute. From that, we design a χ-structured metric learning objective. It pulls samples close to those same-class interpolated prototypes, then intra-class samples become compact, and the influence of the bias attribute is removed. Our χ^2-model learns debiased representation effectively and achieves remarkable improvements on various datasets. Our contributions are summarized as

- We claim and verify that Intermediate Attribute Samples (IASs) distributed around attribute decision boundaries facilitate learning a debiased representation.
- Based on the diverse learning behavior of different attribute types, we mine samples with varying bias levels, especially IASs. From that, we interpolate bias attribute conditioned on the intrinsic one and compact intra-class samples to remove the negative effect of bias.
- Experiments on benchmarks and a newly constructed real-world dataset from NICO [17] validate the effectiveness of our χ^2-model in learning debiased representations.

2. A Closer Look at Learning on Bias Attribute

After the background of learning on a biased dataset in subsection, we analyze the training dynamics of the model.

2.1. Problem Definition

Given a training set $\mathcal{D}_{\text{train}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, each sample $\mathbf{x}_i$ is associated with a class label $y \in \{1, 2, \cdots, C\}$. We aim to find a decision rule $h_{\boldsymbol{\theta}}$ that maps a sample to its label. $h_{\boldsymbol{\theta}}$ is optimized by fitting all the training samples, *e.g.*, minimizing the cross entropy loss as follows:

$$\mathcal{L}_{\text{CE}} = \mathbb{E}_{(\mathbf{x}_i, y_i)\sim\mathcal{D}_{\text{train}}} \left[-\log \Pr \left(h_{\boldsymbol{\theta}}(\mathbf{x}_i) = y_i \mid \mathbf{x}_i \right) \right] . \quad (1)$$

We denote $h_{\boldsymbol{\theta}} = \arg\max_{c\in[C]} \mathbf{w}_c^\top f_{\boldsymbol{\phi}}(\mathbf{x})$, where $f_{\boldsymbol{\phi}} \to \mathbb{R}^d$ is the feature extraction network and $\{\mathbf{w}_c\}_{c\in[C]}$ is top-layer C-class classifier. The $\boldsymbol{\theta}$ represents the union of learnable parameters $\boldsymbol{\phi}$ and $\mathbf{w}$. We expect the learned $h_{\boldsymbol{\theta}}$ to have the high discerning ability over the test set $\mathcal{D}_{\text{test}}$ which has the same form as the training set $\mathcal{D}_{\text{train}}$.

In addition to its class label, a sample could be described based on various attributes. If an attribute is spuriously correlated with the target label, we name it *non-target bias* attribute a_b. The attribute that intrinsically determines the class label is the *target* attribute a_y. For example, when we draw different handwritten digits in the MNIST dataset with specific colors [22], the *color* attribute will not help in the model generalization since we need to discern digits by the *shape*, *e.g.*, "1" is like the stick. However, if almost all training images labeled "1" are in the same "yellow" color, the decision rule *image in "yellow" is digit "1"* will perform well on a such biased training set.

In the task of learning with a biased training set [22, 29, 36], the bias attribute a_b on most of the same-class samples are consistent, and spuriously correlated with the target label (as example digit "1" in "yellow" above), so a model $h_{\boldsymbol{\theta}}$ that relies on a_b or the target attribute a_y will both perform well on $\mathcal{D}_{\text{train}}$. In real-world applications, it is often easier to learn to rely on a_b than on a_y, such as "background" or "texture" is easier-to-learn than the object [42, 51]. Therefore a model is prone to recognize based on the a_b. Such a *simplicity bias* [3, 37, 38, 42] dramatically hurts the generalization of an unbiased test set. Nam *et al*. [36] also observe that the loss dynamics indicate the easier a_b is learned first, where the model is distracted and fails to learn a_y.

Based on the behaviors of the "ultimate" biased model, samples in $\mathcal{D}_{\text{train}}$ are split into two sets. Those training samples that could be correctly predicted based on the bias attribute a_b are named as *Bias-Aligned* (BA) samples (as example "yellow digit 1" above), while the remaining ones are *Bias-Conflicting* (BC) samples (as digit "1" of other colors). The number of BC samples is extremely small, and previous

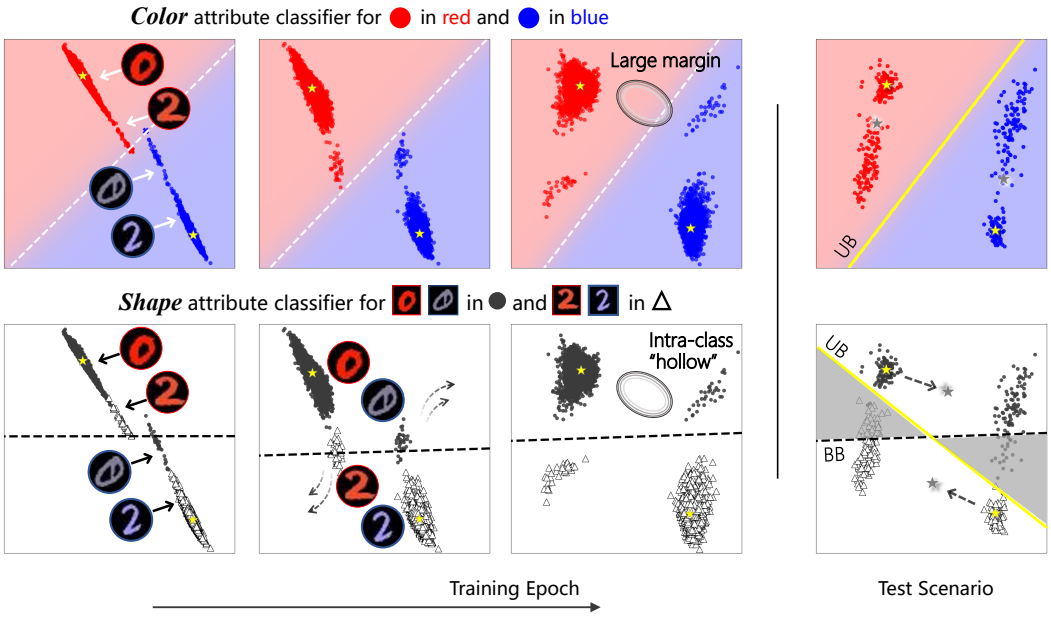


Figure 2. **An illustration of the training dynamics of a naïvely-trained model on a biased dataset.** The different attribute classes are drawn with a specific *color* (top row) or *shape* (bottom row). The first three columns correspond to the sequential training progress of these two classifiers, and the final column shows the test scenario. The easier-to-learn *color* is fitted first, which leaves a large margin on decision boundary and correspondingly triggers the *shape* attribute intra-class "hollow". "UB" and "BB" are the abbreviations of "unbiased boundary" and "biased boundary". The biased model cannot generalize well as the shadow area in the last column frames the *shape*-misclassified samples. We use yellow stars to indicate the shifted class centers of the training samples and gray stars for those of the test samples.

