# Contrastive Adapters for Foundation Model Group Robustness

**Michael Zhang and Christopher Ré**
Department of Computer Science
Stanford University
{mzhang, chrismre}@cs.stanford.edu

## Abstract

While large pretrained foundation models (FMs) have shown remarkable zero-shot classification robustness to dataset-level distribution shifts, their robustness to subpopulation or group shifts is relatively underexplored. We study this problem, and find that foundation models such as CLIP may not be robust to various group shifts. Across 9 robustness benchmarks, zero-shot classification with their embeddings results in gaps of up to 80.7 percentage points (pp) between average and worst-group accuracy. Unfortunately, existing methods to improve robustness require retraining, which can be prohibitively expensive on large foundation models. We also find that efficient ways to improve model inference (*e.g.*, via adapters, lightweight networks that transform FM embeddings) do not consistently improve and can sometimes *hurt* group robustness compared to zero-shot. We therefore develop an adapter training strategy to effectively and efficiently improve FM group robustness. Our motivating observation is that while poor robustness results from groups in the same class being embedded far apart in the foundation model "embedding space," standard adapter training may not actually bring these points closer together. We thus propose contrastive adapting, which contrastively trains adapters to bring sample embeddings close to both their ground-truth class embeddings *and* same-class *sample* embeddings. Across the 9 robustness benchmarks, contrastive adapting consistently improves group robustness, raising worst-group accuracy by 8.5 to 56.0 pp over zero-shot. Our approach is also efficient, doing so without any FM finetuning and only a fixed set of FM embeddings. On popular benchmarks such as Waterbirds and CelebA, this leads to worst-group accuracy comparable to state-of-the-art methods, while only training $\leq 1\%$ of the model parameters.

## 1 Introduction

Foundation models (FMs)—large pretrained models trained on massive datasets—offer an exciting new paradigm for deep learning. Recent works have shown that without any finetuning, foundation models can generalize well to various datasets [11, 36, 59, 69] and exhibit impressive robustness to certain distribution shifts [42, 76]. Under this zero-shot paradigm, practitioners can avoid training task-specific models, and instead use FM embeddings for efficient and effective inference.

However, an underexplored question is how robust this zero-shot inference is to "group shifts," distribution shifts between subpopulations or meaningful groups in data. Prior works have established that *group robustness—i.e.* performing well on all groups—is a fundamental and real-world challenge for modern deep learning [5, 12, 40, 51, 55, 66, 71]. Yet most prior foundation model evaluations focus on overall or average performance [42, 59, 76]; few works consider FM accuracy across groups.

In this work, we thus study foundation model group robustness. We motivate this problem by first showing that foundation models can have poor zero-shot group robustness. Evaluating 11 foundation

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

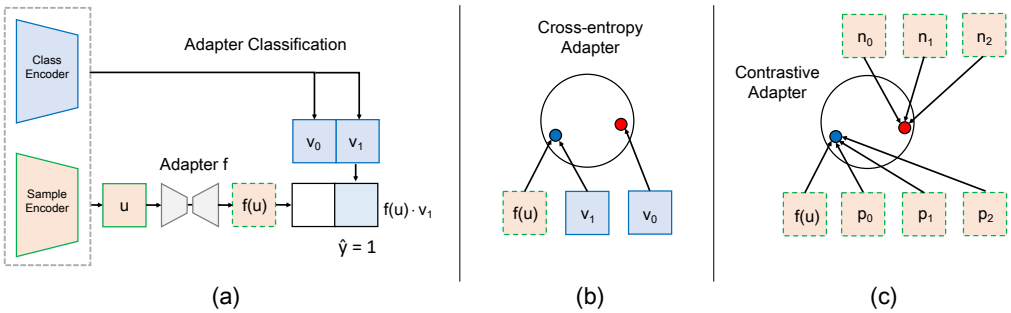


Figure 1: (a) Adapter classification with FM embeddings. Adapters learn transformations to align sample embeddings to ground-truth class embeddings. (b) Cross-entropy loss encourages alignment between class embeddings [22]. (c) Contrastive adapting adds alignment between sample embeddings.

models across 9 robustness benchmarks, we find they achieve up to an 80.7 percentage point (pp) gap between average and worst group accuracy, only classifying 6.0% of worst-group samples correctly.

We therefore aim to improve FM group robustness. This poses several challenges and open questions. First, while improving group robustness in machine learning is well-studied, existing robustness methods require retraining one (and often more than one) entire models [1, 16, 39, 47, 51, 65, 71, 72, 79]. This can be prohibitively expensive for foundation models due to their size and scale, raising the question of whether we can make these models more robust without any retraining or finetuning. Second, for zero-shot classification, many practitioners may also only access foundation model outputs or embeddings (*e.g.*, via APIs[1]). To improve robustness, ideal solutions should only require pretrained FM embeddings. However, these same embeddings lead to poor zero-shot robustness, raising the question of if they even encode the information needed to classify all groups correctly.

Motivated by these challenges and questions, we study effective *and* efficient solutions for better FM group robustness. As a baseline, we first find that while efficient methods to improve FM inference—such as training linear probes [42, 59] and adapters [22, 33] on top of FM embeddings—can improve group robustness over zero-shot (reducing the gap by up to 50.2 pp on representative benchmarks), they fail to do so consistently, and can *hurt* robustness. They reduce worst-group accuracy by up to 37.9 pp, and increase the accuracy gap by up to 74.9 pp. To reason about this inconsistency, we note that poor zero-shot robustness results when FMs embed same-class samples in different groups "far apart" in embedding space. While adapter training achieves higher robustness than linear probing, we find settings where it still fails to close this distance, *e.g.*, if training data is group-imbalanced.

To then handle these scenarios and consistently improve group robustness over zero-shot, we propose *contrastive adapting*, a simple adapter training method that places greater emphasis on bringing these initially "far apart" points together. For each task, we first use foundation models to compute embeddings for each training sample and class. We then train adapters—small bottleneck MLPs—on these embeddings. Like prior work [22], these adapters take sample embeddings as inputs, and output transformed embeddings with greater cosine similarity to their ground-truth class embeddings. However, the key difference is that contrastive adapting also applies a supervised contrastive loss over other *sample* embeddings. Specifically, we provide a way to "pull together" far apart sample embeddings in the same class, and "push apart" nearby sample embeddings in different classes.

In our experiments, we validate that contrastive adapting effectively and efficiently improves FM group robustness. First, across all 9 robustness benchmarks, we find contrastive adapting consistently improves worst-group accuracy over zero-shot (by 8.5 to 56.0 pp), using no training group labels and only training MLPs with 0.1% to 0.3% of the original FM parameters. Then, on a representative set of benchmarks with various group shifts and training data group sizes, we find contrastive adapting can substantially outperform prior adapter training strategies, and outperforms other approaches that only use fixed FM embeddings (achieving up to 12.4 pp higher worst-group accuracy than the next best method on average). Finally, beyond just improving FM robustness, we find contrastive adapting also achieves effective and efficient group robust classification in general. We achieve near state-of-the-art (SoTA) or SoTA worst-group accuracy on popular robustness benchmarks with only 1.0% of the trainable parameters (*e.g.*, improving 0.2 pp over the prior SoTA [52] on CelebA [48]).

[1] https://beta.openai.com/docs/introduction., https://studio.ai21.com/docs/, https://docs.cohere.ai/

In summary, we find that while FM zero-shot classification may not be group-robust, we can significantly improve robustness without any finetuning. This suggests the information to classify groups is frequently in their pretrained embeddings; we may just need proper methods to extract it.

## 2 Related Work

Our work builds on (i) methods to improve group robustness, and (ii) methods to improve foundation model inference without accessing or finetuning their original weights. We briefly describe these works here, and include an expanded discussion in Appendix D.

**Improving group robustness.** Many works aim to improve group robustness. If training group labels are known, prior methods often balance group sizes during training, via sample balancing [17, 28, 34, 39], importance weighting [13, 68], or robust optimization [2, 65]. We do not assume training group labels. With these assumptions, a common approach first trains a model with empirical risk minimization (ERM), before using this model's predictions to infer groups. Methods then train a second robust model with sample balancing [47, 51] or robust optimization [16, 52, 71] using inferred group labels, or representation learning to learn similar representations for groups in the same class [79]. While effective at improving group robustness, these solutions require training one (and often more than one) models. This can make applying them to foundation models prohibitively expensive.

**Improving foundation model inference efficiently.** Other prior works improve foundation model downstream performance, without having to finetune or update original model weights. *Prompt tuning* optimizes the inputs of a FM while keeping the original model weights frozen. Optimizing either text [43, 45, 83, 84] or image [3, 77] inputs can improve a frozen foundation model's downstream task accuracy. However, doing so can require multiple passes through the foundation model, which may become expensive in certain situations (*e.g.*, interacting with the model via a commercial API). Another paradigm adds small trainable parameters to the original model, either within its layers or on top of its embeddings. These include linear probes (linear classifiers) [59] and adapters (small bottleneck MLPs) [33, 57, 58, 60]. Recently, Kumar et al. [42], Wortsman et al. [76] propose methods with linear probes to improve robustness after finetuning to out-of-distribution (OOD) shifts [30, 32, 62, 74]. Gao et al. [22] train adapters on pretrained embeddings to improve average downstream accuracy. We focus on *group shifts* within a dataset. We also show standard adapter training can hurt group robustness, and propose alternatives to consistently improve group robustness.

## 3 Problem

In Section 3.1, we first describe the group robustness problem setting. In Section 3.2, we illustrate this problem with foundation models. We show that zero-shot classification with foundation models, and existing baseline approaches to improve downstream inference, can result in poor group robustness.

### 3.1 Preliminaries: group robustness and task setup

We emphasize robustness to distribution shifts between groups in this work. For setup, we follow prior works [40, 47, 65, 71] that alternatively describe the phenomenon as *hidden stratification* [71] or *subpopulation shift* [40]. For some task, we have $N$ samples $\{(x_i, y_i, g_i)\}_{i=1}^N$, with sample features or inputs $x_i \in \mathcal{X}$, class labels $y_i \in \mathcal{Y}$, and group labels $g_i \in \mathcal{G}$. Let $C = |\mathcal{Y}|$ be the number of classes. We use $g_i$ to indicate the group that each sample belongs in, but do not observe group labels during training. Distribution shifts may occur between samples in different groups but the same class.

