The Two Dimensions of Worst-case Training and Their Integrated Effect for Out-of-domain Generalization

Zeyi Huang[1,2⋆] Haohan Wang[1⋆] Dong Huang[1] Yong Jae Lee[2†] Eric P. Xing[1†]
[1]Carnegie Mellon University [2]University of Wisconsin-Madison
{zeyih@andrew, haohanw@cs, donghuang@, epxing@cs}.cmu.edu yongjaelee@cs.wisc.edu

Abstract

Training with an emphasis on "hard-to-learn" components of the data has been proven as an effective method to improve the generalization of machine learning models, especially in the settings where robustness (e.g., generalization across distributions) is valued. Existing literature discussing this "hard-to-learn" concept are mainly expanded either along the dimension of the samples or the dimension of the features. In this paper, we aim to introduce a simple view merging these two dimensions, leading to a new, simple yet effective, heuristic to train machine learning models by emphasizing the worst-cases on both the sample and the feature dimensions. We name our method W2D following the concept of "Worst-case along Two Dimensions". We validate the idea and demonstrate its empirical strength over standard benchmarks.

1. Introduction

The remarkable empirical performance of deep learning over *i.i.d* data, sometimes paralleling the human visual system [23, 37], has encouraged the community to challenge potentially more demanding scenarios where the models are trained with data from one or more distributions but tested with data from other distributions. We refer to this scenario as the out-of-distribution (OOD) generalization testing setting following the terminology used in [74].

In this OOD test scenario, deep learning techniques often underdeliver the promising results made with *i.i.d* data, as observed by multiple preceding works with different strategies to generate the test data, such as with salient patterns added to the data [17, 27], with carefully constructed imperceptible noise perturbing the data (adversarial attacks) [20, 60], or with additionally collected datasets that humans can nonetheless generalize to despite potentially significant disparities between the training and test distributions (*e.g.*, domain adaptation/generalization) [6, 48].

⋆,† equal contribution

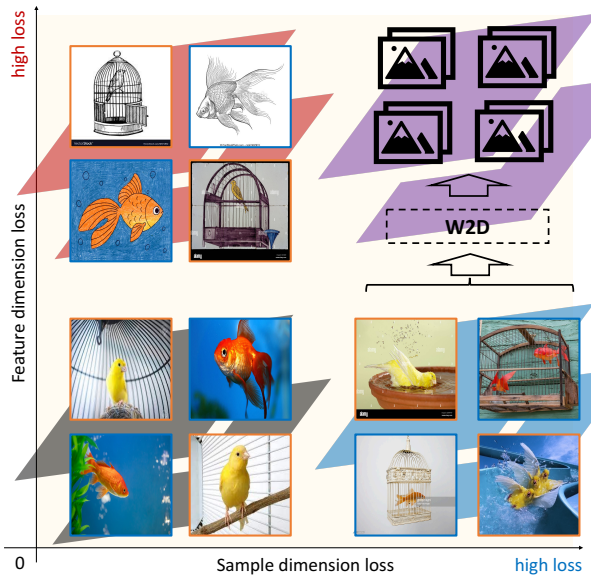

Figure 1. The conceptual illustration of our main idea W2D for a simple example of canary vs. goldfish image classification. For a regular model trained on standard images (left-bottom block), there are two dimensions of hard samples: following the categorization in [74], the vertical dimension corresponds to the "diversity shift" of the images (*e.g.*, photos of the animals vs. cartoons of the animals) and the horizontal dimension corresponds to the "correlation shift" of the images (*e.g.*, birds in cage and fish in water vs. fish in cage and birds in water). The W2D algorithm conceptually selects the images that are hard at the sample dimension and augment these samples toward being harder at the feature dimension.

Despite the variation in multiple OOD settings, the underlying reason leading to the performance drop may have a shared theme: the models' incapability to learn what humans will consider important in the data, as discussed previously in empirical [70] and statistical [68] perspectives.

Thus, we conjecture that a key to training models that can perform consistently well in the OOD setting is to design a new training heuristic that can better imitate the hu-

man's learning behaviors. In addition, we also hope the new heuristic is simple and general so that it can be directly plugged into and benefit existing methods across different architectures, optimizers, losses, or regularizations.

A psychological prior: In seeking the answer of how a human can learn most efficiently, we notice that a world-renowned psychologist (Dr. K. Anders Ericsson) has devoted his life-long career decoding the habits of people with expert-level performances. His main conclusion [14] is that the high-end performances are the result of extensive practices *beyond one's comfort zone*.

Back to the discussion of machine learning, we analogize the *beyond comfort zone* elements of one's daily life to the elements of the training data that are particularly *hard to learn* for a model. We notice this "element" can be interpreted with two perspectives: one interpretation is that a certain pattern across many images is hard to learn; and the other is that some specific images in the dataset are hard to learn.

Previous discussions devoted exclusively to either one of these two elements has been expanded extensively. For example, a line of methods have been invented to counter the model's tendency to learn some simple patterns [4, 50, 65, 66], and another line of methods have been introduced to push the model to learn patterns represented by a small set of samples [38, 57]. However, there seems to be no discussion aiming to train the model to overcome the limitations raised from both of these perspectives, while doing so would intuitively improve the model's performances, as well as align well with the psychological findings mentioned above.

In this paper, inspired by the psychological prior above, we aim to introduce a simple training heuristic that will push the model to learn the hard-to-learn concept on both the feature dimension and the sample dimension. As our method is intuitively a combination of worst-case training at the feature level and worst-case training at the sample level, the new technique can serve as a simple heuristic to replace the existing training procedure of deep learning models regardless of model architecture, optimizer, loss, or regularization *etc*, as long as the optimization is within the gradient descent family. We name our method W2D following the concept of "Worst-case along Two Dimensions".

