# On Feature Learning in the Presence of Spurious Correlations

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

Deep learning classifiers are known to rely on spurious correlations — patterns which are semantically irrelevant but predictive of the target on the training data. In this paper we explore the quality of feature representations learned by standard empirical risk minimization (ERM) and specialized group robustness training, as well as the effect of key factors including architecture, pre-training strategy, regularization and others. Following recent work on Deep Feature Reweighting (DFR), we evaluate the feature representations by re-training the last layer of the model on a held-out set where the spurious correlation is broken. Through this procedure, we reveal how much information about the core semantic features is contained in the learned representations. On multiple vision and NLP problems, we show that the features learned by simple ERM are highly competitive with the features learned by specialized group robustness methods targeted at reducing the effect of spurious correlations. Moreover, we show that the quality of learned feature representations is largely affected by the choice of data augmentation, model architecture and pre-training strategy. On the other hand, we find that strong regularization, and long training are generally not helpful for improving the learned feature representations. Finally, using insights from our analysis, we significantly improve upon the best results reported in the literature on the popular Waterbirds, CelebA hair color prediction and WILDS-FMOW problems, achieving 97%, 92% and 50% worst-group accuracies respectively.

## 1 Introduction

Spurious features, patterns that are predictive of the target on the training data, but irrelevant to the true labeling process, are a significant problem in modern deep learning [17]. Neural network classifiers are known to rely on the spurious features such as backgrounds [87] or texture [18] in computer vision and specific words or constructions in natural language processing [52, 20]. In extreme cases, neural networks completely ignore task-relevant *core* features and only use spurious features in their predictions [91, 73].

In recent work, Kirichenko et al. [37] showed that, surprisingly, standard Empirical Risk Minimization (ERM) learns a high-quality representation of the core features on datasets with spurious correlations, even when the model primarily relies on spurious features to make predictions. Moreover, they showed that it is often possible to recover state-of-the-art performance on benchmark problems by simply retraining the last layer of the model on a small held-out dataset where the spurious correlation does not hold. This procedure is called Deep Feature Reweighting (DFR). However, while Kirichenko et al. [37] showed that DFR with ERM training can match state-of-the-art worst group performance, they did not attempt to combine DFR with group robustness methods, hypothesizing that such combination may further improve performance.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

In this paper, we provide an in-depth study of the factors that affect the quality of learned representations in the presence of spurious correlations. Following Kirichenko et al. [37], we break the problem of learning a robust classifier into two tasks: extracting feature representations and training a linear classifier on these features. In order to study the feature learning in isolation, we use the DFR procedure to learn an optimal linear classifier on the feature representations, and evaluate the features learned with different training methods, neural network architectures, and hyper-parameters.

First, on a range of problems with spurious correlations we show that while specialized group robustness methods such as group distributionally robust optimization (Group DRO) [70] can significantly outperform the standard ERM training, the quality of the features learned by ERM is highly competitive, and by applying the DFR procedure to the features learned by ERM and Group DRO we achieve similar performance. Furthermore, we show that the performance improvements of Group DRO are largely explained by the better weighting of the learned features in the last classification layer, and not by learning a better representation of the core features. This observation has high practical significance, as the problem of training the last layer of the model is much simpler both conceptually and computationally than training the full model to avoid spurious correlations [37].

Next, focusing on the ERM training, we explore the effect of model class, pretraining strategy and regualarization on feature learning. We find a linear dependence between the in-distribution accuracy of the model and the worst group accuracy after applying DFR, meaning that good generalization implies good feature learning, even in the presence of spurious features. Further, we show that the pre-training strategy and data augmentation have a very significant effect on the quality of the learned features, while regularization and length of training have a limited effect for most benchmarks.

Finally, by finetuning a state-of-the-art ConvNext model [48], we significantly outperform the best reported results on the popular Waterbirds [70], CelebA hair color and WILDS FMOW [40] spurious correlation benchmarks, using only simple ERM training followed by DFR.

## 2 Related Work

Numerous works describe how neural networks can rely on spurious correlations in real world problems. In vision, neural networks can learn to rely on an image's background [87, 70, 57], secondary objects [42, 67, 76, 74, 2, 56], object textures [18] and other semantically irrelevant features [8, 45]. Spurious correlations are especially problematic in high-risk domains such as medical imaging, where it was shown that neural networks can use hospital-specific metal tokens [91] or cues of disease treatment [61] rather than symptoms to perform automated diagnosis on chest X-ray images. Spurious features are also extremely prevalent in NLP, where models can achieve good performance on benchmarks without properly solving them [60, 20, 34, 52]. For a comprehensive survey of the area, see Geirhos et al. [18].

Because of the high practical significance of spurious correlations, many *group robustness* methods have been proposed. These methods aim to reduce the reliance of deep learning models on spurious correlations and improve worst group performance. Group DRO [70] is the state-of-the-art group robustness method, which minimizes the worst-group loss instead of the average loss. Other works focus on automatically identifying the minority group examples [46, 58, 12], learning several diverse classifiers that use different features [44, 62, 79] or using partially available group labels [77, 59]. Group subsampling was shown to be a strong baseline for some benchmarks [32, 72].

In this work, we focus on *feature learning* in the presence of spurious correlations. Hermann and Lampinen [29] perform a conceptually similar study, but focusing on synthetic datasets. Similar to their work, we explore how well the different features of the data can be decoded from the features learned by deep neural networks. Lovering et al. [50] explore the factors which affect the extractability of features after pre-training and fine-tuning of NLP models. Kaushik et al. [33] construct counterfactually augmented sentiment analysis data (CAD) and show that combining CAD with the original data removes reliance on spurious correlations. Kaushik et al. [35] explain their observations and show that while adding noise to causal features degrades in-distribution and out-of-distribution performance, adding noise to non-causal features improves robustess. Eisenstein [16] and Veitch et al. [80] formally define and study spurious features in NLP from the perspective of causality. Hermann et al. [28] explore the feature learning in the context of texture bias [18], finding that data augmentation has a profound effect on the texture bias while architectures and training objectives have a relatively small effect. Suvra Ghosal et al. [78] show that Vision Transformer models pretrained on

ImageNet22k [41] significantly outperform standard CNN models on several spurious correlation benchmarks. Finally, Kirichenko et al. [37] show that models trained with standard ERM training often learn high-quality representations of the core features, and proposed the DFR procedure (see Section 3) which we use extensively in this paper.