Every sample $(x_i, y_i, g_i)$ is drawn from some unknown joint distribution $P$. Let $P_g$ be the specific distribution conditioned on $g$ for any $g \in \mathcal{G}$. For classification loss $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$ and classifier $f_\theta : \mathcal{X} \mapsto \mathcal{Y}$, we want $f_\theta$ to be accurate, *i.e.* achieving low average error:

$$\mathcal{L}_{\text{avg}}(f_\theta) := \mathbb{E}_{(x,y,g)\sim P}[\ell(f_\theta(x), y)] \tag{1}$$

and *group robust*, *i.e.*, achieving a small gap between its average error and its worst-group error:

$$\mathcal{L}_{\text{wg}}(f_\theta) := \max_{g\in\mathcal{G}} \mathbb{E}_{(x,y,g)\sim P_g}[\ell(f_\theta(x), y)] \tag{2}$$

Different from domain generalization or OOD evaluation settings [31, 32, 44, 64, 82], we observe each data group in training, validation, and test splits. However, standard training via empirical risk minimization (ERM) can still lead to poor test set group robustness because training groups may be imbalanced [65, 71, 79]. Here, foundation models are *not trained* on the training data, but we show that zero-shot classification with foundation models can still result in poor group robustness.

Figure 2: Samples of different group shifts for robust evaluation (2 classes, 2 groups per class shown).

### 3.2 Empirical findings of poor foundation model group robustness

To motivate the rest of this work, we now demonstrate the group robustness problem with foundation models. We first describe different natural group shifts for evaluation. We next detail primary baseline approaches. We finally summarize our findings after evaluating these baselines on 11 popular foundation models across 9 standard group robustness benchmarks used in prior work [40, 49, 61, 65, 67]. We present four representative scenarios based on training data assumptions and group robustness outcome. Critically, we find that zero-shot classification with foundation models may result in poor group robustness. We also find that baseline methods to improve downstream transfer do not consistently improve group robustness, and can make group robustness worse.

**Dataset group shifts.** We benchmark methods on the following sources of group shift (Figure 2):

- **Spurious confounders.** We evaluate across groups which may or may not carry spurious confounders—input features predictive for some, but not all groups in a class. For example, in Waterbirds [65, 75], a water background is a confounder for the `waterbirds` class.
- **Subclass variance.** We evaluate across groups which are different fine-grained subclasses. For example, in BREEDS Living-17 [67], the `ape` class includes images of gibbons and gorillas.
- **Data source variance.** We evaluate across groups which are the same class but sourced from different datasets. For example, we set up the CIFAR-10.02 dataset by combining CIFAR-10 [41] and CIFAR-10.2 [49]. The `airplanes` class contains samples from both datasets.

**Baseline methods.** To evaluate foundation model group robustness, we consider the following baseline methods. Following prior work [20, 36, 46, 50, 59], for all approaches we first compute $N$ sample embeddings and $C$ class embeddings using a foundation model. With foundation model embedding dimension $D$, let $u_n \in \mathbb{R}^D$ be a sample embedding and $c_n \in \mathbb{R}^D$ be a class embedding.

- **Zero-shot classification** [59]: We classify each sample via the nearest class embedding to its sample embedding $u_n$. Specifically, we compute the class-wise logits for each sample $x_n$ as

$$f_\theta(x_n; \tau) = \hat{W}^\top \hat{u}_n / \tau \tag{3}$$

  where $\hat{u}_n = u_n / \|u_n\|$ is the ($\ell$-2) normalized sample embedding of $x_n$, $\hat{W} \in \mathbb{R}^{D \times C}$ is a matrix whose columns are the normalized class embeddings $\{\hat{v}_c\}_{c=1}^C$, and $\tau$ is a temperature parameter. The highest class logit corresponds to the nearest neighbor and largest dot product. As standard, for class embeddings we convert each class name to a natural language prompt, *e.g.*, "photo of a `[class name]`", and feed the tokenized prompt to a foundation model's text encoder. As in prior work [59], we engineer class prompts by trying several templates. We defer details to Appendix A.2, such as optimal templates (Table 11) and a list of all templates tried (Table 20).
- **Linear Probe** [59, 76]: We train a linear classifier on top of training data sample embeddings. Specifically, with classifier $f_\theta(u) = W^\top u$, we update the weights $W \in \mathbb{R}^{D \times C}$ with a cross-entropy loss applied over training data sample embeddings $\{u_n\}_{n=1}^N$ and labels $\{y_n\}_{n=1}^N$.
- **Adapter** [22, 60]: We train a single 2-layer bottleneck multilayer perception (MLP) to output transformed sample embeddings, which we use instead of the original sample embeddings to classify with in the zero-shot procedure above. Specifically, with adapter hidden-layer dimension $H$, ReLU activation function $\sigma$, and adapter weights $\phi = [W_1, W_2]$—where $W_1 \in \mathbb{R}^{D \times H}$ is a linear down-projection and $W_2 \in \mathbb{R}^{H \times D}$ a linear up-projection—we compute "adapted" embeddings

$$f_\phi(u) = W_2^\top \sigma\left(W_1^\top u\right) \tag{4}$$

  We classify samples with the zero-shot class matrix $\hat{W}$, temperature $\tau$, and normalized adapted embeddings $\hat{f}_\phi(u) = f_\theta(u) / \|f_\phi(u)\|$. The final outputs are given by $f_\theta(u; \hat{W}, \tau) = \hat{W}^\top \hat{f}_\phi(u) / \tau$. Like with linear probes, we update $\phi$ with a cross-entropy loss using training data labels $\{y_n\}_{n=1}^N$ and a softmax over the dot product-computed logits as class-wise probabilities.

For evaluation, we train both linear probes and adapters with standard empirical risk minimization (ERM), which aims to minimize the empirical risk: $\hat{\mathcal{L}}(f_\theta) = \frac{1}{N} \sum_{n=1}^N \ell(f_\theta(u_n), y_n)$.

Table 1: Baseline worst-group (WG) and average (Avg) accuracies with zero-shot classification, linear probes, and adapters. Best metric **in bold**. While training linear probes and adapters can improve group robustness (reducing the worst-group versus average accuracy gap by 57.4 pp on BREEDS Living-17), it can also result in poorer robustness (in red), increasing the gap by 74.9 pp on CelebA.

| Method | Waterbirds | | | CelebA | | | BREEDS Living-17 | | | CIFAR-10.02 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap |
| Zero-shot | 36.6 | 92.2 | 55.6 | **74.0** | 81.9 | **7.9** | 6.0 | 86.7 | 80.7 | 39.1 | 69.9 | 30.8 |
| Linear Probe | 7.9 | 93.5 | 85.6 | 11.9 | 94.7 | 82.8 | 53.3 | 90.8 | 37.5 | 51.3 | 77.7 | 26.4 |
| Adapter | **60.8** | 96.0 | **35.2** | 36.1 | 94.2 | 58.1 | **70.7** | 94.0 | **23.3** | **68.8** | 86.0 | **17.2** |

Table 2: Representative outcomes for improving group robustness.

| | | Class-wise Group Size | | | Improved Group Robustness? | |
|---|---|---|---|---|---|---|
| Example Dataset | Group Shift | Largest | Smallest | Balanced? | Linear Probe | Adapter |
| Waterbirds | Confounder | 1057 | 56 | ✗ | ✗ | ✓ |
| CelebA | Confounder | 22880 | 1387 | ✗ | ✗ | ✗ |
| BREEDS Living-17 | Subclass | 1076 | 1009 | ✓ | ✓ | ✓ |
| CIFAR-10.02 | Data source | 4039 | 431 | ✗ | ✓ | ✓ |

**Discussion and representative outcomes.** In Table 1, we report worst-group and average accuracies along with their corresponding gaps on four representative group robustness datasets, using zero-shot classification, linear probes, and adapters on CLIP ResNet-50 embeddings. We select datasets to report based on training data setup and group robustness outcome, where we find that (i) the relative group size ratios, (ii) the type of group shift, and (iii) the choice of adapter or linear probe influences group robustness improvements. We note descriptive characteristics and outcomes in Table 2, and summarize three main takeaways below. Appendix A contains results for all datasets and models.

1. **Foundation model zero-shot classification may not be group robust**: Across datasets, we find that zero-shot classification with CLIP ResNet-50 embeddings can achieve 7.9 to 80.7 pp gaps between worst-group and average accuracy. Worryingly, poor group robustness is accompanied by high *average* error (from 69.9% to 92.9%), the usual metric for evaluating zero-shot classification. This further supports the importance of improving group robustness.
2. **Efficient baselines do not consistently improve robustness**: We find that while previously proposed linear probes and adapters are efficient ways to improve accuracy on downstream tasks, these benefits do not consistently carry over to improving group robustness.
   - When training data is balanced, both linear probes and adapters can substantially improve group robustness and worst-group accuracy (reducing the robustness gap by 43.2 and 54.7 pp respectively on BREEDS Living-17). However, when minority groups are rare, in some instances, approaches can hurt group robustness. On CelebA, adapters and linear probes increase the gap by 50.2 and 74.9 pp, and reduce worst-group accuracy by 37.9 and 62.1 pp.
3. **We can improve group robustness with only foundation model embeddings**: Our positive results suggest that poor zero-shot classification may not be because sample embeddings lack the information required to classify groups correctly. Rather, we may just require the right training strategies to learn how to better classify by this information.

Altogether, takeaways 1 and 2 motivate the need for methods to effectively improve robustness in the foundation model setting. Takeaway 3 suggests we can make progress on this problem.

# 4 Method

Having established the group robustness problem in Section 3, we now propose a simple contrastive adapter training strategy to improve group robustness. In Section 4.1, we setup our approach by identifying possible sources of limitation with standard adapter training. In Section 4.2, we then use these insights to propose a simple yet effective approach that counteracts these limitations.