The remainder of this paper is organized as follows. In Section 2, we first introduce the background of this paper, with an emphasis on the "worst-case training" along the two dimensions, and their corresponding effects on OOD generalization, which inspired us to investigate the integrated effect of these two worst-case training dimensions. In Section 3, we introduce our new heuristic that combines these two directions, and we demonstrate the method's empirical strength in Sections 4 and 5. We offer several related discussion in Section 6 before we conclude with Section 7.

2. Background

We introduce the background of our work in this section. We first offer a brief summary of works that improve a model's OOD generalization performance. We then focus on related work that are devoted to solving the two challenges at the feature level and at the sample level, respectively. As we notice that some of the methods solving these two problems have a common theme of emphasizing the hard-to-learn elements from the data, we continue the discussion with a focus on the worst-case training methods along both dimensions. Finally, we wrap up this section with a summary of the key contributions made in this paper.

2.1. Domain Adaptation, Domain Generalization, and New Paradigms

The investigation of a model's generalization ability across distributions can probably be traced back to the study of domain adaptation [6, 7], which studies the general problem of maintaining a model's performance over a test distribution that is different from the training distribution. Early-stage theoretical work suggests one of the key factors of learning cross-domain generalizable models is to enforce invariance across distributions [6], and this has inspired a long line of work aiming to learn invariant representations across the training and testing distributions [13, 15, 18, 34, 44, 56, 71, 77], with the most popular recent examples being domain adversarial neural networks [16].

Domain generalization [48] is another mainstream research topic in the study of OOD generalization. It extends the setup of domain adaptation to a setting in which the testing distribution data, even unlabelled, is not available during training. Instead, models are trained with data from multiple training distributions, and enforcing invariance across these training distributions has become a major theme [1, 10, 19, 22, 42, 47, 52, 54, 69].

However, recently, the paradigm of learning invariant representations has been challenged by the argument that invariance is not sufficient for cross-domain generalization if the data have different labelling functions [72, 78], potentially leading to a new paradigm of learning across distributions with disparate labelling functions [68], as will be detailed with examples in the next section.

As a result, recent studies are not always bounded by the concepts of domain adaptation or domain generalization, but are conducted in the new paradigm with disparate labelling functions.

2.2. Hard-to-learn Patterns and Solutions

One of the mainstream studies in robust machine learning focuses on the issue of models learning some patterns in the training data that are not present in the test data, with the most popular example probably being the snow background in husky vs. wolf classification [55]. This problem is of-

ten referred to as biases [61], spurious features [64], confounding factors [45], or superficial features [66], but the solution to counter the problem usually has a unified theme of leveraging the human knowledge of the differences between training and testing distributions to either regularize the hypothesis space [4, 50, 65, 66] or to augment the data [17, 28, 29, 67], as summarized in [68].

Interestingly, the RSC method [31] also aims to solve the challenge along this line, but it does not require prior knowledge over the patterns. Building upon an assumption that learning all the features, instead of just the most discriminative ones, will benefit the OOD generalization, RSC essentially uses a selective dropout mechanism to perform augmentation, and achieves good benchmark performances on popular OOD datasets [31, 74]. Conceptually, RSC prepares the features for each sample by dropping out the most predictive features (*i.e.*, creating features that are challenging for the model to learn).

2.3. Hard-to-learn Samples and Solutions

On the other hand, hard-to-learn samples pose a different challenge: some samples in the training data are ignored by the model because these samples are considered to be the "minority" in the training set [12, 46]. To counter this problem, training procedures that emphasize the minority samples have been introduced, such as the family of DRO methods [30, 39, 50, 51, 57], with different strategies to identify the minority samples and to interpolate the training set according to weighting factors that favor the minority samples. Intuitively, these methods prepare the batches of samples with an emphasis on the samples that are challenging for the model to learn.

Further along this line, the community extends the idea of interpolation to extrapolation by adjusting the weighting factors so that the weights for simpler samples can even be negative, to further push the models to focus on the hard-to-learn samples. The VREx method [38] is introduced in this context and achieves leading empirical performances on benchmarks [38, 74].

2.4. Worst-case Training in Each Dimension and the Corresponding Effects

With the proliferation of methods introduced to improve models' OOD performances, and various empirical claims, there have been some integrated comparisons of the various methods. For example, interestingly, DomainBed suggests that all these new methods are still shy of the conventional Empirical Risk Minimization (ERM) method [21] under their extensive range of hyperparameter choices. While this message sets a striking alarm to the community, it may seem overly pessimistic (more details on this are offered later in Section 6).

Recently, the OOD-bench [74] extends the spirit of DomainBed but offers a more finer-grained analysis of the models' performances. It comprehensively examines the performances of the recent models over popular benchmark datasets, but with separate discussions on "diversity shift" datasets and "correlation shift" datasets. Diversity shift datasets refer to the benchmarks with a relatively significant style shift from training distribution to the test distribution (such as from photos to sketches), and correlation shift datasets refer to the benchmarks with clear defined spurious features that are correlated with the label (such as when the color of the digits are associated with the labels of the digits in the image digit classification task).

OOD-bench investigates the popular methods along these two directions, and shows that for each direction, there are only a couple of methods that outperform ERM. Interestingly, the best performing method for diversity shift is RSC [31], one of the most advanced methods aimed to learn hard-to-learn patterns, with a heuristic to push the model to train with generated worst-case features. On the other hand, the best performing method for correlation shift is VREx [38], one of the most advanced methods aimed to learn hard-to-learn samples, with a heuristic to push the model to train with selected worst-case samples.

Our contribution is inspired by the above discussion: if worst-case training in the feature dimension excels at diversity shift, while worst-case training in the sample dimension excels at correlation shift, if we can integrate these two worst-case training methods into one simple heuristic, the new method will likely lead to sufficiently good empirical performance for both diversity and correlation shift.