While we build on the observations of Kirichenko et al. [37], our work provides profound new insights and greatly expands on the scope of their work. In particular, we explore the feature representations learned by methods beyond standard ERM, explore the effect of model architecture, pre-training, regularization and data augmentation. We also extend our analysis beyond the standard spurious correlation benchmarks studied by Kirichenko et al. [37],

by considering NLP problems as well as challenging real world satellite imaging and chest X-ray datasets.

## 3 Background

**Preliminaries.** We consider classification tasks with inputs $x \in \mathcal{X}$ and classes $y \in \mathcal{Y}$. We assume that the data distribution consists of groups $\mathcal{G}$ which are not equally represented in the train data. The distribution of groups can change between the training and test distributions, with majority groups becoming less common or minority groups becoming more common. Because of the imbalance in train data, models trained with ERM often have a gap between average and worst group performance on test. Throughout this paper, we will be studying *worst group accuracy* (*WGA*), i.e. the lowest test accuracy across all the groups $\mathcal{G}$. For most problems considered in this paper, we assume that each data point has an attribute $s \in \mathcal{S}$ which is spuriously correlated with the label $y$, and the groups are defined by a combination of the label and spurious attribute: $\mathcal{G} \in \mathcal{Y} \times \mathcal{S}$. In test distribution we might find that $s$ is no longer correlated with $y$, and thus a model that has learned to rely on the spurious feature $s$ during training will perform poorly at test time. Models that rely on the spurious features will typically achieve poor worst group accuracy, while models that rely on core features will have more even accuracies across the groups. In Appendix A, we describe the groups, spurious and core features in the datasets that we use in this paper. In order to perform controlled experiments, we assume that we have access to the spurious attributes $s$ (or group labels) for train or validation data, which are used for training of group robustness baselines and feature quality evaluation. However, we emphasize that our results on the features learned by ERM hold generally, even when spurious attributes are unknown, as ERM does not use the information about the spurious features: we only use the spurious attributes to perform analysis.

**Deep feature reweighting.** Suppose we are given a model $m : \mathcal{X} \to \mathcal{C}$, where $\mathcal{X}$ is the input space and $\mathcal{C}$ is the set of classes. Kirichenko et al. [37] assume that the model $m$ consists of a feature extractor (typically, a sequence of convolutional or transformer layers) followed by a classification head (typically, a single linear layer): $m = h \circ e$, where $e : \mathcal{X} \to \mathcal{F}$ is a feature extractor and $h : \mathcal{F} \to \mathcal{C}$ is a classification head. They discard the classification head, and use the feature extractor $e$ to compute the set of embeddings $\hat{\mathcal{D}}_e = \{(e(x_i), y_i)\}_{i=1}^n$ of all the datapoints in the reweighting dataset $\hat{\mathcal{D}}$; the reweighting dataset is used to retrain the last layer of the model, and contains group-balanced data where the spurious correlations does not hold. Finally, they train a logistic regression classifier $l : \mathcal{F} \to \mathcal{C}$ on the dataset $\hat{\mathcal{D}}_e$. Then, the final model used on new test data is given by $m_l = l \circ e$. Thoughout this paper, we use a group-balanced held-out dataset (subset of the validation dataset where each group has the same number of datapoints) as the reweighting dataset $\hat{\mathcal{D}}$; Kirichenko et al. [37] denote this variation of the method as $\text{DFR}^{\text{Val}}_{\text{Tr}}$ .

## 4 Experimental Set-Up and Evaluation Procedure

Here, we describe the datasets, models and evaluation procedure that we use throughout the paper.

**Datasets.** In order to cover a broad range of practical scenarios, we consider four image classification and two text classification problems.

- *Waterbirds* [70] is a binary image classification problem, where the class corresponds to the type of the bird (landbird or waterbird), and the background is spuriously correlated with the class. Namely, most landbirds are shown on land, and most waterbirds are show over water.

- *CelebA* hair color [47] is a binary image classification problem, where the goal is to predict whether a person shown in the image is blond; the gender of the person serves as a spurious feature, as $94\%$ of the images with the "blond" label depict females.
- WILDS-*FMOW* [11, 40, 71] is a satellite image classification problem, where the classes correspond to one of 62 land use or building types, and the spurious attribute $s$ corresponds to the region (Africa, Americas, Asia, Europe, Oceania or Other; the "Other" region is not used in the evaluation). We note that for the FMOW datasets the groups $\mathcal{G}_s$ are defined by the value of the spurious attribute, and not the combination of the spurious attribute and the class label $\mathcal{G}_{y,s}$, as described in Section 3. For more details, please see Appendix A.
- *CXR-14* [83] is a dataset with chest X-ray images for which we focus on a binary classification of pneumothorax prediction. Oakden-Rayner et al. [61] showed that there is a hidden stratification in the dataset such that most images from the positive class contain a chest drain, which is a non-causal feature related to treatment of the disease. While for all other benchmarks we report WGA on test data, for this dataset, following prior work [61, 44, 65], we report worst group AUC because of the heavy class imbalance.[1]
- *MultiNLI* [85, 70] is a text classification problem, where the task is to classify the relationship between a given pair of sentences as a contradiction, entailment or neither of them. In this dataset, the presence of negation words (e.g. "never") in the second sentence is spuriously correlated with the "contradiction" class.
- *CivilComments* [7, 40] is a text classification problem, where the goal is to classify whether a given comment is toxic. We follow Idrissi et al. [32] and use the coarse version of the dataset, where the spurious attribute is $s = 1$ if the comment mentions at least one of the following categories: male, female, LGBT, black, white, Christian, Muslim, other religion; otherwise, the spurious label is $0$. The presence of the eight categories above is spuriously correlated with the comment being classified as toxic.