## 4.1 Understanding prior limitations via embedding metrics

To guide a first-step strategy for improving robustness, we first outline high-level reasoning for why zero-shot and ERM-trained adapters fail to classify groups correctly. Recall that a key property of group robust classification is that all sample embeddings belonging to the same class should embed closer to their ground-truth class embedding than any other class embedding. If zero-shot classification for a specific class is accurate on average but not group robust, then in the pretrained

foundation model embedding space there exists groups that embed "close" to their ground-truth class embedding, and groups in the same class that embed "far away" (measured via cosine similarity). One way to interpret standard adapter training with FM embeddings via ERM is that it aims to bring these initially far apart sample embeddings closer to their ground-truth class embedding. Restating the standard sample cross-entropy loss with adapters makes this clear as an InfoNCE loss [14, 56]:

$$\ell(f_\theta(u), y) = -\log \frac{\exp(\hat{f}_\theta(u)^\top \hat{v}/\tau)}{\sum_{c=1}^{C} \exp(\hat{f}_\theta(u)^\top \hat{v}_c/\tau)} \tag{5}$$

with sample embedding $u$ as an anchor, class embedding $v$ of ground-truth $y$ as a single positive, and the other $C-1$ class embeddings as negatives. Via ERM of the sample cross-entropy loss, adapters thus bring zero-shot-incorrect anchors closer to their class embedding positives (minimizing Eq. 5).

However, in Section 3 we found this loss works in some scenarios but not others. Intuitively, Eq. 5 can fail to bring samples closer to their correct class embedding (*e.g.*, on CelebA). To find additional ways to bring points together, we hypothesize that poor robustness also accompanies poor similarity between *sample embeddings* from different groups but the same class. We verify this in Figure 3 by empirically measuring the average pairwise cosine similarity and group alignment loss $\mathcal{L}_{\text{align}}$ [79]—which measures the pairwise Euclidean distance—between sample embeddings in the same class but different groups. We compare these metrics with embeddings computed with trained adapters and the initial foundation model embeddings, and find that higher worst-group accuracy corresponds to higher cosine similarity and lower alignment loss between groups in the same class.

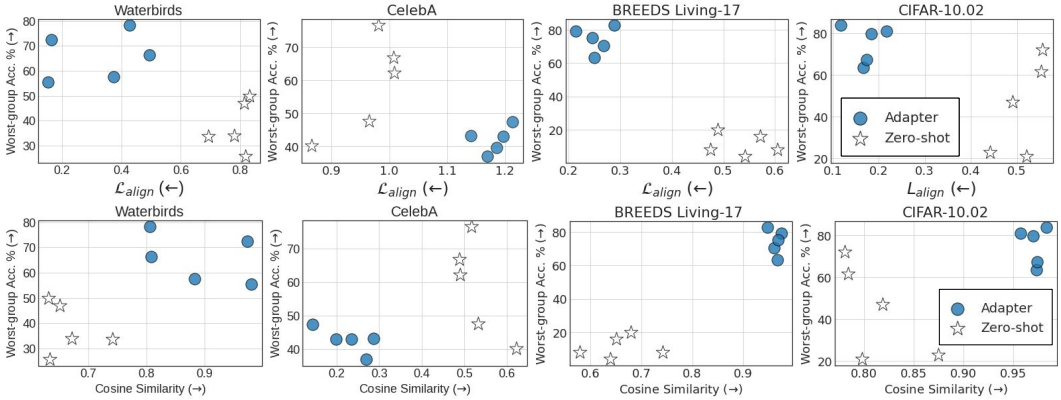


Figure 3: Across CLIP model architectures, cosine similarity and alignment loss between groups of the same class tracks worst-group error. Notably, training ERM adapters may fail to move these metrics in the desired direction, which corresponds with poorer robustness (*e.g.*, on CelebA).

### 4.2 Approach: Contrastive Adapting

To improve robustness, we therefore propose to more effectively bring far away samples together by introducing greater training signal via other *sample* embeddings. Instead of limiting ourselves to a single class embedding positive and a limited set of $C-1$ negatives, we expand our positives by including *sample* embeddings for points in the same class far away from the anchors among pretrained embeddings (*e.g.*, likely in different groups). We expand our negatives with sample embeddings from different classes. Following prior work [23, 79] that finds sampling *hard negatives* beneficial for robust contrastive learning, we also use the computed foundation model sample embeddings to sample negatives from points nearest to the anchors but in different classes. As the number of training data points $N$ is often much larger than the number of classes $C$, these choices are further supported by prior work suggesting more positives and negatives are beneficial for contrastive learning [38, 63]. In practice, contrastive adapting is simple to implement with three components:

- **Foundation model embedding and prediction**: We compute FM embeddings over labeled training data. To guide sampling, we collect zero-shot predictions over this data.
- **Contrastive sampling**: For each class, we identify an "anchor" sample embedding $u \in \mathcal{U}$ that zero-shot predicts incorrectly, and $P$ "positive" sample embeddings $\mathcal{P}(u)$ that zero-shot classifies correctly. We do this as a heuristic for finding samples "far apart" in the FM embedding space, so pushing them together improves robustness over zero-shot. We also identify $M$ hard "negative" sample embeddings $\mathcal{M}(u)$ by computing the nearest neighbors to the anchors in different classes, using cosine similarity between the sample embeddings.

- **Training objective**: We use a supervised contrastive loss [38] with the sample embeddings, *i.e.*

$$\ell_{\text{con}}^{\text{sup}}(f_\theta(u)) = \frac{-1}{P} \sum_{p \in \mathcal{P}(u)} \log \frac{\exp(\hat{f}_\theta(u)^\top \hat{f}_\theta(p)/\tau)}{\exp(\hat{f}_\theta(u)^\top \hat{f}_\theta(p)/\tau) + \sum_{m \in \mathcal{M}(u)} \exp(\hat{f}_\theta(u)^\top \hat{f}_\theta(m)/\tau)} \quad (6)$$

  We also use a standard cross-entropy loss over minibatches of sample embeddings and their class embeddings. This aims to keep adapted embeddings close to their ground-truth class embeddings. To avoid undoing Eq. 6 and push "far away" points closer to their ground-truth class embeddings, we upsample the number of zero-shot incorrect samples to equal the number of zero-shot correct samples in each minibatch. We thus use contrastive supervision from class and sample embeddings.

**Robust generalization with adapters.** The contrastive loss in Equation 6 is also supported by recent results suggesting that minimizing the class-wise alignment loss $\mathcal{L}_{\text{align}}$ helps bound the worst-group versus average error gap for that class (*cf*. Thm 3.1, Zhang et al. [79]). The bound however scales with the Lipschitz constant of the neural network, and upper bounds for estimating this constant can grow with the size of the network [19, 73]. However, as our adapters are small 2-layer MLPs, estimates of this constant suggest we can obtain better generalization with fewer training samples [25, 37, 53, 78]. In Section 5.3, we later show this corresponds to better data efficiency.

# 5 Experiments

We now validate that contrastive adapting enables effective and efficient group robustness. First, in Section 5.1, we evaluate the effectiveness of contrastive adapting against efficient methods to improve FM inference. We study whether the approach consistently improves worst-group accuracy and group robustness over zero-shot classification, how contrastive adapting compares against other efficient methods that only require pretrained model embeddings, and whether contrastive adapting scales to a variety of pretrained model architectures. Next, in Section 5.2, we shed further light on contrastive adapting's performance by studying the importance of its individual components, ablating the contrastive objective and sampling strategy. Finally, in Section 5.3, we study the efficiency of contrastive adapting against effective group robustness approaches. We find that the prior robustness gains are not only relative to other efficient FM training methods; contrastive adapting also enables state-of-the-art robustness on popular benchmarks, but with greater parameter and data efficiency.

## 5.1 Robustness comparison for efficient foundation model methods

To first judge the effectiveness of contrastive adapting, we evaluate the method across the same set of initial robustness benchmarks and foundation model architectures discussed in Section 3. As in prior group robustness evaluation, we do not assume training groups labels, but do assume group labels in validation data for hyperparameter tuning and model selection [40]. We include experimental details for all models and hyperparameters in Appendix C.

As baselines, we compare against zero-shot classification [59], ERM linear probing [42, 59], and ERM adapter training [22]. We also compare against recent methods designed to improve downstream transfer in related settings, while similarly only requiring pretrained model embeddings:

- **Weight space ensembling (WiSE-FT)** [76], which first trains a linear classifier with standard ERM, and then ensembles the classifier outputs with the initial zero-shot predictions. While proposed for both training linear classifiers and finetuning the original weights of a foundation model, we focus on the linear classifier version for fair comparison in our setting.
- **Deep feature reweighting (DFR)** [39], which first trains a linear probe on embeddings computed from a pretrained model over group-balanced data. As we do not assume training group labels, we first infer groups using zero-shot classification with foundation model embeddings. As in prior work [47, 79], we treat the incorrect and correctly classified samples as proxies for different groups.

Finally, if we have validation group labels, we plausibly know what groups are in the test data. We thus also compare against **group-informed prompting** (Group Prompt ZS), which performs zero-shot classification using prompts with group information (*e.g.*, "a waterbird on a land background").

**Consistent robustness improvements over zero-shot.** In Figure 4 we report contrastive adapting's relative gains in worst-group accuracy over zero-shot classification on all 9 robustness benchmarks. Unlike prior adapter training approaches, contrastive adapting consistently improves group robustness over zero-shot classification, achieving 8.5 to 56.0 pp higher worst-group accuracy.