methods emphasize their role with various strategies [11, 27, 34, 36]. For additional related methods of learning a debiased model [2, 7, 9, 10, 18, 24, 25, 29, 31, 40, 49, 60] please see the supplementary material. They can be extrapolated and applied to other application scenarios [16, 45, 52, 54, 57, 58].

2.2. The Training Dynamics of Biased Dataset

We analyze the training dynamics of a naïvely trained model in Eq. 1 on the Colored MNIST dataset. The non-target bias attribute is the *color* and the target attribute is the *shape*. For visualization shown in Figure 2, we set the output dimension of the penultimate layer as two. In addition to the learned classifier on shape attribute a_y , simultaneously, we add another linear classifier on top of the embedding to show how the decision boundary of color attribute a_b changes. More details are described in the supplementary material. Focusing on the precedence relationship for learning a_y and a_b, we have the following observations:

- **The easier-to-learn bias attribute *color* is fitted soon.** The early training stage is shown in the first column. Both color and shape attribute classifiers discern by different colors and do correctly on almost all BA samples (red "0" and blue "2", *about* 95% *of the training set*).
- **The target attribute *shape* is learned later in a "lazy" manner.** To further fit all shape labels, the model focuses on the limited BC samples (blue "0" and red "2", *correspondingly about* 5%) that cannot be perfectly classified by color. It pushes minor BC representations to the other (correct) side instead of adjusting the decision boundary.
- The ahead-*color* and lagged-*shape* learning process leaves **a large margin of *color* attribute boundary**, which further triggers **the *shape* attribute intra-class "hollow"**. Because the representation of different colors is continuously pushed away (classified) before that of the shape, the gaps between different color attribute clusters are significantly larger than that of the shape attribute.
- Since there is an intra-class "hollow" between BA and BC samples which is conditioned on a particular *shape*, **the true class representation is deviated toward *color***. The fourth column shows that the training class centers (yellow stars) and the test ones (gray stars) are mismatched. The true class center is located in the low-density "hollow" between shape-conditioned BA and BC samples.

Previous observations indicate that the before-and-latter learning process on attributes of different learning difficulties leaves the model to lose intra-class compactness, primarily when learning relying on the bias attribute is easier. To alleviate class center deviation towards the BA samples, only emphasizing the BC samples is insufficient due to their scarcity. In addition, we propose to utilize Intermediate Attribute Samples (IASs), *i.e.*, the samples near the attribute decision boundary and remodel the shifted representation.

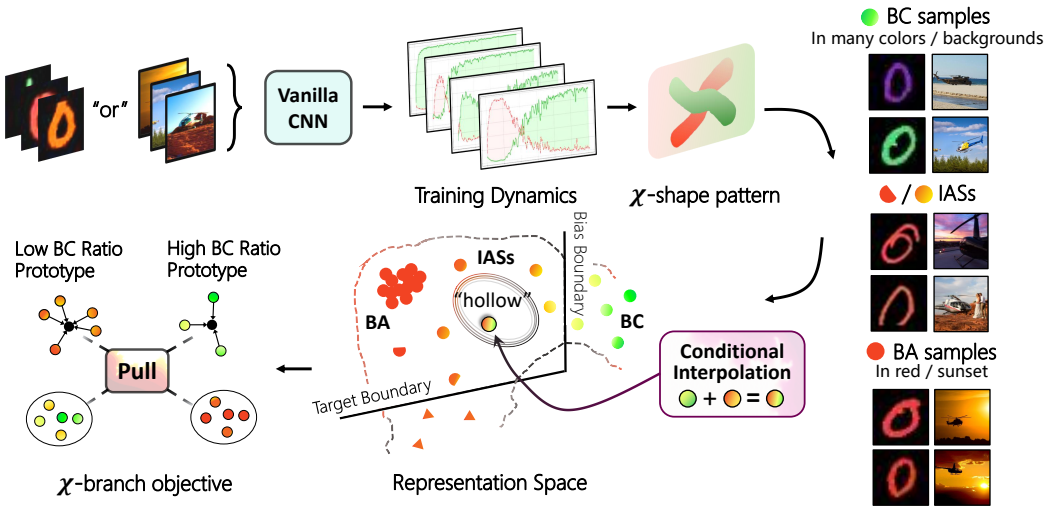


Figure 3. **An illustration of the χ^2-model.** In the first stage (top row, *left to right*), we match and mine all samples with a χ-shape pattern. As shown in the right part, the images are getting biased towards the side of peripheral bias attribute from top to bottom, *i.e.*, from various *colors* or *backgrounds* to a single *red* or *sunset*. Different shapes (○ or Δ) and colors (orange or green) drawn in the *Representation Space* indicate the target and bias attribute, respectively. In the second stage (bottom row, *right to left*), we construct prototypes by conditional interpolating around IASs with various ratios and design the χ-structured metric learning objective to pull the intra-class samples.

Especially when conditioned on the target attribute, the IASs vary on bias attribute and fill in the low-density intra-class "hollow" between BA and BC samples.