The Waterbirds, CelebA, CivilComments and MultiNLI datasets are commonly used to benchmark the performance of group robustness methods [see e.g. 32, 46, 59]. The FMOW and CXR-14 datasets present challenging real-world problems with spurious correlations. In these datasets, the inputs do not resemble natural images from datasets such as ImageNet [69], so models have to learn the relevant features from data to achieve good performance, and cannot simply rely on transfer. We provide detailed descriptions of the data and show example datapoints in Appendix A.

**Models.** Following prior work [70, 32] we use a ResNet-50 [24] model pretrained on ImageNet1k [69] on Waterbirds, CelebA and FMOW. For the NLP problems, we use a BERT model [13] pretrained on Book Corpus and English Wikipedia data. On the CXR-14, following prior work [e.g. 65, 61, 44] we use a DenseNet-121 model [30] pretrained on ImageNet1k. In Section 6, we provide an extensive study of the effect of architecture and pretraining on the image classification problems.

**Evaluation strategy.** We use DFR to evaluate the quality of the learned feature representations, as described in Section 3. Intuitively, with this procedure, we measure how well the core features can be decoded from the learned representations. In some of the experiments, we also train a classifier to predict the spurious attribute $s$ instead of the class label $y$. Using this classifier, we can evaluate the decodability of the spurious feature from the learned feature representation. We refer to this procedure as $s$-DFR and the corresponding worst group accuracy (in predicting the spurious attribute $s$) as DFR $s$-WGA. Additionally, we evaluate the worst group accuracy and mean[2] accuracy of the base model without applying DFR, which we refer to as *base WGA* and *base accuracy* respectively.

## 5 ERM vs Group Robustness Training

Multiple methods have been proposed for training classifiers which are more robust to spurious correlations, with significant improvements in worst group accuracy compared to standard ERM. In

[1] We take the minimum out of the two scores on test: AUC for classifying negative class against positive examples with chest drain and AUC for negative class against positive without chest drain.

[2] Following Sagawa et al. [70] and Kirichenko et al. [37], we evaluate the mean accuracy according to the group distribution in the training set. This way, mean accuracy represents the in-distribution generalization of the model.

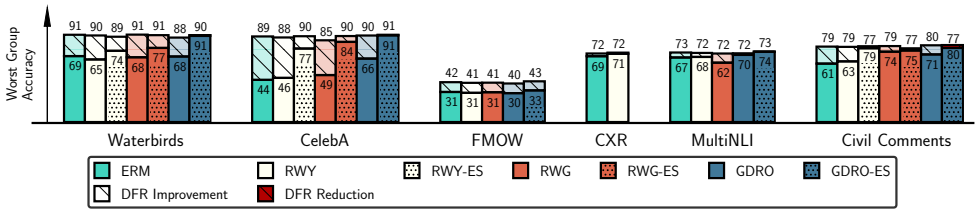



Figure 1: **ERM vs group robustness methods.** Performance of ERM on group robustness methods on vision and NLP benchmark problems. *ES* stands for early stopping. For each method on each dataset we report the base model worst group accuracy (shown with the number inside each bar), and the worst group accuracy after applying DFR (shown above each bar). On CXR-14, we report worst group AUC. While on many datasets the base performance of ERM is much worse compared to group robustness methods, after applying, the performance of the different methods is similar after we apply DFR, suggesting that the strong performance of group robustness methods is largely caused by better weighting of the learned features rather than better feature representations.

this section, we use DFR to investigate whether the improvements of group robustness methods are caused by better feature representations or by better weighting of the learned features.

**Methods.** We consider 4 methods for learning the features. *ERM* or Empirical Risk Minimization is the standard training on the original training data. *RWG* reweights the loss on each of the groups according to the size of the group and *RWY* reweights the loss on each class according to the size of the class [32]. *Group DRO* [70] is a state-of-the-art method which uses the group information on the training data to minimize the worst group loss instead of the average loss. Group DRO is often considered as an oracle method or upper-bound on the worst group performance under spurious correlations [46, 12].

On the CXR dataset the group labels are not available on the train data, so we cannot apply RWG or Group DRO; on this dataset we compare ERM to RWY. On several datasets, the performance of RWY and RWG methods deteriorates during training. For these datasets, we additionally report the results for the checkpoint obtained with early stopping (RWY-ES and RWG-ES).

**Hyper-parameters selection.** We train ERM models, RWG and RWY with the same hyper-parameters shared between all the image datasets (apart from batch size which is set to 32 on Waterbirds, and 100 on the other datasets) and between the natural language datasets. We do not tune the hyper-parameters of these methods for worst group accuracy. For Group DRO, we run a grid search over the values of the generalization adjustment $C$, weight decay and learning rate hyper-parameters, and select the best combination according to the worst-group accuracy on validation data with early stopping. For details, please see Appendix B.

### 5.1 Results

We compare feature learning methods on all datasets in Figure 1. As expected, the worst group accuracy of group robustness methods is significantly better than the ERM worst group accuracy on most datasets. For example, on Waterbirds, ERM only gets $68.8\%$ WGA, while group DRO with early stopping gets $90.6\%$. However, after applying DFR the performance of ERM and Group DRO is very close, with a slight advantage for ERM ($91.1\%$ for ERM and $90\%$ for Group DRO), and similar observations hold on all datasets.