Table 3: Evaluation of methods for improving group robustness of CLIP models. Across representative benchmarks and CLIP models, contrastive adapters consistently improve worst-group accuracy over zero-shot classification (by 10.2 to 76.0 pp). **1st** / 2nd best worst-group (WG) and robustness gaps **bolded** / underlined.

| | | Waterbirds | | | CelebA | | | BREEDS Living-17 | | | CIFAR-10.02 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Method / Acc. (%) | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap |
| CLIP ResNet-50 | Zero-shot (ZS) | 36.6 | 92.2 | 55.6 | 74.0 | 81.9 | 7.9 | 6.0 | 86.7 | 80.7 | 39.1 | 69.9 | 30.8 |
| | Group Prompt ZS | 55.9 | 87.8 | 31.9 | 70.8 | 82.6 | 11.8 | 30.0 | 90.6 | 60.6 | N/A | N/A | N/A |
| | ERM Linear Probe | 7.9 | 93.5 | 85.6 | 11.9 | 94.7 | 82.8 | 53.3 | 90.8 | 37.5 | 51.3 | 77.7 | 26.4 |
| | ERM Adapter | 60.8 | 96.0 | 35.2 | 36.1 | 94.2 | 58.1 | **70.7** | 94.0 | **23.3** | **68.8** | 86.0 | **17.2** |
| | WiSE-FT | 49.8 | 91.0 | 41.2 | 85.6 | 88.6 | 3.0 | 53.3 | 90.8 | 37.5 | 58.2 | 79.1 | 20.9 |
| | DFR (Subsample) | 63.9 | 91.8 | 27.9 | 76.9 | 92.5 | 15.6 | 46.7 | 89.4 | 42.7 | 45.0 | 75.0 | 30.0 |
| | DFR (Upsample) | 51.3 | 92.4 | 41.1 | 89.6 | 91.8 | 2.2 | 44.0 | 86.4 | 42.4 | 38.5 | 77.9 | 39.4 |
| | **Contrastive Adapter** | **83.7** | 89.4 | **5.7** | **90.0** | 90.7 | **0.7** | 62.0 | 90.9 | 28.9 | 60.7 | 80.9 | 20.2 |
| CLIP ViT-L/14 | Zero-shot (ZS) | 25.7 | 87.3 | 61.6 | 62.1 | 71.9 | 9.8 | 4.0 | 86.6 | 82.6 | 72.0 | 93.2 | 21.2 |
| | Group Prompt ZS | 27.4 | 85.5 | 58.1 | 72.4 | 81.8 | 9.4 | 48.0 | 96.6 | 48.6 | N/A | N/A | N/A |
| | ERM Linear Probe | 65.9 | 97.6 | 31.7 | 28.3 | 94.7 | 66.4 | **84.0** | 98.6 | 14.6 | **87.5** | 96.1 | **8.6** |
| | ERM Adapter | 78.4 | 97.8 | 19.4 | 36.7 | 94.2 | 57.5 | 82.8 | 98.2 | 15.5 | 87.0 | 96.9 | 9.9 |
| | WiSE-FT | 65.9 | 97.6 | 31.7 | 80.0 | 87.4 | 7.4 | **84.0** | 98.6 | 14.6 | **87.5** | **97.0** | 9.5 |
| | DFR (Subsample) | 51.9 | 95.7 | 43.8 | 76.3 | 92.1 | 15.8 | **84.0** | 98.5 | **14.5** | 85.5 | 96.6 | 11.1 |
| | DFR (Upsample) | 65.9 | 96.1 | 30.2 | 83.7 | 91.2 | 7.5 | 78.7 | 97.3 | 18.6 | 72.5 | 93.8 | 21.3 |
| | **Contrastive Adapter** | **86.9** | 96.2 | **9.3** | **84.6** | 90.4 | **5.8** | 80.0 | 97.5 | 17.5 | 82.2 | 96.1 | 13.9 |

Table 4: On the Waterbirds dataset, contrastive adapters consistently improve group robustness across various vision-language large pretrained models (CLIP [59], CLOOB [20]) and backbones (ResNets and ViTs).

| | CLOOB RN-50 | | | CLOOB RN-50x4 | | | CLIP RN-101 | | | CLIP ViT-B/32 | | | CLIP ViT-B/16 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap |
| Zero-shot | 41.6 | 60.4 | 18.8 | 24.1 | 51.1 | 27 | 33.6 | **90.0** | 56.4 | 47.0 | **88.8** | 41.8 | 34.0 | 88.1 | 54.1 |
| Contrastive Adapter | **83.0** | **86.8** | **3.8** | **85.8** | **88.5** | **2.7** | **82.0** | 86.0 | **4.0** | **80.7** | 84.2 | **3.5** | **83.1** | **90.9** | **7.8** |

**Representative dataset evaluation.** In Table 3 we compare contrastive adapting to other lightweight methods for improving robustness. We evaluate with group-imbalanced and balanced training data across spurious confounder, subclass, and data source group shifts, using CLIP ResNet-50 (RN-50) and CLIP ViT-L/14 models. On average, contrastive adapters raise worst-group accuracy by 12.4 and 4.1 pp over the next best methods on CLIP RN-50 and ViT-L/14 models.

**Transfer across architectures.** We also study how the prior contrastive adapting improvements transfer to other pretrained models. Table 4 shows contrastive adapters substantially improve group robustness for models such as CLOOB [20]. The method also scales across model sizes, raising worst-group accuracy by 33.7 to 61.7 pp via training adapters with only 0.52% to 1.03% of the model parameters [20, 59].

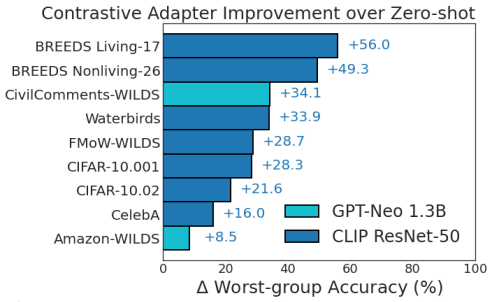


Figure 4: Across 9 group robustness benchmarks, contrastive adapting consistently improves worst-group acc. over pretrained zero-shot classification.

### 5.2 Ablation on sampling strategy and contrastive training objective

To next better understand how contrastive adapting's individual components affect group robustness, we ablate the proposed contrastive objective (Eq. 6) and "hard" sampling strategy, and report worst-group and average accuracies on CLIP RN-50 adapters (Table 5). On three datasets, we find the contrastive loss alone improves robustness more than hard sampling alone. However, on Waterbirds and CelebA—where ERM adapters perform poorly—having both components substantially improves robustness (+5.5 to 8.5 pp). Meanwhile, on BREEDS Living-17 and CIFAR-10.02—where ERM adapters perform best across all methods—removing hard sampling improves contrastive adapting performance. On these datasets, the random sampling in ERM may be *beneficial* (discussed further in App. E.6). Contrastive adapting may thus also benefit from random sampling in these settings.

### 5.3 Measuring efficiency among effective group robustness solutions

While in Section 5.1, we found contrastive adapters could significantly improve group robustness for foundation models, we now expand on contrastive adapting's efficiency. We find that for group robust classification in general, contrastive adapting can achieve state-of-the-art performance despite only training $\leq$1% of the usual model parameters. The lightweight nature of contrastive adapting also leads to better data efficiency than existing state-of-the-art approaches.

Table 5: Contrastive adapter ablation over contrastive loss and sampling. When ERM obtains poor worst-group (WG) accuracy, both contrastive loss and "hard" sampling lead to best robustness. When ERM improves WG acc. over zero-shot (c.f. Table 3, *i.e.* random sampling helps), no hard sampling also helps contrastive adapters.

| Adapter Ablation | Contrastive Loss | Hard Sampling | Waterbirds | | CelebA | | BREEDS Living-17 | | CIFAR-10.02 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | WG | Avg. | WG | Avg. | WG | Avg. | WG | Avg. |
| ERM | ✗ | ✗ | 60.8 ± 0.9 | 96.0 ± 0.1 | 36.1 ± 1.4 | 94.2 ± 0.2 | **70.7 ± 0.9** | 94.0 ± 0.1 | **68.8 ± 0.5** | 86.0 ± 0.5 |
| Hard sample only | ✗ | ✓ | 56.3 ± 1.5 | 81.4 ± 0.5 | 84.5 ± 3.2 | 92.6 ± 0.4 | 58.7 ± 4.9 | 89.6 ± 0.8 | 58.5 ± 2.0 | 80.4 ± 0.7 |
| Contrastive only | ✓ | ✗ | 75.2 ± 1.0 | 94.0 ± 0.1 | 51.4 ± 5.9 | 93.2 ± 2.6 | 67.4 ± 0.9 | 91.8 ± 0.2 | 66.9 ± 1.2 | 82.9 ± 0.3 |
| Default proposed | ✓ | ✓ | **83.7 ± 0.7** | 89.4 ± 0.9 | **90.0 ± 0.4** | 90.7 ± 0.4 | 62.0 ± 1.6 | 90.9 ± 0.3 | 60.7 ± 1.7 | 80.9 ± 0.2 |

Table 6: On popular Waterbirds and CelebA benchmarks, contrastive adapters achieve near state-of-the-art worst-group accuracy (WG Acc.) with $\leq$1% of the trainable parameters. $\Delta$Acc. is percentage point gap with prior SoTA. **1st** / 2nd best metrics **bolded** / underlined. We report numbers from original works.

| Model | # Trained Params | % Params | Method | Waterbirds | | CelebA | |
|---|---|---|---|---|---|---|---|
| | | | | WG Acc. (%) | $\Delta$Acc. | WG Acc. (%) | $\Delta$Acc. |
| ResNet-50 | 25557032 | 100 | EIIL [16] | 78.7 | -10.3 | 83.3 | -6.5 |
| | | | CIM [72] | 83.6 | -5.4 | 83.6 | -6.2 |
| | | | JTT [47] | 86.7 | -2.3 | 81.1 | -8.7 |
| | | | RWY [34] | 86.1 | -2.9 | 82.9 | -6.9 |
| | | | CNC [79] | 88.5 | -0.5 | 88.8 | -1.0 |
| | | | **SSA** [52] | **89.0** | 0.0 | 89.8 | 0.0 |
| Adapter + CLIP RN-50 | 263424 | 1.03 | Ours | 83.7 | -5.3 | **90.0** | 0.2 |
| Adapter + CLIP ViT-L/14 | 197632 | 0.77 | Ours | 86.9 | -2.1 | 84.6 | -5.2 |

**Robustness comparison to state-of-the-art methods.** In Table 6, we evaluate how contrastive adapting with CLIP RN-50 and ViT-L/14 embeddings compares to current state-of-the-art robustness techniques. We use the popular Waterbirds and CelebA datasets. Existing group robustness methods train ImageNet-pretrained ResNet-50s. On both datasets, contrastive adapting achieves comparable worst-group accuracy to these methods, despite only training $\leq$1% of their parameters. Notably, contrastive adapting outperforms some methods by up to 5.0 and 10.1 pp for Waterbirds and CelebA, and only falls short of the state-of-the-art Spread Spurious Attribute (SSA) method by 2.1 pp on Waterbirds. These results suggest contrastive adapters not only effectively improve group robustness for pretrained models, but also enable competitive robust classification in general at a fraction of prior approaches' trainable parameter counts.

**Data efficiency evaluation.** Beyond model parameter count, we also study if the lightweight nature of contrastive adapting transfers to better data efficiency. We compare contrastive adapting to the best performing SSA on subsampled versions of Waterbirds. To evaluate how well methods maintain group robustness with less training data available, we keep group ratios preserved (*i.e.*, 1% of all training samples belongs to the smallest Waterbirds group [65]).