3. χ^2-Model

To mitigate the representation deviation and compact the intra-class "hollow", we leverage IASs to encode how bias attribute changes from one extreme (major BA side) to another (BC side). Then, the variety of the bias attribute could be interpolated when conditioned on a particular target attribute. We propose our two-stage χ^2-model, whose notion is illustrated in Figure 3. First, the χ^2-model discovers IASs based on the training dynamics of the vanilla model in subsection 3.1. Next, we analyze where the top-ranked samples with a χ-shape pattern are as well as their effectiveness in debiasing in subsection 3.2. A conditional attribute interpolation step with IASs then fills in the low-density "hollow" to get a better estimation of class-specific prototype. By pulling samples to the corresponding prototype, the χ-structured metric learning makes intra-class samples compact in subsection 3.3. We further investigate the Colored MNIST dataset. Results on other datasets are consistent.

3.1. Scoring Samples with a χ-shape Pattern

From the observations in the previous section, we aim to collect IASs to reveal how BC samples shift and leave the intra-class "hollow" between them and BA ones. As discussed in subsection 2.2, the vanilla model fits BC samples later than BA ones, which motivates us to score the samples from their training dynamics. Once we have the score pattern to match and distinguish BA and BC samples, IASs, with intermediate scores can be extracted and available for the next debiasing stage. In the following, we denote the posterior of the Ground-Truth class (GT-class) y_i for a sample $\mathbf{x}_i$ as

$$\Pr\left(h_{\boldsymbol{\theta}}(\mathbf{x}_i) = y_i \mid \mathbf{x}_i\right) = \mathrm{softmax}(\mathbf{w}_c^\top f_{\boldsymbol{\phi}}(\mathbf{x}))_{y_i}\ , \qquad (2)$$

the larger the posterior, the more confident a model predicts $\mathbf{x}_i$ with y_i. For notation simplicity, we abbreviate the posterior as $\Pr\left(y_i \mid \mathbf{x}_i\right)$. The target posterior of a BA sample reaches one or becomes much higher than other categories soon after training several epochs, while the posterior of a BC sample has a delayed increase. To capture the clues on the change of bias attribute, we also analyze the posterior of the *most obvious non-GT* attribute, which reveals crucial bias influences. Denote the model at the t-th epoch plus the superscript t, such as $h_{\boldsymbol{\theta}}^t$. we take the bias class for the sample $\mathbf{x}_i$ at epoch t as $b_i^t = \arg\max_{c\in[C], c\neq y_i} \left(\mathbf{w}_c^\top f_{\boldsymbol{\phi}}(\mathbf{x})\right)^t$. Then, we define the non-GT bias class as the most frequent b_i^t along all epochs, *i.e.*, $b_i = \mathrm{max\_freq}\{b_i^t\}_{t=1}^T$. A sample has a larger bias class posterior when it has low confidence in its target class and vice versa.

Taking posteriors of both y_i and b_i into account, a BA sample has larger $\Pr\left(y_i \mid \mathbf{x}_i\right)$ and small $\Pr\left(b_i \mid \mathbf{x}_i\right)$ along all its training epochs. For a BC sample, $\Pr\left(y_i \mid \mathbf{x}_i\right)$ increases gradually and meanwhile $\Pr\left(b_i \mid \mathbf{x}_i\right)$ decreases. We verify the phenomenon on Colored MNIST dataset in Figure 4 (left). For BA samples (yellow "1"), the two curves demonstrate a "rectangle", while for BC samples (blue "1"), the two curves have an obvious intersection and reveal a "χ" shape. The statistics for the change of posteriors are shown

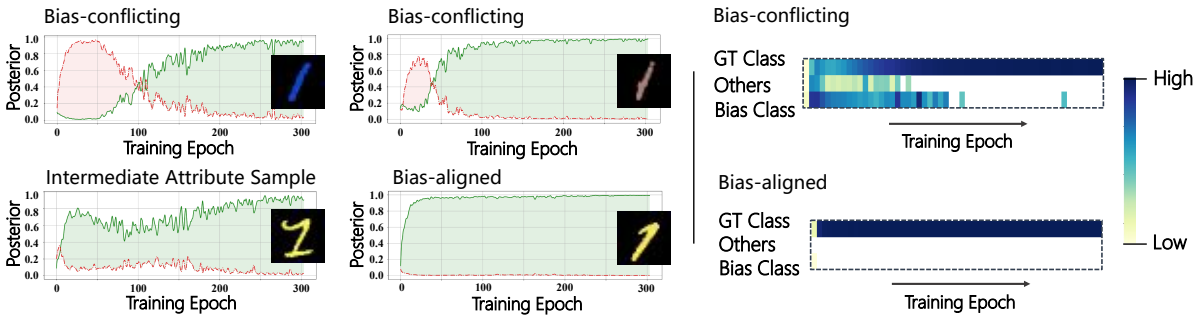


Figure 4. **Left**: The change of posterior over the GT-class (green curve) as well as the bias class (red curve) over four samples. The two curves over BC samples and IAS demonstrate a χ-shape, which is different from the curves over the BA sample. **Right**: The change of prediction frequencies of BC or BA samples along with the training epochs. The statistics are calculated over all BC or BA samples. A BA sample is easily predicted as the GT-class from the initial training stage, while a BC sample changes its prediction from the bias class to the GT-class gradually.

in Figure 4 (right). Therefore, how much the training dynamics match the "χ" shape reveals the probability of a sample that shifts from the major BA clusters to minor BC ones. We design a χ-shape for the dynamics of losses to capture such BC-specific properties. The change of sample-specific loss for ground-truth label and bias label over T epochs could be summarized by $\mathcal{L}_{\text{CE}}$. Then, we use two exponential χ-shape functions χ_{pattern} to capture the ideal loss shape of the BC sample, *i.e.*, the severely shifted case.