The results for RWG and RWY are analogous. Namely, when combined with early stopping, these methods outperform ERM on base model performance. Once we apply DFR, however, the gap in performance between the methods becomes very small. In fact, on all the datasets and for all the methods the improvement in worst group accuracy from using any of the considered group robustness methods compared to ERM does not exceed $1$–$2\%$ after applying DFR.

These results suggest that the improvements over ERM in base model performance for methods such as Group DRO and RWG are largely the result of better weighting of the learned features rather than learning better representations of the core features. Indeed, if the core feature was better represented

by group robustness methods, we would expect to see a significant improvement over ERM after applying DFR.

This observation is significant both practically and scientifically. Practically, the problem of training the last layer is simpler, more data efficient and less compute intensive than retraining the full model [37]. Our results suggest that practitioners can primarily focus on retraining the last layer, as training the feature extractor model with group robustness methods does not provide significant improvements. Scientifically, robustness to spurious correlations is often implicitly or explicitly associated with the quality of learned feature representations [e.g. 3, 4, 68, 94, 44, 93]. Our results suggest that the quality of feature representations is not significantly affected by Group DRO, refining our understanding of group robustness training and representation learning in the presence of spurious correlations.

**Effect of early stopping.** Early stopping is crucial to achieving strong base model performance with RWY, RWG and Group DRO on many of the datasets. In Figure 1 we report the results both with and without early stopping on datasets where the validation worst group accuracy significantly degrades over the course of training. Generally, early stopping does not appear to significantly improve the quality of the learned feature representations even in these problems: after applying DFR, methods with and without early stopping achieve similar performance. For ERM, we explore the DFR WGA performance as a function of the training step in Section 7 also finding that the length of training has a relatively small effect on the final DFR WGA.

**Group DRO analysis.** In Figure 2 we report the worst group accuracy of multiple Group DRO runs before and after applying DFR. For each run, we evaluate the best checkpoint according to validation accuracy (i.e. the checkpoint selected by early stopping) and the last checkpoint saved after a fixed number of epochs. We observe that while the last checkpoints perform poorly in terms of base WGA in most runs, DFR can significantly improve their performance, removing the need for early stopping. Interestingly, we find that the best performing Group DRO models cannot be improved by DFR on each of the datasets. This result suggests that the weighting of the features learned by Group DRO is already close to optimal, again indicating that the success of Group DRO can largely be attributed to learning a better weighting for the features in the last linear layer, rather than learning better features.

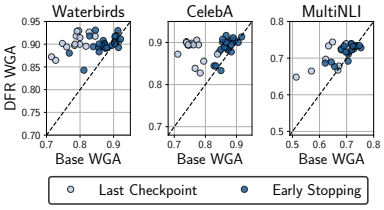



Figure 2: **Group DRO.** Base WGA and DFR WGA for a large set of Group DRO runs.

## 6 Effect of the Base Model

Most of the prior work on spurious correlation considers a fixed model class for each problem: for example, on Waterbirds and CelebA datasets almost all the papers use a ResNet-50 base model pre-trained on ImageNet1k [e.g. 70, 46, 44, 37, 77, 59, 94]. Recently, Suvra Ghosal et al. [78] showed that vision transformer models [14] may provide better robustness to spurious correlations if pretrained on a large dataset. Here, we perform a systematic large-scale evaluation of the effect of base model choice on the quality of learned feature representations.

### 6.1 Effect of the architecture and pretraining strategy

In Figure 3, we plot the base model mean and worst group accuracy and DFR worst group accuracy for a wide range of models and pretraining strategies on each of the four image classification datasets. We provide complete results, model descriptions and training hyper-parameters in Appendix C. We train a total of 78 models on Waterbirds, 78 on CelebA, 40 on FMOW and 40 on CXR.

**Accuracy on the line.** Miller et al. [55] showed that for many distribution shifts in practice the out-of-distribution performance is highly correlated with the in-distribution generalization performance. In Figure 3 (left 4 panels), for each dataset we show scatter plots of base model mean accuracy vs base model worst group accuracy, analogously to Miller et al. [55]. For FMOW (which was also considered by Miller et al. [55]) we observe a linear correlation between the base model mean and worst group accuracies. On CXR, the dependence is also roughly linear. However, both on CelebA and on WaterBirds, the correlation does not appear entirely linear. In particular, on waterbirds there is a large number of models that have similar worst group accuracy $\approx 20\%$, for which there appears

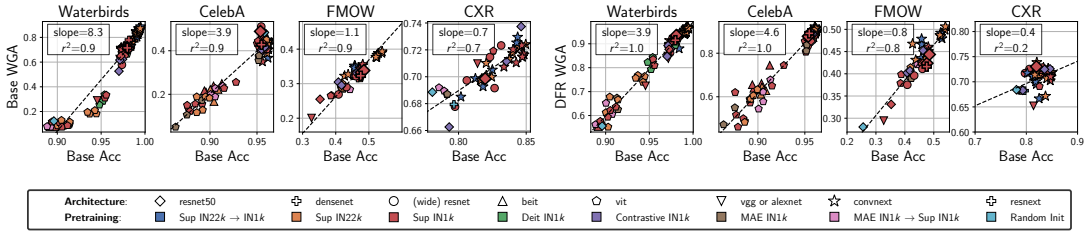



Figure 3: **Effect of the model architecture and pretraining.** Base model in-distribution accuracy plotted against base worst-group accuracy (left 4 panels) and DFR worst group accuracy (right 4 panels). For each panel, we additionally estimate the a linear slope and the $r^2$ score for the linear fit to the data. On all datasets other than CXR, the DFR WGA is linearly correlated with base model in-distribution performance. For the base model WGA, the correlation with in-distribution performance is not entirely linear. Generally, models with better in-distribution performance provide better worst-group performance and better core feature representation.

to be little correlation between the Base model WGA and mean accuracy. On CelebA, the same phenomenon occurs for the best performing models, with WGA between $40\%$ and $50\%$.