2.5. Key Contributions

In comparison to the previous methods discussed along these two lines, we believe the key contributions we make in this paper are as follows.

- We discuss the previous methods with a unified theme of worst-case training along two different dimensions, and doing so naturally leads to the integration of these methods.
- We introduce a new method, called W2D, as an integration of these two types of training methods. W2D is a simple heuristic that can be directly plugged into any training process regardless of model architecture, loss function, regularization or optimizer.
- We demonstrate strong empirical performance on multiple benchmark datasets, and conduct ablation studies to understand the contribution of each of component of the method.

3. Method

In this section, we first formalize the two worst-case training methods, which naturally leads to the introduction of our method. Then, with the main framework of our

method introduced, we continue to discuss a whole-batch patching heuristic we use in the experiments that have benefited our method empirically with non-negligible margins.

3.1. W2D Method

We first introduce our notations. We use $(\mathbf{X}, \mathbf{Y})$ to denote a dataset with n (data,label) paired samples. Thus $\mathbf{X} \in \mathcal{R}^{n \times p}$ and $\mathbf{Y} \in \mathcal{R}^n$. We use $f(\cdot;\theta)$ to denote the model we aim to train, and use $e(\cdot;\theta_e)$ and $d(\cdot;\theta_d)$ to denote encoder and decoder, respectively. Thus, we have $f(\cdot;\theta) = d(e(\cdot;\theta_e);\theta_d)$. We use $\mathbf{w}$ to denote a weight vector of length n. We use $\mathbf{m}$ to denote a masking vector with some elements to be 0 and others to be 1; the length of $\mathbf{m}$ is the same as the feature dimension (the output of $e(\cdot;\theta_e)$). We use $l(\cdot,\cdot)$ to denote a generic loss function.

The vanilla training process of a model is

$$\widehat{\theta}_{\text{vanilla}} = \arg\min_{\theta} \frac{1}{n} \sum_i l(f(\mathbf{X}_i;\theta), \mathbf{Y}_i)$$

Worst-case along feature dimension We formalize the first worst-case method, with a generic form as follows:

$$\widehat{\theta}_{\text{w\_feature}} = \arg\min_{\theta} \frac{1}{n} \sum_i \max_{\mathbf{m}} l(d(\mathbf{m} \odot e(\mathbf{X}_i;\theta_e);\theta_d), \mathbf{Y}_i), \tag{1}$$

where $\odot$ denotes the element-wise product.

In particular, the RSC method [31] introduces the $\mathbf{m}$ with a hyperparameter ρ, which denotes the ρ fraction of the elements are zeros. The maximization step is achieved by examining the magnitude of the gradient of $\frac{\partial d(\mathbf{e};\theta_d)}{\partial \mathbf{e}}$.

Worst-case along sample dimension We formalize the second worst-case method, with a generic form as follows:

$$\widehat{\theta}_{\text{w\_sample}} = \arg\min_{\theta} \frac{1}{n} \sum_i \max_{\mathbf{w}_i} \mathbf{w}_i l(f(\mathbf{X}_i;\theta), \mathbf{Y}_i), \quad \text{subject to} \sum_i \mathbf{w}_i = 1$$

In general, the choice of $\mathbf{w}_i$ depends on $l(f(\mathbf{X}_i;\theta), \mathbf{Y}_i)$, with concrete differences across different methods such as [39, 50, 57]. A common theme is that, the higher the loss is, the bigger $\mathbf{w}_i$ is.

In practice, because we use batch-wise optimization, the estimation of $\mathbf{w}$ is not straightforward. Fortunately, we can use a simple alternative: for each batch, we select the samples with high losses (through a forward pass), and then use these samples to update the model. This is a heuristic used by multiple methods such as [9, 11, 33, 73].

Algorithm 1: W2D Algorithm

Input: data set $(\mathbf{X}, \mathbf{Y})$, percentage of samples used per batch ρ, percentage of whole batch patching κ, batch size η, maximum number of epochs T, and other RSC hyperparameters;
Output: Classifier $f(\cdot;\theta)$;
randomly initialize the model θ_0;
calculate the number of iterations $K = n/\eta$;
while $t \leq (1-\kappa)T$ **do**
  for *a batch of data* $(\mathbf{X}, \mathbf{Y})_k$ *where* $k \leq K$ **do**
    forward pass to calculate the loss $l(f(\mathbf{X}_i;\theta_{t,k-1}), \mathbf{Y}_i)$ of every sample in the batch;
    select the top $\eta\rho$ samples with highest loss to construct $(\mathbf{X}, \mathbf{Y})_{k,\rho}$;
    Train the model with $(\mathbf{X}, \mathbf{Y})_{k,\rho}$ following (1).
  end
end
while $(1-\kappa)T < t \leq T$ **do**
  for *a batch of data* $(\mathbf{X}, \mathbf{Y})_k$ *where* $k \leq K$ **do**
    Train the model with $(\mathbf{X}, \mathbf{Y})_k$ following (1).
  end
end

W2D Method Integrating the two methods above, we have

$$\widehat{\theta}_{\text{W2D}} = \arg\min_{\theta} \frac{1}{n} \sum_i \max_{\mathbf{m},\mathbf{w}_i} \mathbf{w}_i l(d(\mathbf{m} \odot e(\mathbf{X}_i;\theta_e);\theta_d), \mathbf{Y}_i), \quad \text{subject to} \sum_i \mathbf{w}_i = 1$$

In practice, we use the RSC method to identify the $\mathbf{m}$ for worst-case training in the feature dimension, and use the above heuristic to perform worst-case training in the sample dimension.