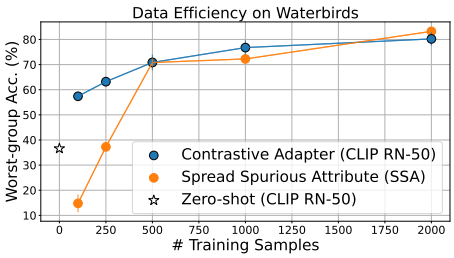


Figure 5: With less data, contrastive adapters maintain higher group robustness than zero-shot and significantly outperform standard models trained with the state-of-the-art SSA method.

Figure 5 shows that contrastive adapting substantially outperforms SSA in lower data regimes. With only 100 and 250 training samples, contrastive adapting outperforms SSA-trained ResNet-50s by 42.6 and 25.9 pp. With 100 training samples, contrastive adapting still achieves 20.8 pp higher worst-group accuracy than zero-shot classification with only 1 training sample in the smallest Waterbirds group. In contrast, SSA's accuracy drops significantly, resulting in 17.6 pp lower worst-group accuracy than zero-shot classification. To connect this fewer data result to our generalization discussion in Section 4.2, with prior methods [19] we estimate the trained adapter Lipschitz constant. We estimate the constant to be 29.3, much lower than that reported for larger networks (*e.g.*, ResNet-50s) [19, 73].

## 6 Conclusion

We study the group robustness of popular foundation models. We find their zero-shot classification may not be robust to various group shifts, establish that baseline linear probe and adapter strategies do not reliably improve robustness, and propose a simple adapter strategy to significantly and consistently improve FM robustness without finetuning. This suggests FM embeddings do contain group-relevant information, and we show that we can use FM embeddings to efficiently achieve state-of-the-art robust classification. We recognize the limitations of computational solutions to subgroup performance disparities, and the need to understand FMs in broader socio-technical systems [9].

## 7 Acknowledgements and Funding

We thank Simran Arora, Megan Leszczynski, Kush Bhatia, Maya Varma, Gautam Machiraju, and Laurel Orr for helpful discussions and feedback.

We gratefully acknowledge the funding support of NIH under No. U54EB020405 (Mobilize), NSF under Nos. CCF1763315 (Beyond Sparsity), CCF1563078 (Volume to Velocity), and 1937301 (RTML); ARL under No. W911NF-21-2-0251 (Interactive Human-AI Teaming); ONR under No. N000141712266 (Unifying Weak Supervision); ONR N00014-20-1-2480: Understanding and Applying Non-Euclidean Geometry in Machine Learning; N000142012275 (NEPTUNE); NXP, Xilinx, LETI-CEA, Intel, IBM, Microsoft, NEC, Toshiba, TSMC, ARM, Hitachi, BASF, Accenture, Ericsson, Qualcomm, Analog Devices, Google Cloud, Salesforce, Total, the HAI-GCP Cloud Credits for Research program, the Stanford Data Science Initiative (SDSI), and members of the Stanford DAWN project: Facebook, Google, and VMWare. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views, policies, or endorsements, either expressed or implied, of NIH, ONR, or the U.S. Government.

We have no other additional revenues to disclose related to this work.

## References

[1] Faruk Ahmed, Yoshua Bengio, Harm van Seijen, and Aaron C. Courville. Systematic generalisation with group invariant predictions. In *ICLR*, 2021.

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

[3] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Visual prompting: Modifying pixel space to adapt pre-trained models. *arXiv preprint arXiv:2203.17274*, 2022.

[4] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper/2019/file/97af07a14cacba681feacf3012730892-Paper.pdf`.

[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.

[6] Hugo Berg, Siobhan Mackenzie Hall, Yash Bhalgat, Wonsuk Yang, Hannah Rose Kirk, Aleksandar Shtedritski, and Max Bain. A prompt array keeps the bias away: Debiasing vision-language models with adversarial learning. *arXiv preprint arXiv:2203.11933*, 2022.

[7] Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March 2021. URL `https://doi.org/10.5281/zenodo.5297715`. If you use this software, please cite it using these metadata.

[8] Su Lin Blodgett, Lisa Green, and Brendan O'Connor. Demographic dialectal variation in social media: A case study of african-american english. *arXiv preprint arXiv:1608.08868*, 2016.

[9] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[10] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. *CoRR*, abs/1903.04561, 2019. URL `http://arxiv.org/abs/1903.04561`.

[11] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[12] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018.

[13] 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.

[14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[15] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6172–6180, 2018.

[16] Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *International Conference on Machine Learning*, pages 2189–2200. PMLR, 2021.

[17] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019.

[18] Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11162–11173, 2021.

[19] Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George Pappas. Efficient and accurate estimation of lipschitz constants for deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[20] Andreas Fürst, Elisabeth Rumetshofer, Viet Tran, Hubert Ramsauer, Fei Tang, Johannes Lehner, David Kreil, Michael Kopp, Günter Klambauer, Angela Bitto-Nemling, et al. Cloob: Modern hopfield networks with infoloob outperform clip. *arXiv preprint arXiv:2110.11316*, 2021.

[21] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

[22] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021.

[23] Songwei Ge, Shlok Mishra, Chun-Liang Li, Haohan Wang, and David Jacobs. Robust contrastive learning using negative samples with diminished semantics. *Advances in Neural Information Processing Systems*, 34, 2021.

[24] Giorgio Giannone, Serhii Havrylov, Jordan Massiah, Emine Yilmaz, and Yunlong Jiao. Just mix once: Mixing samples with implicit group distribution. In *NeurIPS 2021 Workshop on Distribution Shifts*, 2021. URL `https://www.amazon.science/publications/just-mix-once-mixing-samples-with-implicit-group-distribution`.

[25] Henry Gouk, Eibe Frank, Bernhard Pfahringer, and Michael J Cree. Regularisation of neural networks by enforcing lipschitz continuity. *Machine Learning*, 110(2):393–416, 2021.

[26] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020.

[27] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pages 1929–1938. PMLR, 2018.

[28] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

[29] 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.

[30] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, D. Song, J. Steinhardt, and J. Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8320–8329, Los Alamitos, CA, USA, oct 2021. IEEE Computer Society. doi: 10.1109/ICCV48922.2021.00823.

[31] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=HJz6tiCqYm`.

[32] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15262–15271, June 2021.

[33] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/houlsby19a.html`.

[34] Badr Youbi Idrissi, Martin Arjovsky, Mohammad Pezeshki, and David Lopez-Paz. Simple data balancing achieves competitive worst-group-accuracy. *arXiv preprint arXiv:2110.14503*, 2021.

[35] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[36] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021.

[37] Matt Jordan and Alexandros G Dimakis. Exactly computing the local lipschitz constant of relu networks. *Advances in Neural Information Processing Systems*, 33:7344–7353, 2020.

[38] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673, 2020.

[39] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. *arXiv preprint arXiv:2204.02937*, 2022.

[40] 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.

[41] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009.

[42] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

[43] Yoav Levine, Itay Dalmedigos, Ori Ram, Yoel Zeldes, Daniel Jannai, Dor Muhlgay, Yoni Osin, Opher Lieber, Barak Lenz, Shai Shalev-Shwartz, et al. Standing on the shoulders of giant frozen language models. *arXiv preprint arXiv:2204.10019*, 2022.

[44] D. Li, Y. Yang, Y. Song, and T. M. Hospedales. Deeper, broader and artier domain generalization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5543–5551, 2017.

[45] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

[46] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*, 2021.

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

[48] 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.

[49] Shangyun Lu, Bradley Nott, Aaron Olson, Alberto Todeschini, Hossein Vahabi, Yair Carmon, and Ludwig Schmidt. Harder or different? a closer look at distribution shift in dataset reproduction. In *ICML Workshop on Uncertainty and Robustness in Deep Learning*, 2020.

[50] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. *arXiv preprint arXiv:2112.12750*, 2021.

[51] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. In *Advances in Neural Information Processing Systems*, volume 33, pages 20673–20684, 2020.

[52] Junhyun Nam, Jaehyung Kim, Jaeho Lee, and Jinwoo Shin. Spread spurious attribute: Improving worst-group accuracy with spurious attribute estimation. *arXiv preprint arXiv:2204.02070*, 2022.

[53] Behnam Neyshabur, Srinadh Bhojanapalli, and Nathan Srebro. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017.

[54] Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 188–197, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1018. URL `https://aclanthology.org/D19-1018`.

[55] Luke Oakden-Rayner, Jared Dunnmon, Gustavo Carneiro, and Christopher Ré. 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, 2020.

[56] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[57] Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. AdapterHub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.7. URL `https://aclanthology.org/2020.emnlp-demos.7`.

[58] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.39. URL `https://aclanthology.org/2021.eacl-main.39`.

[59] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[60] S-A Rebuffi, H. Bilen, and A. Vedaldi. Learning multiple visual domains with residual adapters. In *Advances in Neural Information Processing Systems*, 2017.

[61] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do cifar-10 classifiers generalize to cifar-10? *arXiv preprint arXiv:1806.00451*, 2018.

[62] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5389–5400. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/recht19a.html`.

[63] J. Robinson, Ching-Yao Chuang, S. Sra, and S. Jegelka. Contrastive learning with hard negative samples. In *International Conference on Learning Representations*, 2021.

[64] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision – ECCV 2010*, pages 213–226, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-15561-1.

[65] 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. In *International Conference on Learning Representations*, 2019.

[66] Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pages 8346–8356. PMLR, 2020.

[67] Shibani Santurkar, Dimitris Tsipras, and Aleksander Madry. Breeds: Benchmarks for subpopulation shift. *arXiv preprint arXiv:2008.04859*, 2020.

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

[69] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. *arXiv preprint arXiv:2112.04482*, 2021.

[70] Sahil Singla, Mazda Moayeri, and Soheil Feizi. Core risk minimization using salient imagenet. *arXiv preprint arXiv:2203.15566*, 2022.

[71] Nimit Sohoni, Jared Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. In *Advances in Neural Information Processing Systems*, volume 33, pages 19339–19352, 2020.

[72] Saeid Asgari Taghanaki, Kristy Choi, Amir Khasahmadi, and Anirudh Goyal. Robust representation learning via perceptual similarity metrics. *arXiv preprint arXiv:2106.06620*, 2021.