$$\mathcal{L}_{\text{CE}}\left(\mathbf{x}_i\right) = \begin{pmatrix} \mathcal{L}^{gt}_{\text{CE}}\left(\mathbf{x}_i\right) = \left\{-\log \Pr^t\left(y_i \mid \mathbf{x}_i\right)\right\}_{t=1}^{T} \\ \mathcal{L}^{b}_{\text{CE}}\left(\mathbf{x}_i\right) = \left\{-\log \Pr^t\left(b_i \mid \mathbf{x}_i\right)\right\}_{t=1}^{T} \end{pmatrix},$$
$$\chi_{\text{pattern}} = \begin{pmatrix} \mathrm{p}^{gt} = \left\{e^{-A_1 t}\right\}_{t=1}^{T} \\ \mathrm{p}^{b} = \left\{e^{A_2 t}\right\}_{t=1}^{T} \end{pmatrix}, \tag{3}$$

where A_1 and A_2 are the matching factors. They could be determined based on the dynamics of prediction fluctuations. For more details please see the supplementary material. The χ_{pattern} encodes the observations for the most deviated BC samples. To match the loss dynamics with the pattern, we use the inner product over the two curves:

$$\begin{aligned} \mathbf{s}(\mathbf{x}_i) &= \left\langle \mathcal{L}_{\text{CE}}\left(\mathbf{x}_i\right), \chi_{\text{shape}} \right\rangle \\ &= \left\langle \mathcal{L}^{gt}_{\text{CE}}\left(\mathbf{x}_i\right), \mathrm{p}^{gt} \right\rangle + \left\langle \mathcal{L}^{b}_{\text{CE}}\left(\mathbf{x}_i\right), \mathrm{p}^{b} \right\rangle \\ &= \sum_{t=1}^{T} \left(-e^{-A_1 t} \cdot \log \Pr\left(h_{\boldsymbol{\theta}}\left(\mathbf{x}_i\right) = y_i \mid \mathbf{x}_i\right) \right. \\ &\quad \left. -e^{A_2 t} \cdot \log \Pr\left(h_{\boldsymbol{\theta}}\left(\mathbf{x}_i\right) = b_i \mid \mathbf{x}_i\right)\right) . \end{aligned} \tag{4}$$

The inner product $\mathbf{s}(\mathbf{x}_i)$ takes the area under the curves (AUC) into account, which is more robust w.r.t. the volatile loss changes. When $\mathbf{s}(\mathbf{x}_i)$ score goes from low to high, the sample varies on the bias level, *i.e.*, from BA samples to IASs, and then to BC samples.

Table 1. The classification accuracy on the unbiased test sets of vanilla models. Various training sampling strategies are compared. "0-1" denotes only using BC samples. "Step-wise" denotes applying uniformly higher and lower weights on BC and BA samples. "χ-pattern" denotes sampling with scores calculated by our χ^2-model. The best results are in bold, while the second-best ones are with underlines. C-CIFAR-10 is a similarly biased dataset as C-MNIST.

Dataset	**C-MNIST**		**C-CIFAR-10**	
Ratio (%)	99.9	99.5	99.9	99.5
Vanilla	28.58	59.29	26.91	30.16
+ 0-1	**54.78**	70.41	18.73	25.06
+ Step-wise	41.68	<u>73.52</u>	<u>32.12</u>	<u>35.91</u>
+χ-pattern	<u>52.67</u>	**80.29**	**35.47**	**37.83**

3.2. Where IASs Are and Why IASs Can Help to Learn a Debiased Representation?

Combining the analysis in subsection 2.2 and collecting ranked samples by $\mathbf{s}(\mathbf{x}_i)$, we find there are two types of IASs according to the representation near the target attribute decision boundary (as "0" for complex shapes in Figure 3), or that of the bias attribute (as helicopter in intermediate transitional "sunset" background). **(1)** If an IAS has an intermediate target attribute value, it may be a difficult samples and contains rich information about the target class boundaries. **(2)** If an IAS is in an intermediate state on the bias attribute, it may help to fill in the intra-class vacant "hollow" when conditioning (fixing) on the target attribute. Both types of IASs are similar to BC samples but from two directions, *i.e.*, compared to the BA samples, they contain richer semantics on target or bias attributes. In the representation space, they are scattered between BA and BC samples, compensating for the sparsity of BC samples and valuable for debiasing. We will show how χ-structured objective with IASs help to remodel the true class centers in the following subsection.

We illustrate the importance of IASs with simple experi-

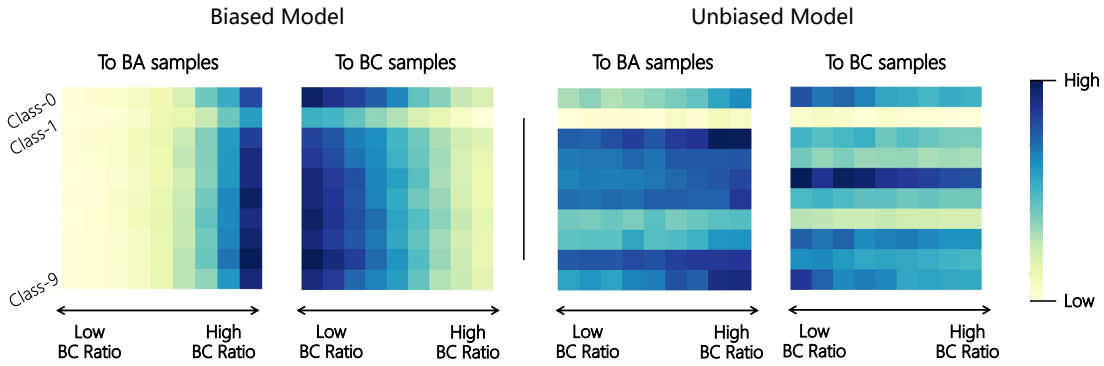


Figure 5. **Heatmap of the mean distance from the sample to its conditional interpolated prototypes.** We construct the prototype with mixing over the same-class subset but with different ratios of BC samples. Then the mean distances between the sample and those prototypes are measured. For a biased model (left two), when the BC ratio γ of interpolated prototype $\mathbf{p}_\gamma$ changes from low to high (horizontal direction), the distance of a sample to different $\mathbf{p}_\gamma$ varies hugely. The BA samples are closer to the low ratio ones, while the BC samples behave the opposite. An intra-class "hollow" exists. For an unbiased model (right two), the distances from any sample to $\mathbf{p}_\gamma$ with different ratios are almost the same.

ments on biased Colored-MNIST and Corrupted CIFAR-10 datasets. The datasets are described in subsection 4.1. We investigate whether various reweighting strategies on the vanilla model improve the generalization ability over an unbiased test set. We use "0-1" to denote the strategy that utilizes only the BC samples. "step-wise" means we apply uniformly higher (ratio of BA samples) and lower weights (one minus above ratio) to BC and BA samples. Our "χ-pattern" smoothly reweights all samples with the matched scores, where BC samples as well as IASs have relatively larger weights than the remaining BA ones. The results in Table 1 shows that simple reweighting strategies can improve the performance of a vanilla classifier, supporting the significance of emphasizing BC-like samples. Our "χ-pattern" gets the best results in most scenarios, indicating that higher resampling weights on the IASs and BC samples assist the vanilla model to better frame the representation space.