3.2. Whole-batch Patching Heuristic

As introduced above, W2D selects the worst-case training samples with highest loss in each training iteration, as the model evolves over training, it is possible that what was once considered easy becomes hard, and vice versa. Therefore, we can intuitively expect the model to see all the samples in the training set given sufficient training iterations. However, chances are that some of the samples are never seen by the model during training as these samples are always considered not hard enough; it is obvious not ideal that there is a chance the model does not take full advantage of the training set.

To counter this potential issue, we simply switch to whole batch training during the last $\kappa\%$ of training epochs,

which leads to better empirical performances across different model selection strategies. We verify the effectiveness of this simple approach in our ablation study. More results can be found in Section 5.4.

The full description of W2D with the whole batch patching heuristic is detailed in Algorithm 1.

4. Experiments

4.1. Experimental Setup

We follow the setting in [74] and evaluate domain generalization on both types of distribution shift: diversity shift and correlation shift. Specifically, we use the same strategy for model selection, dataset splitting, and network backbone. More details of the experimental settings can be found in the discussion and supplementary materials.

4.2. Datasets, Hyperparameter Search and Model Selection

We choose datasets that cover as much variety as possible from the various OoD research areas for our experiments. We conduct experiments on seven OOD datasets: CMNIST [2], CelebA [43], NICO [25], Terra Incognita [5], OfficeHome [63], WILDS-Camelyon, [35] and PACS [40]. These datasets are divided into two categories based on their estimated diversity and correlation shift.

We use the same hyperparameter search protocol as [21, 74]: a 20-times random hyperparameter search is conducted for every dataset and algorithm pair, and then the search process is repeated for another two random series of hyperparameter combinations, weight initializations, and dataset splits. The three series yield the three best accuracies in total over which a mean and standard error bar is computed for every dataset-algorithm pair.

To be consistent with existing line of work, models trained on PACS, OfficeHome, and Terra Incognita are selected by training-domain validation; models trained on WILDS-Camelyon and NICO are selected by leave-one-domain-out validation; while models trained on Colored MNIST and CelebA are selected by test-domain validation. Details of these selection strategies can be found in [74].

4.3. Empirical Results

The benchmark results are shown in Table 1 and Table 2. In addition to mean accuracy and standard error bar, we follow the Ood-bench [74] to report a ranking score for each algorithm with respect to Empirical Risk Minimization (ERM) [62]. Specifically, depending on whether the attained accuracy is lower than, within, or higher than the standard error bar of ERM accuracy on the same dataset, scores -1, 0, +1 are assigned to every dataset-algorithm pair. Adding up the scores across all datasets listed in the table produces the ranking score for each algorithm. The ranking score reflects a relative degree of robustness against diversity and correlation shift compared to ERM.

Note, for CMNIST, Ood-bench [74] evaluates the results using the -90 as testing domain while DomainBed [21] reports the results averaged over the +90, +80, and -90 domains. We follow Ood-bench's setting to report the results in Table 2 and we also report the results of the DomainBed's setting in the supplementary material. The choice of the settings does not affects our ranking score in Table 2. In Section 5, we discuss a special property of CMNST and propose a modified version of W2D which can achieve a significant gain on CMNIST.

We observe that W2D is the only algorithm that can achieve consistently better performance than ERM on both types of distribution shifts. Specifically, W2D is among the top three in both the datasets dominated by diversity shift as well as the datasets dominated by correlation shift. This comprehensive evaluation supports the view that W2D could serve as a simple heuristic to replace existing training approaches for real-world applications because real-world data have both kinds of distribution shifts.

5. Ablation Study

There are altogether four hyperparameters for W2D, two of them are directly inherited from RSC [31]: *feature dropping percentage*, ϕ, controls the different dropping percentages to mute feature maps; *batch dropping percentage*, β, controls the different batch size percentages to apply feature dropping. There are two new hyperparameters introduced along the sample dimension by W2D: *worse-case sample percentage*, ρ, controls the fraction of the batch size samples with the highest loss used for training; *whole batch patching percentage*, κ, controls the percentage of training time trained using the whole batch.

We use the default hyperparameters from RSC for feature dropping percentage and batch dropping percentage. For selecting the worst-case along the sample dimension, we conduct two ablation studies on possible configurations for it on the standard benchmarks. All results are produced by following Ood-bench's setting.

Overall, we set the hyperparameter search space of W2D as $\phi \in [0.1, 0.4]$, $\beta \in [0.1, 0.3]$, $\rho \in [0.1, 0.5]$, $\kappa \in [0.2, 0.4]$.

5.1. Effect of Worst-case sample percentage ρ

We test W2D with different percentages of worst-case batch samples in Table 3. For PACS, the fewer the worst-case samples used for training, the higher the test-validation accuracy. This result suggests that focusing on more hard-to-learn worse-case samples can better push the limit of the model's potential generalization power as indicated by the higher test-validation accuracy.