[73] Aladin Virmaux and Kevin Scaman. Lipschitz regularity of deep neural networks: analysis and efficient estimation. *Advances in Neural Information Processing Systems*, 31, 2018.

[74] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.

[75] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.

[76] Mitchell Wortsman, Gabriel Ilharco, Mike Li, Jong Wook Kim, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. *arXiv preprint arXiv:2109.01903*, 2021.

[77] Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Cpt: Colorful prompt tuning for pre-trained vision-language models. *arXiv preprint arXiv:2109.11797*, 2021.

[78] Yuichi Yoshida and Takeru Miyato. Spectral norm regularization for improving the generalizability of deep learning. *arXiv preprint arXiv:1705.10941*, 2017.

[79] Michael Zhang, Nimit S Sohoni, Hongyang R Zhang, Chelsea Finn, and Christopher Ré. Correct-n-contrast: A contrastive approach for improving robustness to spurious correlations. *arXiv preprint arXiv:2203.01517*, 2022.

[80] Renrui Zhang, Rongyao Fang, Peng Gao, Wei Zhang, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021.

[81] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. *arXiv preprint arXiv:2010.00747*, 2020.

[82] Bingchen Zhao, Shaozuo Yu, Wufei Ma, Mingxin Yu, Shenxiao Mei, Angtian Wang, Ju He, Alan L. Yuille, and Adam Kortylewski. Robin : A benchmark for robustness to individual nuisancesin real-world out-of-distribution shifts. *CoRR*, abs/2111.14341, 2021. URL `https://arxiv.org/abs/2111.14341`.

[83] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *arXiv preprint arXiv:2109.01134*, 2021.

[84] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. *arXiv preprint arXiv:2203.05557*, 2022.

## Checklist

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] See Appendix.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See 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] We do not include theoretical results
   (b) Did you include complete proofs of all theoretical results? [N/A] Not Applicable
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] See attached zip in supplementary.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] For space we defer these to 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] For space we defer to 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] See Appendix.
   (b) Did you mention the license of the assets? [Yes] See Appendix.
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A] We only use existing assets, which we discuss in the appendix.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] We use existing popular robustness benchmarks for all our datasets. While some involve people (CelebA, CivilComments-WILDS, Amazon-WILDS), we were not able to locate information on how this data was collected regarding subject consent. However, we do discuss the licenses and agreements set by the original curators of these popular benchmarks.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See Appendix.
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] We do not crowdsource or conduct research with human subjects.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] We do not crowdsource or conduct research with human subjects.
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] We do not crowdsource or conduct research with human subjects.

# Appendix


**A Expanded zero-shot evaluation for group robustness** **18**

A.1 Additional details on robustness datasets and foundation models . . . . . 18

A.2 Group robustness results . . . . . 19

**B Contrastive adapter implementation details** **20**

B.1 Adapter architecture . . . . . 20

B.2 Adapter training sampling . . . . . 20

**C Additional experimental details** **21**

C.1 Model selection and hyperparameters . . . . . 21

C.2 Data splits . . . . . 22

C.3 Additional dataset assets details and discussion . . . . . 22

C.4 Compute and resources . . . . . 23

C.5 Class prompt templates . . . . . 23

**D Additional related work discussion** **23**

**E Additional experimental results** **25**

E.1 Extended main results . . . . . 25

E.2 Contrastive adapter ablations . . . . . 25

E.2.1 Ablation on loss components in training objective . . . . . 25

E.2.2 Effect of contrastive batch size . . . . . 25

E.3 Comparison to TIP-adapter and training sample nearest-neighbors lookup . . . . . 26

E.4 Evaluation with respect to weight-space ensembling trade-off . . . . . 28

E.5 Comparison to recent robustness methods adapted for the pretrained embedding setting 28

E.5.1 Method overview for adapters . . . . . 29

E.5.2 Training details and hyperparameters . . . . . 29

E.5.3 Results . . . . . 29

E.6 Additional study on how dataset properties impact adapter robustness . . . . . 30

E.6.1 Lack of group shift in CIFAR-10.02 . . . . . 30

E.6.2 Balanced groups in BREEDS Living-17 . . . . . 31

**F Limitations and societal impact** **32**

# A Expanded zero-shot evaluation for group robustness

In this section, we expand on the zero-shot evaluation of various foundation models on group robustness benchmarks discussed in Section 3. We first describe the datasets and models used in Appendix A.1. We then include results in Appendix A.2. We find consistent trends of poor group robustness with zero-shot classification, marked by poor worst-group accuracy and large gaps between average and worst-group accuracy.

## A.1 Additional details on robustness datasets and foundation models

**Datasets.** To benchmark zero-shot group robustness, we use a diverse set of datasets with group shifts from prior robustness literature. We describe them below and include details on size of groups and type of group shift in Table 7:

- **Waterbirds** [65, 75]. We classify images by bird type. Each class $\in \{$`waterbird`, `landbird`$\}$ carries two groups: birds on water backgrounds, and birds on land backgrounds.
- **CelebA** [48, 65]. We classify images by celebrity hair color. Each class $\in \{$`not blond`, `blond`$\}$ carries two groups: celebrities labeled as male, and celebrities labeled as female.
- **BREEDS (`Living-17`, `Nonliving-26`)** [67]. For the `Living-17` and `Nonliving-26` datasets in the BREEDS benchmark sourced from ImageNet [67], we classify images by one of several categories. Each class is a coarse category consisting of multiple fine-grained groups. Groups in the same class may be visually distinct (*e.g.*, the `ape` class includes images of gibbons and gorillas). While the original benchmark evaluates how classifiers trained on seen `source` groups generalize to unseen `target` groups, we adapt the datasets for our group robustness setting by adding 5% of the images in each `target` group to the `source` groups, and evaluating worst-group accuracy over all `source` and `target` groups.
- **CIFAR-10.001**, **CIFAR-10.02** [41, 49, 61]. We classify images by one of 10 categories. We combine CIFAR-10 [41] and either CIFAR-10.1 [61] or CIFAR-10.2 [49], which are collected from different sources. The new datasets' classes carry two groups determined by the source dataset. For CIFAR-10.001, 2% of the combined train and validation data is sourced from CIFAR-10.1 (the rest from CIFAR-10). For CIFAR-10.02, 10% of the combined train and validation data is sourced from CIFAR-10.2 (the rest from CIFAR-10). For both, we then split the combined data into 80% train and 20% validation sets. We merely combine the official test splits.
- **FMoW-WILDS** [15, 40]. We classify satellite images into one of 62 building or land-use categories (*e.g.*, `airport`, `zoo`). Each images belongs to one of five groups based on continental region. To test group robustness, we compare the accuracies over all samples in each group as in the WILDS benchmark [40]. We also evaluate only over test images from the same time period as training images (the "IID" split in the original WILDS benchmark [40]).
- **CivilComments-WILDS** [10, 40]. We classify if a text comment is toxic or not. Samples are organized into 8 groups based on mention of a demographic identity (*e.g.*, "female", "LGBTQ").
- **Amazon-WILDS** [40, 54]. We classify if an online text review is positive or negative. Reviews are organized into different groups based on the product category (*e.g.*, `books`, `electronics`). We adapt this dataset from the official Amazon-WILDS split by using the `category_subpopulation` split. We also map the original class labels, which are star-ratings from 1 to 5, to positive or negative reviews by discarding samples with a 3-star rating, and re-labeling 1- and 2-star ratings as `negative` and 4- and 5-start ratings as `positive`.

**Foundation models.** For image datasets, we evaluate pretrained CLIP [59] and CLOOB [20] vision-language models using publicly available weights[2][3]. We evaluate 7 available CLIP models: 3 ResNet image encoder backbones (`RN-50`, `RN-101`, `RN-50x4`), and 5 Vision Transformer image encoder backbones: (`ViT-B/32`, `ViT-B/16`, `ViT-L/14`, `ViT-L/14@336px`) and 2 CLOOB models (all available: `RN-50`, `RN-50x4`). For text datasets, we evaluate 2 pretrained GPT-Neo [7] text models trained on the Pile [21] (`GPT-Neo-125M`, `GPT-Neo-1.3B`) available on HuggingFace[4][5].

[2]CLIP: https://github.com/openai/CLIP/blob/main/clip/clip.py

[3]CLOOB: https://ml.jku.at/research/CLOOB/downloads/checkpoints/

[4]GPT-Neo 125M: https://huggingface.co/EleutherAI/gpt-neo-125M

[5]GPT-Neo 1.3B: https://huggingface.co/EleutherAI/gpt-neo-1.3B

## A.2 Group robustness results

In Figure 6, we chart worst-group and average accuracies achieved by various zero-shot foundation models across the group robustness datasets. Larger gaps between accuracies, *i.e.* high average accuracy yet low worst-group accuracy, indicate poor group robustness. In aggregate, on all datasets except FMoW-WILDS and Amazon-WILDS, we observe a shared pattern of noticeable gaps between average and worst-group accuracy, suggesting that zero-shot classification with popular foundation models may not be group robust. We perform zero-shot classification as described in Section 3. As recommended by Radford et al. [59], for each dataset we consider several prompt templates. We engineer prompts by using the single best template based on validation worst-group accuracy. In Table 11, Appendix C.5 we include a list of optimal prompts used. Table 20 includes a list of all prompt templates tried for each dataset.