3.3. Learning from a χ-Structured Objective

Although the BA samples are severely biased towards the bias attribute, the BC samples, integrating the rich bias attribute semantics, naturally make the representation independent of the biased influence [19]. An intuitive approach for debiasing is to average over BC samples and classified by the BC class centers. However, the sparsity of BC samples induces an erratic estimation which is far from the true class center, as shown in Figure 2.

Benefiting from the analysis that BC-like IASs better estimate the intra-class structure, we target conditional interpolating around it, *i.e.*, mixing the same-class samples with different BC-like scores to remodel the intermediate samples between BA and BC samples. From that, we can construct many prototypes closer to the real class center and pull samples to these prototypes to compact the intra-class space. Combined with the soft ranking score from the χ-pattern in the previous stage, we build two pools (subsets) of the samples denoted as $\mathcal{D}_{\parallel}$ and $\mathcal{D}_{\perp}$. The $\mathcal{D}_{\perp}$ pool collects the top-rank sampels and most of them are BC samples and IASs. The $\mathcal{D}_{\parallel}$ pool is sampled from the remaining (BA) part according to the score. With the help of $\mathcal{D}_{\parallel}$ and $\mathcal{D}_{\perp}$, we construct multiple *bias bags* (subset) $\mathcal{B}_\gamma$ with bootstrapping where the *ratio* of BC samples is γ.

$$\mathcal{B}_\gamma = \left\{ (\mathbf{x}_i, y_i) \,\middle|\, \mathrm{num}\left(\mathcal{D}_{\perp}\right) : \mathrm{num}\left(\mathcal{D}_{\parallel}\right) = \gamma \right\} \ , \tag{5}$$

where $\mathrm{num}\left(\mathcal{D}\right)$ equals the number of samples in $\mathcal{D}$. When γ is low to high, the $\mathcal{B}_\gamma$ contains samples ranging from the extremes of BA samples to the IASs, and then to the BC ones. Based on $\mathcal{B}_\gamma$, we compute the prototype, *i.e.*, averaged on $\mathcal{B}_\gamma$ to interpolate bias attribute conditioned on the particular target attribute. For example, the prototype conditioned on class c is formalized as $\mathbf{p}_{\gamma,c}$:

$$\mathbf{p}_{\gamma,c} = \frac{1}{K} \sum_{(\mathbf{x}_i, y_i) \in \mathcal{B}_\gamma} f_\phi\left(\mathbf{x}_i\right) \cdot \mathbb{I}\left[y_i = c\right] \ . \tag{6}$$

To further demonstrate the significance of intra-class compactness, we design the experiments to study the difference between a biased vanilla model and the unbiased oracle model (well-trained on an unbiased training set). We measure the mean distance between samples and their multiple conditional interpolated prototypes with changing ratio γ. If the prototypes are shifted with changing γ, that indicates a large intra-class deviation exists. As shown in Figure 5, for a biased model, when γ decreases, $\mathbf{p}_\gamma$ is interpolated closer to the BA samples. Opposite phenomena are observed in the BC samples. As for the unbiased oracle model, no matter how the BC ratio γ changes, such mean distance is almost unchanged and shows a lower variance. This coincides with the observation in Figure 2.

Table 2. The classification performance on unbiased test set (in %; higher is better) evaluated on unbiased test sets of Colored MNIST and Corrupted CIFAR-10 with training on varying BA samples ratios. We denote bias pre-provided type by ○ (without any information), ◑ (bias prior knowledge), and ● (explicit bias supervision). The best result is in bold, while the second-best is with underlines.

Dataset		**Colored MNIST**				**Corrupted CIFAR-10**			
Ratio (%)		99.9	99.5	99.0	95.0	99.9	99.5	99.0	95.0
Vanilla	○	28.58	59.29	74.42	87.13	26.91	30.16	37.71	41.60
+ **p** [44]	○	31.01	64.82	76.84	87.86	26.55	29.48	38.07	42.30
RUBi [7]	◑	27.82	70.80	86.58	96.77	33.70	34.70	34.59	47.23
ReBias [4]	◑	27.71	72.89	85.95	96.87	33.65	34.40	35.82	47.45
End [46]	●	28.19	<u>81.81</u>	88.10	96.99	31.30	33.83	34.02	38.77
DI [50]	●	33.18	80.63	86.28	**98.36**	32.09	33.37	37.65	<u>51.27</u>
LfF [36]	○	30.24	68.90	76.69	96.81	29.89	33.68	35.28	45.38
LFA [27]	○	22.31	64.13	81.83	95.45	32.49	35.74	39.63	47.25
χ-pattern + **p**	●	<u>60.33</u>	64.15	**93.53**	<u>98.30</u>	<u>35.33</u>	**39.31**	**41.32**	**53.37**
χ^2-model (Ours)	○	**66.91**	**88.73**	<u>92.15</u>	97.87	**35.67**	<u>37.61</u>	<u>40.74</u>	49.04