Algorithm	PACS	OfficeHome	TerraInc	Camelyon	Average	Ranking score
W2D	83.4 ± 0.3	63.5 ± 0.1	44.5 ± 0.5	95.2 ± 0.3	71.7	+3
RSC [31]	82.8 ± 0.4	62.9 ± 0.4	43.6 ± 0.5	94.9 ± 0.2	71.1	+2
MMD [42]	81.7 ± 0.2	63.8 ± 0.1	38.3 ± 0.4	94.9 ± 0.4	69.7	+2
SagNet [49]	81.6 ± 0.4	62.7 ± 0.4	42.3 ± 0.7	95.0 ± 0.2	70.4	+1
ERM [62]	81.5 ± 0.0	63.3 ± 0.2	42.6 ± 0.9	94.7 ± 0.1	70.5	0
IGA [36]	80.9 ± 0.4	63.6 ± 0.2	41.3 ± 0.8	95.1 ± 0.1	70.2	0
CORAL [59]	81.6 ± 0.6	63.8 ± 0.3	38.3 ± 0.7	94.2 ± 0.3	69.5	0
IRM [2]	80.9 ± 0.4	63.6 ± 0.2	41.3 ± 0.8	95.1 ± 0.1	70.2	0
VREx [38]	81.8 ± 0.4	63.5 ± 0.1	40.7 ± 0.7	94.1 ± 0.3	70.0	-1
GroupDRO [57]	80.4 ± 0.3	63.2 ± 0.2	36.8 ± 1.1	95.2 ± 0.2	68.9	-1
ERDG [79]	80.5 ± 0.5	63.0 ± 0.4	41.3 ± 1.2	95.5 ± 0.2	70.1	-2
DANN [16]	81.1 ± 0.4	62.9 ± 0.6	39.5 ± 0.2	94.9 ± 0.0	69.6	-2
MTL [8]	81.2 ± 0.4	62.9 ± 0.2	38.9 ± 0.6	95.0 ± 0.1	69.5	-2
Mixup [75]	79.8 ± 0.6	63.3 ± 0.5	39.8 ± 0.3	94.6 ± 0.3	69.4	-2
ANDMask [53]	79.5 ± 0.0	62.0 ± 0.3	39.8 ± 1.4	95.3 ± 0.1	69.2	-2
ARM [76]	81.0 ± 0.4	63.2 ± 0.2	39.4 ± 0.7	93.5 ± 0.6	69.3	-3
MLDG [41]	73.0 ± 0.4	52.4 ± 0.2	27.4 ± 2.0	91.2 ± 0.4	61.0	-4

Table 1. Performance of domain generalization algorithms on datasets dominated by diversity shift. W2D achieves better performance than ERM on three datasets with top 1 ranking score.

Algorithm	CMNIST	NICO	CelebA	Average	Prev score	Ranking score
VREx [38]	56.3 ± 1.9	71.0 ± 1.3	87.3 ± 0.2	71.5	-1	+1
GroupDRO [57]	32.5 ± 0.2	71.8 ± 0.8	87.5 ± 1.1	63.9	-1	+1
W2D	31.0 ± 0.3	71.6 ± 0.9	87.7 ± 0.4	63.4	+3	+1
ERM [62]	29.9 ± 0.9	71.4 ± 1.3	87.2 ± 0.6	62.8	0	0
MTL [8]	29.3 ± 0.1	70.2 ± 0.6	87.0 ± 0.7	62.2	-2	0
ERDG [79]	31.6 ± 1.3	70.6 ± 1.3	84.5 ± 0.2	62.2	-2	0
ARM [76]	34.6 ± 1.8	63.9 ± 1.8	86.6 ± 0.7	61.7	-3	0
MMD [42]	50.7 ± 0.1	68.3 ± 1.0	86.0 ± 0.5	68.3	+2	-1
IGA [36]	29.7 ± 0.5	70.5 ± 1.2	86.2 ± 0.7	62.1	0	-1
IRM [2]	60.2 ± 2.4	67.6 ± 1.4	85.4 ± 1.2	71.1	-1	-1
MLDG [41]	32.7 ± 1.1	51.6 ± 6.1	85.4 ± 1.3	56.6	-4	-1
SagNet [49]	30.5 ± 0.7	69.3 ± 1.0	85.8 ± 1.4	61.9	+1	-2
CORAL [59]	30.0 ± 0.5	68.3 ± 1.4	86.3 ± 0.5	61.5	-1	-2
ANDMask [53]	27.2 ± 1.4	72.2 ± 1.2	86.2 ± 0.2	61.9	-2	-2
Mixup [75]	28.6 ± 1.5	66.6 ± 0.9	87.5 ± 0.5	60.6	-2	-2
RSC [31]	27.6 ± 1.8	69.7 ± 0.9	85.9 ± 0.2	61.4	+2	-3
DANN [16]	24.5 ± 0.8	68.6 ± 1.1	86.0 ± 0.4	59.7	-2	-3

Table 2. Performance of domain generalization algorithms on datasets dominated by correlation shift. Prev score indicates the ranking score produced in Table 1. Although W2D is in third place, the top three methods have the same ranking score, and the gap in averaged accuracy mainly comes from the simplest dataset CMNIST.

5.2. Effect of Whole Batch Patching Percentage κ

In Table 4, we vary κ: 0 percent means never training with the whole batch. As we increase κ, we observe higher training-validation accuracy, but lower testing-validation accuracy. This ablation study demonstrates that whole batch training can boost training validation results, while slightly decreasing the model's potential generalization ability at the same time.

5.3. Dimensions of Worst-case Training

In Table 5, we evaluate each component of W2D. Both components (sample dimension and feature dimension) are shown to outperform ERM. We believe each component can be easily plugged into other domain generalization methods and achieve consistent gains. Also, integrating both components is the best setting (W2D) for most of the diversity shift and correlation shift datasets.

Percentage	Dataset	Acc(Train-Val/Test-Val)
10	PACS	82.4 / 83.7
20	PACS	83.0 / 83.5
33	PACS	82.7 / 83.2
50	PACS	82.7 / 83.1

Table 3. Ablation study of Top Worst-case $\rho\%$. We fix other hyperparamters here and only change ρ.

Percentage	Dataset	Acc(Train-Val/Test-Val)
0	PACS	82.2 / 83.7
5	PACS	82.5 / 83.5
10	PACS	82.7 / 83.3
20	PACS	83.0 / 83.3
40	PACS	82.9 / 83.3

Table 4. Ablation study of Whole Batch Training During The Last $\kappa\%$ Epoch. We fix other hyperparameters here and only change κ.