**DFR accuracy is on the line.** Next, in the right 4 panels of Figure 3, for each of the datasets we report the base model mean accuracy vs DFR WGA. On all datasets other than CXR[3], we observe a high linear correlation between the metrics, including the Waterbirds and CelebA. In particular, the models with the best mean (in-distribution) accuracy also achieve the best DFR WGA. Note that this is not the case for the base model WGA on CelebA, where the best base model WGA is 51% achieved by a ResNet-101 model pretrained on ImageNet1k; this model only achieves mean accuracy of 95.45% compared to $96.2\%$ accuracy for the ConvNext XLarge model. The results in Figure 3 confirm that models that achieve better accuracy on the training data distribution generally learn better representations of the core features, and provide better worst group accuracies with DFR.

**ERM features are sufficient for SOTA performance.** The DFR WGA results for the Waterbirds, CelebA and FMOW datasets significantly improve upon the previous best reported results, to the best of our knowledge. In particular, the ConvNext Large model pretrained on ImageNet22k with ImageNet1k fineruning achieves 97.2% DFR WGA on Waterbirds and 92.2% on CelebA; on FMOW, we only considered smaller ConvNext versions due to computational constraints, still achieving 50.6% DFR WGA with ConvNext Small pretrained on ImageNet22k; the current best results on the WILDS leaderboard for this dataset are $47.6\%$ followed by $35.5\%$ [40]. We note that our DFR evaluation uses the validation set to train the last layer of the model, similarly to e.g. Nam et al. [59], and unlike most standard group robustness methods which only use the validation set to tune the parameters. However, the results presented in this section prove that standard ERM with a strong pretrained model can achieve outstanding results on the robustness benchmarks, significantly improving upon specialized group robustness methods using a weaker model.

### 6.2 Does training on the target data improve features?

Above, we have shown that model architecture and pretraining have a large effect on the quality of the learned feature representations, as measured by DFR WGA. It is then natural to ask how much of the feature learning happens during training on the target data, and how much is simply transferred from the pretraining task. To answer this question, for the models trained in the previous section, we run the DFR evaluation on the initial weights of the models, before training on the target data. We report the DFR WGA and DFR $s$-WGA results in Figure 4.

Surprisingly, we find that on Waterbirds, for most models the improvement from training on the target data is very small, if any. For example, for the ImageNet1k-pretrained ResNet-50 model used by most works on this dataset, we get $88.2\%$ worst group accuracy by simply training the last layer on the validation data with DFR. After finetuning on the target data, we can achieve $92.9\%$ DFR WGA.

[3] On CXR, we found that the base model is often hard to improve with DFR, suggesting that it already learns to weight the features correctly. We discuss the possible reasons and the nature of spurious features on this dataset in Appendix A.2.

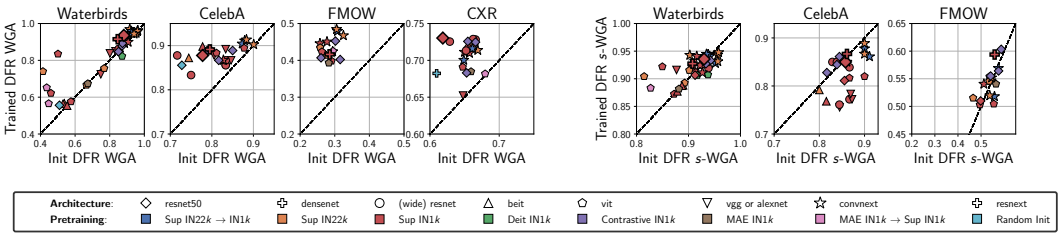



Figure 4: **Effect of training on target data.** DFR WGA (left 4 panels) and DFR $s$-WGA (right 3 panels) before and after training. On Waterbirds, the results do not significantly improve from training on the target data. On the other datasets, the performance improves consistently after training on the target data. Interestingly, on CelebA the DFR $s$-WGA decreases during training for many models, meaning that the spurious gender feature becomes less predictable from the learned representations.

Furthermore, with the ConvNext Large model pretrained on ImageNet22k, we get $94\%$ DFR WGA *without training on the waterbirds data*. This strong performance is not particularly surprising, as ImageNet22k has many of the Waterbirds bird types as classes. The perfromance is almost unchanged by training on the target data: DFR WGA after training is $94.3\%$. From these results, we can conclude that Waterbirds performance should not be used as a primary metric for feature learning performance, especially if large-scale pretraining is used! Indeed, it is possible to achieve outstanding performance, exceeding the previously reported state-of-the-art, without training the features on the target data.

On the other datasets, the feature learning is more pronounced: the DFR WGA improves after training for all the models considered. However, on CelebA it is still possible to achieve $88.3\%$ WGA without training on the target data, with ConvNext XLarge pretrained on ImageNet22k.

**Is the spurious feature representation improved during training?** We additionally explore the quality of representation of the spurious feature via DFR $s$-WGA. We show the results in Figure 4(right). Interestingly, on CelebA the spurious gender feature becomes less predictable during training for many of the models. On FMOW and CelebA there is no consistent trend, and the spurious feature does not become significantly more or less predictable during training.