Motivated by mimicking the oracle, we adopt the conditional interpolated prototypes and construct a customized χ-structured metric learning task. Assuming $\boldsymbol{\gamma}$ is large, we use $\mathbf{p}_{\boldsymbol{\gamma}}$ and $\mathbf{p}_{1-\boldsymbol{\gamma}}$ to denote prototypes in bias bags $\mathcal{B}$ with high and low BC ratios. The model should prioritize pulling the majority of low BC ratio bias bag $\mathcal{B}_{1-\boldsymbol{\gamma}}$ closer to $\mathbf{p}_{\boldsymbol{\gamma}}$, which interpolated into the high BC space. Similarly, the high BC bias bag $\mathcal{B}_{\boldsymbol{\gamma}}$ should be pulled to low BC interpolated $\mathbf{p}_{1-\boldsymbol{\gamma}}$. We optimize the cross-entropy loss $\mathcal{L}_{\text{CE}}$ to enable the pulling operation. Concretely, the posterior via the distance $\mathrm{d}(\cdot,\cdot)$ in the representation space is formalized as:

$$\Pr\left(y_i \mid \mathbf{x}_i\right) = \frac{\exp\left(-\mathrm{d}\left(f_{\boldsymbol{\phi}}\left(\mathbf{x}_i\right), \mathbf{p}_{\boldsymbol{\gamma},y_i}\right)/\tau\right)}{\sum_{c\in[C]}\exp\left(-\mathrm{d}\left(f_{\boldsymbol{\phi}}\left(\mathbf{x}_i\right), \mathbf{p}_{\boldsymbol{\gamma},c}\right)/\tau\right)}, \quad (7)$$

where τ is a scaled temperature. One of the branches of the χ-structure classification task is optimizing the $\mathcal{L}_{\text{CE}}$ between samples in the $\mathcal{B}_{1-\boldsymbol{\gamma}}$ and $\mathbf{p}_{\boldsymbol{\gamma}}$. Similarly, the other branch is optimizing between $\mathcal{B}_{\boldsymbol{\gamma}}$ and $\mathbf{p}_{1-\boldsymbol{\gamma}}$ at the same time. As shown in Figure 3, such a high-and-low correspondence captures and compacts the intra-class "hollow". In summary, The bias bags of high BC ratios $\mathcal{B}_{\boldsymbol{\gamma}}$ with corresponding low BC interpolated prototypes conditioning on the target attribute $\mathbf{p}_{1-\boldsymbol{\gamma}}$, and $\mathcal{B}_{1-\boldsymbol{\gamma}}$ with $\mathbf{p}_{\boldsymbol{\gamma}}$ form the χ-structure crossover objective.

4. Experiments

We conduct experiments to verify whether χ^2-model has effective debiasing capability. We begin by introducing bias details in each dataset (as in subsection 4.1). We present the comparison approaches and training details. In subsection 4.2, the experiments show that χ^2-model achieves superior performance in each stage. Furthermore, we exemplify the inherent quality of the prototype-based classification for debiasing tasks and offer ablation studies in subsection 4.3.

Table 3. The classification performance on the unbiased CelebA and NICO test set. The data source *BA* denotes the measurement on BA samples and *BC* is corresponding the BC samples.

Data	**Biased CelebA**			**NICO**
Source	BA	BC	All	All
LfF [36]	73.69	**70.41**	72.05	<u>34.44</u>
DFA [27]	<u>94.01</u>	58.98	<u>76.50</u>	33.10
χ^2-model	**97.66**	<u>60.79</u>	**79.23**	**36.99**

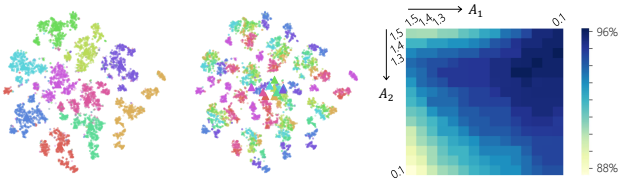
4.1. Experimental Setups

Datasets. To cover more general and challenging cases of bias impact, we validate χ^2-model in a variety of datasets, including two synthetic bias datasets (Colored MNIST [4], Corrupted CIFAR-10 [36]) and two real-world datasets (Biased CelebA [32] and Biased NICO).

The BA samples ratio ρ in the training set is usually high (over 95%), so the bias attribute is highly correlated with the target label. For instance, in the Colored MNIST dataset, each digit is linked to a pre-defined bias *color*. Similarly, there is an *object* target with *corruption* bias in Corrupted CIFAR-10 and a *gender* target with *hair color* bias in Biased CelebA. Following the previous works [19], we use the BA ratio $\rho \in \{95.0\%, 99.0\%, 99.5\%, 99.9\%\}$ for Colored MNIST and Corrupted CIFAR-10, respectively, and approximately 96% for Biased CelebA. The Biased NICO dataset is dedicatedly sampled in NICO [17], initially designed for OOD (Out-of-Distribution) image classification. NICO is enriched with variations in the *object* and *context* dimensions. We select the bias attribute with the highest co-occurrence frequency to the target one, *e.g.*, *helicopter* to *sunset* in training set correlates strongly (see BA samples in Figure 3). The correlation ratio is roughly controlled to 86%. For more details please see the supplementary material.

Table 4. The performance of BC samples mining on Colored MNIST with 99.5% BA ratio. *Acc.* denotes mean accuracy of ranking with top-300. 98%-σ denotes the number of samples required to contain 98% of BC samples. *AP* is average precision. ↑ means higher is better, while ↓ is the opposite.

Measure	Acc. ↑	98%-σ ↓	AP ↑
Entropy [20]	78.33	632	83.52
Confidence [28]	80.33	590	85.61
Loss [36]	<u>94.39</u>	<u>418</u>	<u>98.22</u>
Pleiss *et al*. [39]	82.67	686	89.24
Zhao *et al*. [55]	90.33	451	96.04
χ-pattern (Ours)	**95.84**	**372**	**98.44**



(a) t-SNE on a_t (b) t-SNE on a_b (c) ablation study of A_1, A_2

Figure 6. **(a)** and **(b)** show the t-SNE visualization of our unbiased representation in terms of *digit* (the target attribute) and *color* (the bias one) in Colored MNIST, respectively. **(c)** displays the top-300 mean accuracy of mining BC samples on Colored MNIST with 99.5% BA ratio when A_1 and A_2 are changed.

Baselines. We carefully select the classic and the latest trending approaches as baselines: (1) Vanilla model training with cross entropy as described in subsection 2.1. (2) Bias-tailored approaches with pre-provided bias type: RUBi, Rebias. (3) Explicit approaches under the guidance of total bias supervision: EnD and DI. (4) Implicit methods through general bias properties: LfF and DFA.