Methods	Dataset	Acc(Train-Val/Test-Val)
ERM	PACS	81.5 / 82.2
feature-dim.	PACS	82.8 / 83.3
sample-dim.	PACS	82.2 / 83.5
W2D	PACS	83.4 / 84.0
ERM	OfficeHome	63.3 / 63.5
feature-dim.	OfficeHome	62.9 / 63.3
sample-dim.	OfficeHome	63.3 / 63.7
W2D	OfficeHome	63.5 / 63.8
ERM	TerraInc	42.6 / 43.9
feature-dim.	TerraInc	43.6 / 44.8
sample-dim.	TerraInc	42.9 / 45.1
W2D	TerraInc	44.5 / 46.3
ERM	CelebA	86.3 / 87.2
feature-dim.	CelebA	86.2 / 85.9
sample-dim.	CelebA	85.8 / 87.4
W2D	CelebA	86.5 / 87.7

Table 5. Analysis of each dimension of W2D.

5.4. Training Validation vs Testing Validation

For training-domain validation, each training domain is split into training and validation subsets. The models are trained using the training subsets and the final model is chosen as the one that maximizes the accuracy on the union of the validation subsets. Training validation is designed to apply on real-world applications. For testing-domain validation, the model is selected by maximizing the accuracy on a validation set that follows the distribution of the test domain. Testing validation is used to measure a method's highest potential generalization ability. In Table 5, we see that compared to RSC (feature-dim.), W2D tends to obtain bigger improvements when testing-domain validation is used for model selection than training-domain validation. This is mainly because worst-case training along sample dimension can increase the model's potential generalization power.

5.5. A special property of Colored MNIST

As mentioned earlier, we evaluate the results in CMNIST using the -90 as testing environment in Table 2 following Ood-Bench [74]. In this section, we report the results averaged over three environments (+90, +80 and -90) in CMNIST, which is the protocol used in DomainBed [21]. The study of these results leads us to notice a special property of CMNIST in comparison to other methods used. Then, this special property leads us to introduce a modified version of W2D that improves over ERM by a clear margin by taking advantage of this property.

Treating the -90 domain as a testing domain is considered to be the most difficult setting because the training and testing domain's distribution are totally flipped. (In comparison, the discussions of the other two testing domains, +90 and +80, are omitted as they are much simpler.) If we can train on only a small subset of the training samples that share the same distribution as that of testing, the results could be largely improved. Worst-case training along the sample dimension would be a natural solution for this problem. However, we found that a vanilla usage of this method can heavily affect training for this toy dataset.

To solve this problem, we first train a biased classifier at the beginning of training with a few epochs. The biased classifier is then fixed and used as a pre-trained classifier to select the worse-case samples. Specifically, we utilize the worse-case samples selected in each iteration by the biased classifier to train a debiased classifier. Since the distribution flips between the training and testing domain (from +80/+90 to -90), the worse-case samples selected by the pre-trained biased classifier should share a similar distribution with the samples during testing, which leads to surprisingly high performance in Table 6.

We hope this ablation study can motivate the community to rethink the evaluation method of Colored MNIST. We conjecture a more reasonable protocol is to, rather than only reporting results on -90, evaluate the methods with multiple different distribution domains: *e.g.*, averaged over +/-90, +/-70, +/-50, +/-30, +/-10 domains.

6. Discussion

Additional Benefit with Stochastic Weight Averaging Stochastic Weight Averaging (SWA) [32] is an ensemble technique that finds the solution at the center of a wide flat region of the loss landscape. It performs an equal averaging of the model parameters derived from multiple local minima during the training procedure. SWA was shown to im-

Method	Dataset	Acc(Train-Val/Test-Val)
ERM [62]	CMNIST	51.5 / 58.5
GroupDRO [57]	CMNIST	52.1 / 61.2
VREx [38]	CMNIST	51.8 / 56.3
ARM [76]	CMNIST	56.2 / 63.2
IRM [2]	CMNIST	52.0 / 70.2
RSC [31]	CMNIST	51.7 / 58.5
W2D	CMNIST	51.9 / 59.0
W2D*	CMNIST	**70.8 / 72.9**

Table 6. The CMNST results are adopted from [21] and averaged over three domains. * means modified version of W2D.

prove the performance in semi-supervised learning and domain adaptation [3].

In addition to whole batch training, SWA is another effective way to improve the model's generalization at later epoches. Intuitively, SWA is able to leverage the worse-case samples from different training stages regardless of whether the samples the model considered worse-case in previously epoches later switches to be easy ones or not. In Table 7, we notice that SWA works especially well with worse-case-based methods. For example, in PACS, W2D obtains a 1.3% improvement from SWA while ERM and feature-dim. (RSC) obtains 0.9% and 0.7% improvement, respectively.

Method	Dataset	Acc(Train-Val/Test-Val)
ERM	PACS	81.5 / 82.2
feature-dim.	PACS	82.8 / 83.3
sample-dim.	PACS	82.2 / 83.5
W2D	PACS	83.4 / 84.0
ERM(w SWA)	PACS	82.5 / 83.0
feat-dim.(w SWA)	PACS	83.5 / 83.7
sam-dim.(w SWA)	PACS	83.4 / 83.7
W2D(w SWA)	PACS	84.7 / 84.8

Table 7. W2D can further improve the performances if used together with Stochastic Weight Averaging. We apply SWA at the last 25 percent of training time and do not apply whole batch patching here.