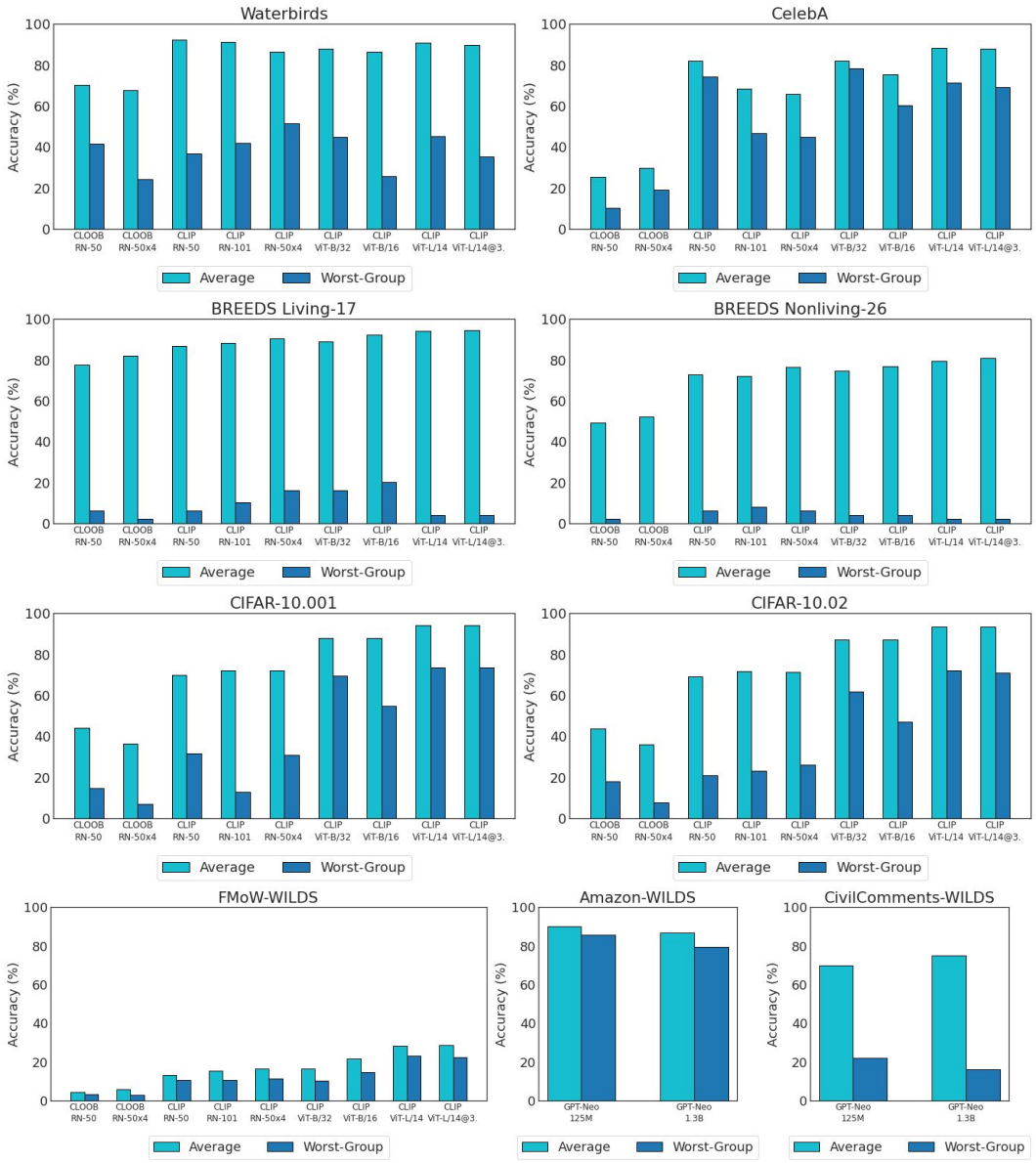


Figure 6: Foundation model zero-shot classification accuracies. We find poor zero-shot group robustness across datasets and models via large gaps between average and worst-group accuracies.

Table 7: Group robustness datasets, source of group shift, and group sizes.

| | | (Class-wise) Group Size | | |
|---|---|---|---|---|
| Dataset | Group Shift | Largest | Smallest | Class-Wise? |
| Waterbirds | Confounder | 1057 | 56 | Yes |
| CelebA | Confounder | 22880 | 1387 | Yes |
| BREEDS Living-17 | Subclass | 1076 | 1009 | Yes |
| BREEDS Nonliving-26 | Subclass | 1043 | 712 | Yes |
| CIFAR-10.001 | Data source | 1000 | 114 | Yes |
| CIFAR-10.02 | Data source | 4039 | 431 | Yes |
| FMoW-WILDS | Subclass | 34816 | 1582 | No |
| Amazon-WILDS | Subclass | 496127 | 110 | No |
| CivilComments-WILDS | Confounder | 4962 | 1003 | Yes |

# B Contrastive adapter implementation details

We provide further details on the adapter architecture and training sampling.

## B.1 Adapter architecture

Similar to prior works [22, 33], the adapters we use are bottleneck 2-layer multilayer perceptrons (MLPs). We set the input dimension and output dimensions as the same as the pretrained foundation model embedding dimension, and pick a smaller dimension for the hidden layer (frequently 128, although this was chosen as a heuristic and not tuned). We also experimented with using a single residual connection [29] and batch normalization layer [35] between the input and output layers, but only found the latter to be helpful. Pytorch-like pseudocode is given below. The adapter is visualized in Figure 7.

```
import torch.nn as nn

class Adapter(nn.Module):
    def __init__(self, input_dim, hidden_dim):
        super().__init__()
        self.arch = nn.Sequential(
            nn.Linear(input_dim, hidden_dim),
            nn.BatchNorm1d(hidden_dim),
            nn.ReLU(),
            nn.Linear(hidden_dim, input_dim)
        )
    def __forward__(self, x):
        return self.arch(x)
```

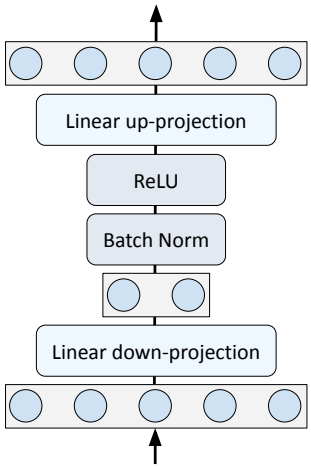


Figure 7: Adapter architecture

## B.2 Adapter training sampling

Recall in Section 4.2 that we train the adapter with both a supervised contrastive loss over specifically sampled "contrastive batches", and a cross-entropy loss over resampled batches, both over the fixed pretrained foundation model embeddings. We use the foundation model's zero-shot classification predictions to guide sampling for both.

We outline the algorithms for sampling training batches in Algorithm 1 and Algorithm 2. We then train an adapter by applying the contrastive loss (Eq. 6) and the cross-entropy loss (Eq. 5) over these batches in Algorithm 3.

**Algorithm 1** Contrastive batch sampling

**Input:** Training dataset sample embeddings $U = \{u_n\}_{n=1}^N$. Ground-truth class labels $Y = \{y_n\}_{n=1}^N$. Foundation model zero-shot predictions $\hat{Y} = \{\hat{y}_n\}_{n=1}^N$.

**Require:** Number of positives $P$ per anchor. Number of negatives $M$ per anchor. Number of nearest neighbors $M^*$ per anchor to sample negatives from.

Initialize set of contrastive batches $B = \{\}$
**for** anchor $u_a \in \{u_i \in U : \hat{y}_i \neq y_i\}$ **do**
  **(Positive sampling)**
  Sample $P$ positives $\{u_p\}_{p=1}^P$ uniform-randomly from $U$ where $\hat{y}_p = y_p$ (and $\hat{y}_p \neq \hat{y}_a$)
  **(Negative sampling)**
  Sample $M$ negatives $\{u_m\}_{m=1}^M$ by computing the $M^*$ sample embeddings with the highest cosine similarity to $u_a$ where $y_m \neq y_a$, then randomly sampling $M$ of these embeddings
  Update contrastive batch sets $B \leftarrow B \cup \left(u_a, \{u_p\}_{p=1}^P, \{u_m\}_{m=1}^M\right)$
**end for**

**Algorithm 2** Resampled training set sampling

**Input:** Training dataset sample embeddings $U = \{u_n\}_{n=1}^N$. Ground-truth class labels $Y = \{y_n\}_{n=1}^N$. Foundation model zero-shot predictions $\hat{Y} = \{\hat{y}_n\}_{n=1}^N$. All unique classes $C$.

Initialize resampled training samples $U^* = \{\}$
**for** class $c \in C$ **do**
  Identify incorrect samples $U^- = \{u_i\}$ where $\hat{y}_i \neq c$
  Identify correct samples $U^+ = \{u_i\}$ where $\hat{y}_i = c$
  Obtain upsampled samples $\tilde{U}^-$ by uniform-randomly sampling from $U^-$ s.t. $|\tilde{U}^-| = |U^+|$
  Update resampled samples $U^* \leftarrow U^* \cup \left(\tilde{U}^- \cup U^+\right)$
**end for**

**Algorithm 3** Contrastive adapting

**Input:** Set of contrastive batches $B$, resampled training samples $U^*$, number of epochs $K$.

Randomly initialize adapter $f_\theta$
**for** epoch $1, \ldots, K$ **do**
  Sample contrastive batch $\{b\}$ from $B$
  Sample randomly-shuffled minibatch of samples $\{u\}$ from $U^*$
  Update $f_\theta$ with Equation 6 over $\{b\}$
  Update $f_\theta$ with Equation 5 over $\{u\}$
**end for**

# C Additional experimental details

## C.1 Model selection and hyperparameters

We describe the hyperparameters used for each dataset and method. As in prior group robustness work [40], we select the best model and hyperparameters based on early stopping that achieves highest worst-group validation accuracy. For all methods and datasets, we train both linear probes and adapters with SGD, and sweep over learning rate $\in$ {1e-3, 1e-4, 1e-5} and weight decay $\in$ {5e-5, 5e-4, 1e-1}. For adapter classification, we used the default temperature used for zero-shot classification in CLIP [59]. We did not tune the contrastive temperature. Unless noted, we ran all numbers.

We list hyperparameters for linear probes (Table 8), adapters (Table 9, both ERM and contrastive), and contrastive-specific hyperparameters (Table 10). We discuss method-specific hyperparameters:

- **Contrastive adapting** requires selecting three additional hyperparameters: the number of positives and negatives, and the number of nearest neighbors to sample negatives from. For these we swept over the following combinations of (number positives, number negatives, number neighbors): (2048, 2048, 2146), (2048, 2048, 4096), (512, 512, 1024).
- **Weight-space ensembling** (WiSE-FT): WiSE-FT requires picking a value $\alpha \in [0, 1]$ to compute a weighted combination of the zero-shot classifier parameters and the trained linear probe parameters. We sweep over intervals of size 0.1, *i.e.* $\alpha \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$.