### 6.3 Effect of pretraining strategy

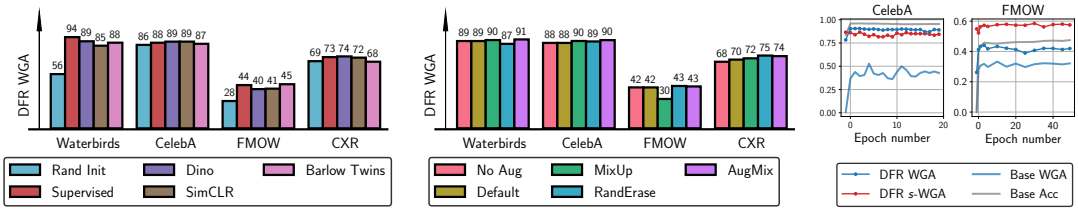
Finally, using a fixed ResNet-50 model architecture we evaluate the effect of pretraining strategy. On each of the four datasets, we train a ResNet-50 model initialized with (1) random initialization, (2) supervised pretraining, (3) DINO pretraining [9], (4) SimCLR pretraining [10] and (5) Barlow Twins pretraining [90] on ImageNet1k. We report the results in Figure 5(a). On Waterbirds and FMOW datasets, the randomly initialized model does not provide competitive performance. On CelebA, it still underperforms the pretrained models, but the gap is much smaller. Among all pretraining methods, supervised is preferable across the board, but the contrastive methods are highly competitive.

## 7 Effect of Regularization

Regularization is used to combat overfitting, including reliance on spurious features. We consider three regularization techniques: weight decay, data augmentation and reduced training length.

**Effect of weight decay.** Using a ResNet-50 model pretrained on ImageNet1k, we run training with a range of weight decay values on each of the four datasets, and report the results in Figure 6. For the base model WGA performance, it is generally helpful to set the weight decay to non-zero values. Indeed, on CelebA the base model WGA for $0$ weight decay is the worst across all weight decay values, losing to the best weight decay by $\approx 5\%$. However, the $0$ weight decay model is in fact the best model according to DFR WGA! On other datasets weight decay $0$ also provides competitive performance. FMOW is the only dataset considered where non-zero weight appears to be helpful for learning high quality representations of the core features.

**Effect of Data Augmentation.** Next, we consider 5 data augmentation policies: (1) no augmentations, (2) default augmentations, i.e. random crops and horizontal flips, (3) MixUp [92] combined with default augmentations, (4) Random Erasing [95], and (5) AugMix [27]. We train a ResNet-50




(a) Pretraining (b) Data Augmentation (c) Training length

Figure 5: **Effect of pretraining and data augmentation.** All the results shown use a ResNet50 architecture except for CXR results in panel (b), which use a DenseNet-121 model pretrained on ImageNet. **(a)** Supervised pretraining provides the best performance across the board, but contrastive methods are competitive. Randomly initialized model performs poorly. **(b)** AugMix augmentation policy consistently provides strong performance across all datasets, while random erasing and mixup hurt performance on Waterbirds and FMOW respectively. **(c)** The DFR WGA converges after only a few epochs of training on the target dataset, longer trainign does not hurt or improve performance.

model pretrained on ImageNet1k on each of the four datasets with each of the augmentation policies, and report the results in Figure 5(b). Augmix provides the best performance on each dataset. However, we find that data augmentation is generally not required to achieve strong performance on any of the datasets: the model trained without augmentations is competitive across the board. Moreover, data augmentation can *hurt* the learned features. For example, while MixUp is helpful on Waterbirds and CelebA, it significantly hurts the preformance on FMOW, with $30\%$ DFR WGA compared to $42\%$ for the model trained without augmentation. We hypothesize that on the FMOW dataset, which contains highly detailed satellite images, mixing the images makes training overly challenging.

Similarly, Random Erasing hurts the DFR WGA on Waterbirds. We hypothesize that the randomly erased image block is more likely to fully cover the bird features than the background features, as the bird occupies a small fraction of the image relative to the background. Consequently, the model is incentivised to focus on the spurious feature: the model trained with Random Erasing learned the highest quality representation of the spurious feature across all augmentations with DFR $s$-WGA of $91.8 \pm 0.1\%$ across 3 runs, compared to e.g. $91.4 \pm 0.5\%$ for the model trained with no augmentation.

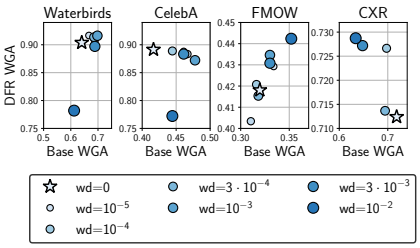



Figure 6: **Effect of weight decay.** While 0 weight decay underperfoorms in base WGA on Waterbirds and CelebA, it provides near-optimal DFR WGA.

**Effect of training length.** Finally, we study how the quality of feature representations changes during training. In Figure 5(c) we plot the base model WGA and mean accuracy as well as the DFR WGA and DFR $s$-WGA as a function of the training epoch on CelebA and FMOW. The observations on the other datasets, including NLP, are analogous and we provide the corresponding plots in Appendix D. Surprisingly, we find that on all datasets, the quality of the core feature representations as measured by DFR WGA improve in the first epoch of training, and then stay roughly constant. On all datasets, training for longer than 5 epochs does not help or hurt the performance.

## 8 Takeaways and Conclusion

Spurious correlations remain an important challenge for modern deep learning. In this paper, we have shown that standard ERM training learns feature representations that are similarly predictive of the core features as the features learned by state-of-the-art group robustness methods such as group DRO. Furthermore, we show that on many practical datasets, the quality of the learned representation of the core features improves linearly with the in-distribution generalization. Using this intuition, we significantly improve upon the best worst group accuracy results reported in the literature using the features learned by standard ERM. We hope that our observations will be useful to both practitioners who strive to achieve the best performance in the presence of spurious correlations, and researchers, who study representation learning and robustness to spurious correlations.

# References


[1] Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. (2018). A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69. PMLR.

[2] Alcorn, M. A., Li, Q., Gong, Z., Wang, C., Mai, L., Ku, W.-S., and Nguyen, A. (2019). Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4845–4854.