Challenges for our method in DomainBed First, methods such as RSC [31] or the family of DRO methods [38,57] are simple heuristic extensions of ERM along the feature or sample dimension. It seems counter-intuitive that these methods cannot compete with ERM if used properly. When closely studying the experimental settings in DomainBed, first, we notice the hyperparameter range of RSC goes as high as dropping 50% of the features, and with such a high dropout rate, we find that RSC can barely learn any useful patterns. Second, DomainBed changes the default model setting of ResNet50 [24] by adding dropout [58] in the fully connected layers. The highest dropout rate in DomainBed is 50%, which might be beneficial to other algorithms but could degrade RSC's performance due to the overuse of dropout from both aspects. For the other dimension, the batchsize range goes as small as 8, which limits the potential of the DRO-family methods to use the hard samples.

Why Ood-bench? First, Ood-bench uses a ranking score to reflect a relative degree of robustness against both kinds of distribution shift instead of mean accuracy over different datasets, which is more reasonable. Several algorithms are superior to ERM in the toy case datasets, but they are still vulnerable to the distribution shift from the real data. Thus, using mean accuracy is a less meaningful way to compare these algorithms. Second, unlike DomainBed, Ood-bench uses a smaller model, ResNet18 [24], for all algorithms and datasets excluding Colored MNIST. It is known that larger models are usually more robust to distribution shift data and thus their performance may be more easily saturated on small datasets [26]. Thus, using a smaller base model on which to build on could provide a better testbed for OoD generalization of different algorithms. Third, the search space of the non-algorithm-specific hyperparameters is carefully designed, such as the learning rate. It allows each algorithm to converge during training at each run. More importantly, Ood-bench measures each method's generalization ability in a more objective and fair manner: it excludes previous domain generalization techniques that can be plugged into any algorithm, such as dropout [58].

Limitations In the more realistic dataset dominated by correlation shift, such as CelebA and NICO, although W2D achieves the best improvement among all the algorithms, it does not surpasses ERM by a statistically significant margin. Despite the high empirical performances, there are no sufficient evidence suggesting that W2D will be ideal when facing common challenge such as spurious correlations. Study to extend W2D to further overcome these challenges is likely expected in the future.

7. Conclusion

Inspired by a simple heuristic that training with a particular focus on hard-to-learn concepts will benefit the learning process, in this paper, we introduce a training heuristic method that can iteratively force the model to learn the hard-to-learn concepts on both the feature dimension and the sample dimension. We name our method W2D following the idea of "Worst-case along Two Dimensions". W2D can be directly applied to almost any model architecture, optimizer, loss, or regularization *etc*. After evaluating W2D in OoD-Bench comprehensively, we observe W2D is the only algorithm that can achieve consistently better performance than ERM on both diversity shift and correlation shift.

Acknowledgements. This work was supported in part by NSF CAREER IIS-2150012 and IIS-2204808. HW was supported by NIH R01GM114311, NIH P30DA035778, and NSF IIS1617583.

References

[1] Kei Akuzawa, Yusuke Iwasawa, and Yutaka Matsuo. Adversarial invariant feature learning with accuracy constraint for domain generalization. *arXiv preprint arXiv:1904.12543*, 2019. 2

[2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. 5, 6, 8

[3] Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. There are many consistent explanations of unlabeled data: Why you should average. *arXiv preprint arXiv:1806.05594*, 2018. 8

[4] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. *arXiv preprint arXiv:1910.02806*, 2019. 2, 3

[5] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pages 456–473, 2018. 5

[6] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010. 1, 2

[7] Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19:137, 2007. 2

[8] Gilles Blanchard, Aniket Anand Deshmukh, Urun Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *arXiv preprint arXiv:1711.07910*, 2017. 6

[9] Jonathon Byrd and Zachary Lipton. What is the effect of importance weighting in deep learning? In *International Conference on Machine Learning*, pages 872–881. PMLR, 2019. 4

[10] Fabio M Carlucci, Paolo Russo, Tatiana Tommasi, and Barbara Caputo. Agnostic domain generalization. *arXiv preprint arXiv:1808.01102*, 2018. 2

[11] Haw-Shiuan Chang, Erik Learned-Miller, and Andrew McCallum. Active bias: Training more accurate neural networks by emphasizing high variance samples. *Advances in Neural Information Processing Systems*, 30:1002–1012, 2017. 4

[12] Alexandra Chouldechova and Aaron Roth. A snapshot of the frontiers of fairness in machine learning. *Communications of the ACM*, 63(5):82–89, 2020. 3

[13] Sofien Dhouib, Ievgen Redko, and Carole Lartizien. Margin-aware adversarial domain adaptation with optimal transport. In *Thirty-seventh International Conference on Machine Learning*, 2020. 2

[14] K Anders Ericsson, Ralf T Krampe, and Clemens Tesch-Römer. The role of deliberate practice in the acquisition of expert performance. *Psychological review*, 100(3):363, 1993. 2

[15] Cangning Fan, Peng Liu, Ting Xiao, Wei Zhao, and Xianglong Tang. Domain adaptation based on domain-invariant and class-distinguishable feature learning using multiple adversarial networks. *Neurocomputing*, 411:178–192, 2020. 2

[16] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016. 2, 6

[17] 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 *International Conference on Learning Representations*, 2019. 1, 3

[18] Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant. A new pac-bayesian perspective on domain adaptation. In Maria-Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 859–868. JMLR.org, 2016. 2

[19] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *Proceedings of the IEEE international conference on computer vision*, pages 2551–2559, 2015. 2

[20] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 1

[21] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020. 3, 5, 7, 8

[22] Beining Han, Chongyi Zheng, Harris Chan, Keiran Paster, Michael R Zhang, and Jimmy Ba. Learning domain invariant representations in goal-conditioned block mdps. *arXiv preprint arXiv:2110.14248*, 2021. 2

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 1

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 8

[25] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognition*, 110:107383, 2021. 5

[26] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021. 8

[27] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. 1

[28] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*, 2019. 3

[29] Katherine Hermann, Ting Chen, and Simon Kornblith. The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33, 2020. 3