Implementation details. Following the existing popular benchmarks [19, 23], we use the four-layer CNN with kernel size 7×7 for the Colored MNIST dataset and ResNet-18 [15] for Corrupted CIFAR-10, Biased CelebA, and Biased NICO datasets. For a fair comparison, we re-implemented the baselines with the same configuration. We mainly focus on unbiased test accuracy for all categories. All models are trained on an NVIDIA RTX 3090 GPU.

Baselines for the first stage. To better demonstrate the effectiveness of χ-pattern, we consider related sample-specific scoring methods [39, 55] and report average precision, top-threshold accuracy, and the minimum samples (threshold) required for 98% accuracy. For more results, such as PR curves, please see the supplementary material.

4.2. Quantitative Evaluation

Performance of χ-shape pattern. As shown in Table 4, our χ-pattern matching achieves state-of-the-art performance on various evaluation metrics. Thus, the χ-structure metric learning objective can leverage more IASs cues to interpolate bias attribute and further learn the debiased representation.

χ^2-model in different types of bias constructions. (1) Synthetic bias on Colored MNIST and Corrupted CIFAR-10: From Table 2 we find that under extreme bias influence, as ρ is 99.9%, the performance of the vanilla model and other baselines decreases catastrophically. In contrast, Our χ^2-model maintains the robust and efficient debiasing capability on the unbiased dataset. Further, more results in Figure 2 present the remarkable performance of our χ^2-model compared to other methods. **(2)** Real-world bias on Biased CelebA and Biased NICO: Table 3 shows that compared to the recent methods which do not pre-provide any bias information as the same as ours, our method also achieves outstanding performance. The above experiments indicate that conditional interpolation among IASs feedback the shift of the intrinsic knowledge and facilitate learning debiased representations even in extremely biased conditions.

4.3. Further analysis

The inherent debiasing capability of prototype-based classification. We directly construct the prototype by averaging the trained representations of the vanilla model (as in Table 2 line two named "+ **p**"). The results show that on some datasets like Colored MNIST, the prototype-based classifier without training achieves performance improvement.

Visualize the test set representation on 2D embedding space via t-SNE. Figure 6 shows the 2D projection of the feature extracted by χ^2-model on Colored MNIST. We color the target and bias attributes separately. The representations follow the target attribute to cluster into classes which indicates that our model learns the debiased representations.

Ablation studies. We further perform the ablation analysis of the matching factors A_1, A_2 in Eq. 3, which directly determine the χ-shape curves. The results show the first stage of χ^2-model is robust to changes in hyperparameters. For more related experiments like on different BC identification thresholds, please see the supplementary material.

5. Conclusion

Although intra-class biased samples with a "hollow" structure impede learning debiased representations, we propose the χ^2-model to leverage *Intermediate* Attribute Samples (IASs) to remodel the representation compactness. χ^2-model works in a two-stage manner, matching and ranking possible IASs based on their χ-shape training dynamics followed by a χ-branch metric-based debiasing objective with conditional attribute interpolation.

Acknowledgments. This research was supported by NSFC (61773198,61921006,62006112), Collaborative Innovation Center of Novel Software Technology and Industrialization, NSF of Jiangsu Province (BK20200313).

References

[1] Vedika Agarwal, Rakshith Shetty, and Mario Fritz. Towards causal VQA: revealing and reducing spurious correlations by invariant and covariant semantic editing. In *CVPR*, pages 9687–9695, 2020. 1

[2] Martín Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *CoRR*, abs/1907.02893, 2019. 3

[3] Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron C. Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In *ICML*, pages 233–242, 2017. 2

[4] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *ICML*, pages 528–539, 2020. 1, 7

[5] Yoshua Bengio, Yann LeCun, and Geoffrey E. Hinton. Deep learning for AI. *Communications of the ACM*, 64(7):58–65, 2021. 1

[6] Wieland Brendel and Matthias Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. In *ICLR*, 2019. 1

[7] Rémi Cadène, Corentin Dancette, Hedi Ben-younes, Matthieu Cord, and Devi Parikh. Rubi: Reducing unimodal biases for visual question answering. In *NeurIPS*, pages 839–850, 2019. 3, 7

[8] Rocío Cañamares and Pablo Castells. Should I follow the crowd?: A probabilistic analysis of the effectiveness of popularity in recommender systems. In *SIGIR*, pages 415–424, 2018. 1

[9] Pengyu Cheng, Weituo Hao, Siyang Yuan, Shijing Si, and Lawrence Carin. Fairfil: Contrastive neural debiasing method for pretrained text encoders. In *ICLR*, 2021. 3

[10] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. In *EMNLP-IJCNLP*, pages 4067–4080, 2019. 3

[11] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *ICLR*, 2019. 1, 3

[12] Karan Goel, Albert Gu, Yixuan Li, and Christopher Ré. Model patching: Closing the subgroup performance gap with data augmentation. In *ICLR*, 2021. 1

[13] Yue Guo, Yi Yang, and Ahmed Abbasi. Auto-debias: Debiasing masked language models with automated biased prompts. In *ACL*, pages 1012–1023, 2022. 1

[14] He He, Sheng Zha, and Haohan Wang. Unlearn dataset bias in natural language inference by fitting the residual. In *EMNLP-IJCNLP Workshop*, pages 132–142, 2019. 1

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 8

[16] Rundong He, Zhongyi Han, Yang Yang, and Yilong Yin. Not all parameters should be treated equally: Deep safe semi-supervised learning under class distribution mismatch. In *AAAI*, pages 6874–6883, 2022. 3

[17] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognition*, 110:107383, 2021. 2, 7

[18] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *ECCV*, pages 793–811, 2018. 3

[19] Youngkyu Hong and Eunho Yang. Unbiased classification through bias-contrastive and bias-balanced learning. In *NeurIPS*, pages 26449–26461, 2021. 6, 7, 8

[20] Ajay J. Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *CVPR*, pages 2372–2379, 2009. 8

[21] Aditya Khosla, Tinghui Zhou, Tomasz Malisiewicz, Alexei A. Efros, and Antonio Torralba. Undoing the damage of dataset bias. In *ECCV*, pages 158–171, 2012. 1

[22] Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *CVPR*, pages 9012–9020, 2019. 1, 2