[3] Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. (2019). Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

[4] Bahng, H., Chun, S., Yun, S., Choo, J., and Oh, S. J. (2020). Learning de-biased representations with biased representations. In *International Conference on Machine Learning*, pages 528–539. PMLR.

[5] Bao, H., Dong, L., and Wei, F. (2021). Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*.

[6] Biewald, L. (2020). Experiment tracking with weights and biases. Software available from wandb.com.

[7] Borkan, D., Dixon, L., Sorensen, J., Thain, N., and Vasserman, L. (2019). Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion proceedings of the 2019 world wide web conference*, pages 491–500.

[8] Brendel, W. and Bethge, M. (2019). Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *arXiv preprint arXiv:1904.00760*.

[9] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660.

[10] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

[11] Christie, G., Fendley, N., Wilson, J., and Mukherjee, R. (2018). Functional map of the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

[12] Creager, E., Jacobsen, J.-H., and Zemel, R. (2021). Environment inference for invariant learning. In *International Conference on Machine Learning*, pages 2189–2200. PMLR.

[13] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[14] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

[15] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. (2012). Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226.

[16] Eisenstein, J. (2022). Informativeness and invariance: Two perspectives on spurious correlations in natural language. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4326–4331.

[17] Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. (2020). Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

[18] Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. (2018). Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*.

[19] Goyal, P., Duval, Q., Reizenstein, J., Leavitt, M., Xu, M., Lefaudeux, B., Singh, M., Reis, V., Caron, M., Bojanowski, P., Joulin, A., and Misra, I. (2021). Vissl. https://github.com/facebookresearch/vissl.

[20] Gururangan, S., Swayamdipta, S., Levy, O., Schwartz, R., Bowman, S. R., and Smith, N. A. (2018). Annotation artifacts in natural language inference data. *arXiv preprint arXiv:1803.02324*.

[21] Hardt, M., Price, E., and Srebro, N. (2016). Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29.

[22] Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825):357–362.

[23] He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. (2021a). Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*.

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

[25] He, P., Gao, J., and Chen, W. (2021b). Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

[26] He, P., Liu, X., Gao, J., and Chen, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

[27] Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., and Lakshminarayanan, B. (2019). Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*.

[28] Hermann, K., Chen, T., and Kornblith, S. (2020). The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33:19000–19015.

[29] Hermann, K. and Lampinen, A. (2020). What shapes feature representations? exploring datasets, architectures, and training. *Advances in Neural Information Processing Systems*, 33:9995–10006.

[30] Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

[31] Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95.

[32] Idrissi, B. Y., Arjovsky, M., Pezeshki, M., and Lopez-Paz, D. (2021). Simple data balancing achieves competitive worst-group-accuracy. *arXiv preprint arXiv:2110.14503*.

[33] Kaushik, D., Hovy, E., and Lipton, Z. (2020). Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.

[34] Kaushik, D. and Lipton, Z. C. (2018). How much reading does reading comprehension require? a critical investigation of popular benchmarks. *arXiv preprint arXiv:1808.04926*.

[35] Kaushik, D., Setlur, A., Hovy, E., and Lipton, Z. C. (2021). Explaining the efficacy of counterfactually augmented data. *International Conference on Learning Representations (ICLR)*.

[36] Khani, F., Raghunathan, A., and Liang, P. (2019). Maximum weighted loss discrepancy. *arXiv preprint arXiv:1906.03518*.

[37] Kirichenko, P., Izmailov, P., and Wilson, A. G. (2022). Last layer re-training is sufficient for robustness to spurious correlations. *arXiv preprint arXiv:2204.02937*.

[38] Kleinberg, J., Mullainathan, S., and Raghavan, M. (2016). Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*.

[39] Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., and Willing, C. (2016). Jupyter notebooks – a publishing format for reproducible computational workflows. In Loizides, F. and Schmidt, B., editors, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press.

[40] Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., et al. (2021). Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR.

[41] Kolesnikov, A., Beyer, L., Zhai, X., Puigcerver, J., Yung, J., Gelly, S., and Houlsby, N. (2020). Big transfer (bit): General visual representation learning. In *European conference on computer vision*, pages 491–507. Springer.

[42] Kolesnikov, A. and Lampert, C. H. (2016). Improving weakly-supervised object localization by micro-annotation. *arXiv preprint arXiv:1605.05538*.

[43] Krizhevsky, A. (2014). One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*.

[44] Lee, Y., Yao, H., and Finn, C. (2022). Diversify and disambiguate: Learning from underspecified data. *arXiv preprint arXiv:2202.03418*.

[45] Li, Y., Li, Y., and Vasconcelos, N. (2018). Resound: Towards action recognition without representation bias. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 513–528.

[46] Liu, E. Z., Haghgoo, B., Chen, A. S., Raghunathan, A., Koh, P. W., Sagawa, S., Liang, P., and Finn, C. (2021). Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*, pages 6781–6792. PMLR.

[47] Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738.

[48] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. (2022). A convnet for the 2020s. *arXiv preprint arXiv:2201.03545*.

[49] Loshchilov, I. and Hutter, F. (2017). Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

[50] Lovering, C., Jha, R., Linzen, T., and Pavlick, E. (2020). Predicting inductive biases of pre-trained models. In *International Conference on Learning Representations*.

[51] Marcel, S. and Rodriguez, Y. (2010). Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1485–1488.

[52] McCoy, R. T., Pavlick, E., and Linzen, T. (2019). Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

[53] McInnes, L., Healy, J., and Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

[54] McKinney, W. (2010). Data structures for statistical computing in python. In van der Walt, S. and Millman, J., editors, *Proceedings of the 9th Python in Science Conference*, pages 51 – 56.