[30] Weihua Hu, Gang Niu, Issei Sato, and Masashi Sugiyama. Does distributionally robust supervised learning give robust classifiers? In *International Conference on Machine Learning*, pages 2029–2037. PMLR, 2018. 3

[31] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 124–140. Springer, 2020. 3, 4, 5, 6, 8

[32] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018. 7

[33] Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *International conference on machine learning*, pages 2525–2534. PMLR, 2018. 4

[34] Myeongjin Kim and Hyeran Byun. Learning texture invariant representation for domain adaptation of semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12975–12984, 2020. 2

[35] Pang Wei Koh, Shiori Sagawa, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021. 5

[36] Masanori Koyama and Shoichiro Yamaguchi. Out-of-distribution generalization with maximal invariant predictor. *arXiv preprint arXiv:2008.01883*, 2020. 6

[37] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[38] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pages 5815–5826. PMLR, 2021. 2, 3, 6, 8

[39] Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. Fairness without demographics through adversarially reweighted learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 728–740. Curran Associates, Inc., 2020. 3, 4

[40] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017. 5

[41] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 6

[42] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2018. 2, 6

[43] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015. 5

[44] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *COLT 2009 - The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009*, 2009. 2

[45] JH McDonald. Confounding variables. *Handbook of biological statistics, 3rd edn. Sparky House Publishing, Baltimore*, pages 24–28, 2014. 3

[46] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021. 3

[47] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *The IEEE International Conference on Computer Vision (ICCV)*, volume 2, page 3, 2017. 2

[48] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18, 2013. 1, 2

[49] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap via style-agnostic networks. *arXiv preprint arXiv:1910.11645*, 2(7):8, 2019. 6

[50] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: Training debiased classifier from biased classifier. In *Advances in Neural Information Processing Systems*, 2020. 2, 3, 4

[51] Hongseok Namkoong and John C Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 3

[52] A Tuan Nguyen, Toan Tran, Yarin Gal, and Atılım Güneş Baydin. Domain invariant representation learning with domain density transformations. *arXiv preprint arXiv:2102.05082*, 2021. 2

[53] Giambattista Parascandolo, Alexander Neitz, Antonio Orvieto, Luigi Gresele, and Bernhard Schölkopf. Learning explanations that are hard to vary. *arXiv preprint arXiv:2009.00329*, 2020. 6

[54] Mohammad Mahfujur Rahman, Clinton Fookes, and Sridha Sridharan. Discriminative domain-invariant adversarial net-

work for deep domain generalization. *arXiv preprint arXiv:2108.08995*, 2021. 2

[55] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016. 2

[56] Kate Saenko, Brian Kulis, Mario Fritz, and T Darrell. Adapting visual category models to new domains. In *Proceedings of the European conference on computer vision*, 2010. 2

[57] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019. 2, 3, 4, 6, 8

[58] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. 8

[59] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016. 6

[60] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. 1

[61] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011. 3

[62] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999. 5, 6, 8

[63] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017. 5

[64] Tyler Vigen. *Spurious correlations*. Hachette books, 2015. 3

[65] Haohan Wang, Songwei Ge, Zachary C. Lipton, and Eric P. Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 10506–10518, 2019. 2, 3

[66] Haohan Wang, Zexue He, Zachary C. Lipton, and Eric P. Xing. Learning robust representations by projecting superficial statistics out. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. 2, 3

[67] Haohan Wang, Zeyi Huang, Xindi Wu, and Eric P Xing. Squared ℓ_2 norm as consistency loss for leveraging augmented data to learn robust and invariant representations. *arXiv preprint arXiv:2011.13052*, 2020. 3

[68] Haohan Wang, Zeyi Huang, Hanlin Zhang, and Eric Xing. Toward learning human-aligned cross-domain robust models by countering misaligned features. *arXiv preprint arXiv:2111.03740*, 2021. 1, 2, 3

[69] Haohan Wang, Aaksha Meghawat, Louis-Philippe Morency, and Eric P Xing. Select-additive learning: Improving generalization in multimodal sentiment analysis. pages 949–954, 2017. 2

[70] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8684–8694, 2020. 1

[71] Z Wang and X Cheng. A dictionary approach to domain-invariant learning in deep networks. *NeurIPS*, 2020. 2

[72] Yifan Wu, Ezra Winston, Divyansh Kaushik, and Zachary Lipton. Domain adaptation with asymmetrically-relaxed distribution alignment. In *International Conference on Machine Learning*, pages 6872–6881, 2019. 2

[73] Jiabin Xue, Jiqing Han, Tieran Zheng, Jiaxing Guo, and Boyong Wu. Hard sample mining for the improved retraining of automatic speech recognition. *arXiv preprint arXiv:1904.08031*, 2019. 4

[74] Nanyang Ye, Kaican Li, Lanqing Hong, Haoyue Bai, Yiting Chen, Fengwei Zhou, and Zhenguo Li. Ood-bench: Benchmarking and understanding out-of-distribution generalization datasets and algorithms. *arXiv preprint arXiv:2106.03721*, 2021. 1, 3, 5, 7

[75] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 6

[76] Marvin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: A meta-learning approach for tackling group distribution shift. *arXiv preprint arXiv:2007.02931*, 2020. 6, 8

[77] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael I. Jordan. Bridging theory and algorithm for domain adaptation. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7404–7413. PMLR, 2019. 2

[78] Han Zhao, Remi Tachet des Combes, Kun Zhang, and Geoffrey J. Gordon. On learning invariant representations for domain adaptation. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7523–7532. PMLR, 2019. 2

[79] Shanshan Zhao, Mingming Gong, Tongliang Liu, Huan Fu, and Dacheng Tao. Domain generalization via entropy regularization. *Advances in Neural Information Processing Systems*, 33, 2020. 6