[23] Eungyeup Kim, Jihyeon Lee, and Jaegul Choo. Biaswap: Removing dataset bias with bias-tailored swapping augmentation. In *ICCV*, pages 14972–14981, 2021. 8

[24] Nayeong Kim, SEHYUN HWANG, Sungsoo Ahn, Jaesik Park, and Suha Kwak. Learning debiased classifier with biased committee. In *NeurIPS*, pages 18403–18415, 2022. 3

[25] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. In *ICLR*, 2023. 3

[26] Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. 1

[27] Jungsoo Lee, Eungyeup Kim, Juyoung Lee, Jihyeon Lee, and Jaegul Choo. Learning debiased representation via disentangled feature augmentation. In *NeurIPS*, 2021. 1, 3, 7

[28] Mingkun Li and Ishwar K. Sethi. Confidence-based active learning. *IEEE TPAMI*, 28(8):1251–1261, 2006. 8

[29] Yi Li and Nuno Vasconcelos. REPAIR: removing representation bias by dataset resampling. In *CVPR*, pages 9572–9581, 2019. 2, 3

[30] Yingwei Li, Qihang Yu, Mingxing Tan, Jieru Mei, Peng Tang, Wei Shen, Alan L. Yuille, and Cihang Xie. Shape-texture debiased neural network training. In *ICLR*, 2021. 1

[31] Evan Zheran 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 *ICML*, pages 6781–6792, 2021. 3

[32] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, pages 3730–3738, 2015. 7

[33] Michael Mendelson and Yonatan Belinkov. Debiasing methods in natural language understanding make bias more accessible. In *EMNLP*, pages 1545–1557, 2021. 1

[34] Matthias Minderer, Olivier Bachem, Neil Houlsby, and Michael Tschannen. Automatic shortcut removal for self-

supervised representation learning. In *ICML*, pages 6927–6937, 2020. 1, 3

[35] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. Controlling fairness and bias in dynamic learning-to-rank. In *SIGIR*, pages 429–438, 2020. 1

[36] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. In *NeurIPS*, pages 20673–20684, 2020. 1, 2, 3, 7, 8

[37] Giacomo De Palma, Bobak Toussi Kiani, and Seth Lloyd. Random deep neural networks are biased towards simple functions. In *NeurIPS*, pages 1962–1974, 2019. 2

[38] Guillermo Valle Pérez, Chico Q. Camargo, and Ard A. Louis. Deep learning generalizes because the parameter-function map is biased towards simple functions. In *ICLR*, 2019. 2

[39] Geoff Pleiss, Tianyi Zhang, Ethan R. Elenberg, and Kilian Q. Weinberger. Identifying mislabeled data using the area under the margin ranking. In *NeurIPS*, pages 17044–17056, 2020. 8

[40] Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *ICLR*, 2020. 3

[41] Ramprasaath Ramasamy Selvaraju, Stefan Lee, Yilin Shen, Hongxia Jin, Shalini Ghosh, Larry P. Heck, Dhruv Batra, and Devi Parikh. Taking a HINT: leveraging explanations to make vision and language models more grounded. In *ICCV*, pages 2591–2600, 2019. 1

[42] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. In *NeurIPS*, pages 9573–9585, 2020. 2

[43] Sahil Singla and Soheil Feizi. Salient imagenet: How to discover spurious features in deep learning? In *ICLR*, 2022. 1

[44] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *NIPS*, pages 4077–4087, 2017. 7

[45] Lin Sui, Chen-Lin Zhang, and Jianxin Wu. Salvage of supervision in weakly supervised object detection. In *CVPR*, pages 14207–14216, 2022. 3

[46] Enzo Tartaglione, Carlo Alberto Barbano, and Marco Grangetto. End: Entangling and disentangling deep representations for bias correction. In *CVPR*, pages 13508–13517, 2021. 1, 7

[47] Tatiana Tommasi, Novi Patricia, Barbara Caputo, and Tinne Tuytelaars. A deeper look at dataset bias. In *Pattern Recognition*, volume 9358, pages 504–516, 2015. 1

[48] Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. In *CVPR*, pages 1521–1528, 2011. 1

[49] Haohan Wang, Zexue He, Zachary C. Lipton, and Eric P. Xing. Learning robust representations by projecting superficial statistics out. In *ICLR*, 2019. 3

[50] Zeyu Wang, Klint Qinami, Ioannis Christos Karakozis, Kyle Genova, Prem Nair, Kenji Hata, and Olga Russakovsky. Towards fairness in visual recognition: Effective strategies for bias mitigation. In *CVPR*, pages 8916–8925, 2020. 1, 7

[51] Kai Yuanqing Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. In *ICLR*, 2021. 1, 2

[52] Han-Jia Ye, Lu Han, and De-Chuan Zhan. Revisiting unsupervised meta-learning via the characteristics of few-shot tasks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(3):3721–3737, 2023. 3

[53] Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. Causal intervention for leveraging popularity bias in recommendation. In *SIGIR*, pages 11–20, 2021. 1

[54] Yi-Kai Zhang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Audio-visual generalized few-shot learning with prototype-based co-adaptation. In *Interspeech*, pages 531–535, 2022. 3

[55] Bowen Zhao, Chen Chen, Qi Ju, and Shutao Xia. Learning debiased models with dynamic gradient alignment and bias-conflicting sample mining. *CoRR*, abs/2111.13108, 2021. 8

[56] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *EMNLP*, pages 2979–2989, 2017. 1

[57] Da-Wei Zhou, Yang Yang, and De-Chuan Zhan. Learning to classify with incremental new class. *IEEE Trans. Neural Networks Learn. Syst.*, 33(6):2429–2443, 2022. 3

[58] Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Co-transport for class-incremental learning. In *ACM Multimedia*, pages 1645–1654, 2021. 3

[59] Nengjun Zhu, Jian Cao, Yanchi Liu, Yang Yang, Haochao Ying, and Hui Xiong. Sequential modeling of hierarchical user intention and preference for next-item recommendation. In *WSDM*, pages 807–815. ACM, 2020. 1

[60] Wei Zhu, Haitian Zheng, Haofu Liao, Weijian Li, and Jiebo Luo. Learning bias-invariant representation by cross-sample mutual information minimization. In *ICCV*, pages 14982–14992, 2021. 3