[55] Miller, J. P., Taori, R., Raghunathan, A., Sagawa, S., Koh, P. W., Shankar, V., Liang, P., Carmon, Y., and Schmidt, L. (2021). Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International Conference on Machine Learning*, pages 7721–7735. PMLR.

[56] Mo, S., Kang, H., Sohn, K., Li, C.-L., and Shin, J. (2021). Object-aware contrastive learning for debiased scene representation. *Advances in Neural Information Processing Systems*, 34.

[57] Moayeri, M., Pope, P., Balaji, Y., and Feizi, S. (2022). A comprehensive study of image classification model sensitivity to foregrounds, backgrounds, and visual attributes. *arXiv preprint arXiv:2201.10766*.

[58] Nam, J., Cha, H., Ahn, S., Lee, J., and Shin, J. (2020). Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684.

[59] Nam, J., Kim, J., Lee, J., and Shin, J. (2022). Spread spurious attribute: Improving worst-group accuracy with spurious attribute estimation. In *International Conference on Learning Representations*.

[60] Niven, T. and Kao, H.-Y. (2019). Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355*.

[61] Oakden-Rayner, L., Dunnmon, J., Carneiro, G., and Ré, C. (2020). Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *Proceedings of the ACM conference on health, inference, and learning*, pages 151–159.

[62] Pagliardini, M., Jaggi, M., Fleuret, F., and Karimireddy, S. P. (2022). Agree to disagree: Diversity through disagreement for better transferability. *arXiv preprint arXiv:2202.04414*.

[63] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch.

[64] Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. (2017). On fairness and calibration. *Advances in neural information processing systems*, 30.

[65] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., et al. (2017). Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*.

[66] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407.

[67] Rosenfeld, A., Zemel, R., and Tsotsos, J. K. (2018). The elephant in the room. *arXiv preprint arXiv:1808.03305*.

[68] Ruan, Y., Dubois, Y., and Maddison, C. J. (2021). Optimal representations for covariate shift. *arXiv preprint arXiv:2201.00057*.

[69] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

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

[71] Sagawa, S., Koh, P. W., Lee, T., Gao, I., Xie, S. M., Shen, K., Kumar, A., Hu, W., Yasunaga, M., Marklund, H., et al. (2021). Extending the wilds benchmark for unsupervised adaptation. *arXiv preprint arXiv:2112.05090*.

[72] Sagawa, S., Raghunathan, A., Koh, P. W., and Liang, P. (2020). An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pages 8346–8356. PMLR.

[73] Shah, H., Tamuly, K., Raghunathan, A., Jain, P., and Netrapalli, P. (2020). The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33:9573–9585.

[74] Shetty, R., Schiele, B., and Fritz, M. (2019). Not using the car to see the sidewalk–quantifying and controlling the effects of context in classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8218–8226.

[75] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[76] Singla, S. and Feizi, S. (2021). Salient imagenet: How to discover spurious features in deep learning? *arXiv preprint arXiv:2110.04301*.

[77] Sohoni, N., Sanjabi, M., Ballas, N., Grover, A., Nie, S., Firooz, H., and Ré, C. (2021). Barack: Partially supervised group robustness with guarantees. *arXiv preprint arXiv:2201.00072*.

[78] Suvra Ghosal, S., Ming, Y., and Li, Y. (2022). Are vision transformers robust to spurious correlations? *arXiv e-prints*, pages arXiv–2203.

[79] Teney, D., Abbasnejad, E., Lucey, S., and Hengel, A. v. d. (2021). Evading the simplicity bias: Training a diverse set of models discovers solutions with superior ood generalization. *arXiv preprint arXiv:2105.05612*.

[80] Veitch, V., D'Amour, A., Yadlowsky, S., and Eisenstein, J. (2021). Counterfactual invariance to spurious correlations: Why and how to pass stress tests. *arXiv preprint arXiv:2106.00545*.

[81] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

[82] Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. (2011). The caltech-ucsd birds-200-2011 dataset.

[83] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., and Summers, R. M. (2017). Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106.

[84] Wightman, R. (2019). Pytorch image models. https://github.com/rwightman/pytorch-image-models.

[85] Williams, A., Nangia, N., and Bowman, S. R. (2017). A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.

[86] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

[87] Xiao, K., Engstrom, L., Ilyas, A., and Madry, A. (2020). Noise or signal: The role of image backgrounds in object recognition. *arXiv preprint arXiv:2006.09994*.

[88] Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. (2017). Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500.

[89] Zagoruyko, S. and Komodakis, N. (2016). Wide residual networks. *arXiv preprint arXiv:1605.07146*.

[90] Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. (2021). Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR.

[91] Zech, J. R., Badgeley, M. A., Liu, M., Costa, A. B., Titano, J. J., and Oermann, E. K. (2018). Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study. *PLoS medicine*, 15(11):e1002683.

[92] Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. (2017). mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

[93] Zhang, J., Lopez-Paz, D., and Bottou, L. (2022). Rich feature construction for the optimization-generalization dilemma. *arXiv preprint arXiv:2203.15516*.

[94] Zhang, M., Sohoni, N. S., Zhang, H. R., Finn, C., and Ré, C. (2021). Correct-n-contrast: A contrastive approach for improving robustness to spurious correlations. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*.

[95] Zhong, Z., Zheng, L., Kang, G., Li, S., and Yang, Y. (2020). Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008.

[96] Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. (2017). Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Limitations are described in the appendix.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Broader impact is descussed in the appendix.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A] No theoretical results.
   (b) Did you include complete proofs of all theoretical results? [N/A] No theoretical results.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We report errorbars for the main experimental results in the appendix.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We include an estimate of the total compute used in the appendix.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes] We mention the license of all assets in the appendix.
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A] Not including new assets.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] Not including new assets
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] Not including new assets
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A] No research on human subjects or crowdsourcing.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] No research on human subjects or crowdsourcing.
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] No research on human subjects or crowdsourcing.