Table 8: Linear probe hyperparameters

| Dataset | Max Epochs | Learning Rate | Weight Decay | Momentum | Batch Size |
|---|---|---|---|---|---|
| Waterbirds | 100 | 1e-3 | 5e-5 | 0.9 | 128 |
| CelebA | 50 | 1e-3 | 5e-5 | 0.9 | 128 |
| BREEDS Living-17 | 100 | 1e-3 | 5e-5 | 0.9 | 128 |
| BREEDS Nonliving-26 | 100 | 1e-3 | 5e-5 | 0.9 | 128 |
| CIFAR-10.001 | 100 | 1e-3 | 5e-5 | 0.9 | 128 |
| CIFAR-10.02 | 100 | 1e-3 | 5e-5 | 0.9 | 128 |
| FMoW-WILDS | 100 | 1e-3 | 5e-5 | 0.9 | 128 |
| Amazon-WILDS | 100 | 1e-3 | 5e-5 | 0.9 | 16 |
| CivilComments-WILDS | 100 | 1e-3 | 5e-5 | 0.9 | 16 |

Table 9: Adapter hyperparameters. For contrastive adapters, batch size refers to the size of each minibatch sampled for updating with cross-entropy loss.

| Dataset | Max Epochs | Learning Rate | Weight Decay | Momentum | Batch Size | Hidden Dimension | Temperature |
|---|---|---|---|---|---|---|---|
| Waterbirds | 100 | 1e-3 | 5e-5 | 0.9 | 128 | 128 | 0.01 |
| CelebA | 50 | 1e-3 | 5e-5 | 0.9 | 128 | 128 | 0.01 |
| BREEDS Living-17 | 100 | 1e-3 | 5e-5 | 0.9 | 128 | 128 | 0.01 |
| BREEDS Nonliving-26 | 100 | 1e-3 | 5e-5 | 0.9 | 128 | 128 | 0.01 |
| CIFAR-10.001 | 100 | 1e-3 | 5e-5 | 0.9 | 128 | 128 | 0.01 |
| CIFAR-10.02 | 100 | 1e-3 | 5e-5 | 0.9 | 128 | 128 | 0.01 |
| FMoW-WILDS | 100 | 1e-3 | 5e-5 | 0.9 | 128 | 512 | 0.01 |
| Amazon-WILDS | 100 | 1e-3 | 5e-5 | 0.9 | 16 | 512 | 0.01 |
| CivilComments-WILDS | 100 | 1e-3 | 5e-5 | 0.9 | 16 | 512 | 0.01 |

Table 10: Specific contrastive adapter hyperparameters.

| Dataset | Number Positives | Number Negatives | Number Nearest Neighbors | Contrastive Temperature |
|---|---|---|---|---|
| Waterbirds | 2048 | 2048 | 4096 | 0.1 |
| CelebA | 2048 | 2048 | 4096 | 0.1 |
| BREEDS Living-17 | 2048 | 2048 | 4096 | 0.1 |
| BREEDS Nonliving-26 | 512 | 512 | 1024 | 0.1 |
| CIFAR-10.001 | 512 | 512 | 1024 | 0.1 |
| CIFAR-10.02 | 512 | 512 | 1024 | 0.1 |
| FMoW-WILDS | 2048 | 2048 | 2146 | 0.1 |
| Amazon-WILDS | 2048 | 2048 | 2146 | 0.1 |
| CivilComments-WILDS | 2048 | 2048 | 2146 | 0.1 |

### C.2 Data splits

We use the same train, validation, and test splits for Waterbirds, CelebA, FMoW-WILDS, Amazon-WILDS, and CivilComments-WILDS as in prior work. For BREEDS and CIFAR datasets that we adapt for our problem setting, we construct test splits by combining official test splits from the original benchmarks. We then create training and validation splits by combining the rest of the data from these benchmarks, and randomly splitting this into 80% training data and 20% validation data. No original test data is seen during training on our splits.

### C.3 Additional dataset assets details and discussion

**Dataset licenses.** To curate CIFAR-10.0001 we use the CIFAR-10.1 dataset, which is distributed under the MIT License. The FMoW-WILDS dataset is distributed under the FMoW Challenge Public License[6]. The CivilComments-WILDS dataset is distributed under CC0 1.0. The Amazon-WILDS dataset does not have a license, but is requested to be used for research purposes only [40]. We were not able to find explicit license information for CIFAR-10.2, Waterbirds, CelebA, or the BREEDS datasets. We note that the BREEDS datasets are sourced from ImageNet, which is distributed under the BSD 3-Clause License, and set up with code from the MadryLab robustness GitHub repository[7], which is distributed under a MIT license. The authors of the CelebA dataset provide a list of agreements[8], including that the dataset is used only for non-commercial research purposes.

**Existing assets personally identifiable information and offensive content.** The CelebA dataset consists of images of celebrity faces, which are personally identifiable. The dataset is also categorized

[6] https://github.com/fMoW/dataset/blob/master/LICENSE

[7] https://github.com/MadryLab/robustness

[8] http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

Table 11: Class prompt templates or example prompts

| Dataset | Foundation Model | Prompt template / example of prompt |
|---|---|---|
| Waterbirds | CLIP<br>CLOOB | "This is a picture of a `[class_name]`."<br>"a `[class_name]`" |
| CelebA | CLIP<br>CLOOB | "A photo of a celebrity with blond hair."<br>"A photo of a celebrity with blond hair." |
| BREEDS Living-17 | CLIP<br>CLOOB | "This is a picture of a `[class_name]`."<br>"This is a picture of a `[class_name]`." |
| BREEDS Nonliving-26 | CLIP<br>CLOOB | "A photo of a `[class_name]`."<br>"a `[class_name]`" |
| CIFAR-10.001 | CLIP<br>CLOOB | "a `[class_name]`"<br>"a `[class_name]`" |
| CIFAR-10.02 | CLIP<br>CLOOB | "a `[class_name]`"<br>"a `[class_name]`" |
| FMoW-WILDS | CLIP<br>CLOOB | "satellite view of the `[class_name]`"<br>"aerial view of an `[class_name]`" |
| Amazon-WILDS | GPT-Neo | "Negative" |
| CivilComments-WILDS | GPT-Neo | "Not toxic" |

by male and female identification at the time of curation, which may be outdated. The CivilComments-WILDS contains text samples flagged as toxic by toxicity classifiers [40], which contain potentially offensive content. Both datasets are existing assets, and both personal identifiability for CelebA and offensive content for CivilComments-WILDS can be checked by inspecting the original data inputs (images and text comments).

### C.4 Compute and resources

All experiments were run on a machine with 14 CPU cores and a single NVIDIA Tesla P100 GPU. For training a contrastive adapter on top of CLIP ResNet-50 Waterbirds embeddings, this took approximately 30 minutes to run 100 epochs. Other than the numbers reported from their original publications in Table 6, we report all numbers from running experiments on the same machine.

### C.5 Class prompt templates

In Table 11, we list the templates used to generate class prompts for each dataset. As a reminder, for each provided class name in a dataset, we create a prompt by inserting the class name into the prompt template. We then encode this prompt with a foundation model text encoder to get class embeddings. We selected these prompts for each dataset by trying several different prompt templates per dataset, and selecting the best prompt template based on validation set worst-group accuracy. Table 20 contains a full list of all templates tried.

## D Additional related work discussion

We provide additional discussion of related work and connections to our work below.

**Zero-shot classification with foundation models.** Our work builds on a growing literature on applying foundation models, large pretrained models that can be applied to various downstream tasks. These models demonstrate exciting promise in their ability to achieve accurate downstream transfer *without* any additional finetuning [9, 11, 59]. In particular we consider the zero-shot capabilities of pretrained vision-language foundation models. These models, such as CLIP [59], ALIGN [36], and CLOOB [20] are trained on massive amounts of naturally paired image-text data, *e.g.*, Internet

images and their corresponding captions. Consisting of an image encoder (usually a ResNet or Vision Transformer) and a text encoder (usually a Transformer), such foundation models are commonly trained to learn a shared image-text embedding space where embeddings of images are most similar to embeddings of their corresponding caption text. While these objectives have been shown to lead to powerful representations [18, 81], a crucial element for successful zero-shot classification is training data scale [59]. However, added scale can also be a double-edged sword; when zero-shot classification still makes undesirable mistakes, standard ways to correct for these mistakes via retraining can become prohibitively expensive. We study one such motivating instance via group robustness, and provide a first-step solution towards improving group robustness efficiently.

**Robustness of foundation models.** Prior works have studied the robustness of foundation model inference to natural distribution shifts. Radford et al. [59] show that zero-shot CLIP models can be more robust to out-of-distribution (OOD) shifts than prior ImageNet-trained models, measured via better generalization to various dataset-level distribution shifts on ImageNet classes [4, 30, 32, 62, 74]. However, they also show that finetuning, or updating the original weights, of CLIP models on ImageNet can reduce this OOD robustness. Kumar et al. [42], Wortsman et al. [76] thus propose finetuning methods that improve downstream in-distribution accuracy while maintaining out-of-distribution robustness. Kumar et al. [42] specifically study the trade-off between linear probing and finetuning, finding that finetuning on downstream data can improve generalization on in-distribution data over linear probing but more substantially hurt performance OOD data than linear probing. They show theoretically and empirically that a two-step strategy of first linear probing then full fine-tuning can combine the performance boosts of both. Wortsman et al. [76] focus on the OOD trade-off presented by Radford et al. [59] between a finetuned foundation model and its pretrained zero-shot weights. They propose weight-space ensembling (WiSE-FT), which computes a weighted average of the finetuned and pretraiend foundation model parameters, and show that the resulting averaged parameters can in some instances achieve higher performance on both data distributions that the model was finetuned on and unseen OOD data than the initial finetuned and zero-shot or pretrained models. They show this effect with both full finetuning and training a linear probe. Unlike these works, we focus on foundation model robustness to group shifts that occur within a dataset. We also compare against the linear probe version of WiSE-FT, and find that training adapters can be advantageous for achieving higher group robustness on various datasets.

Recently, other works also study foundation model learned spurious correlations and biases. Singla et al. [70] show how various models (including CLIP models) may rely on spurious artifacts to classify ImageNet images. Berg et al. [6] aim to debias CLIP image embeddings of human faces using extra metadata (textual concepts or attributes) that the embeddings should ignore. Our evaluation is complementary, noting poor group robustness across multiple types of data sources (objects, animals, human faces, text). We also provide a method that works without additional training metadata.

**Improving group robustness of deep learning models.** Improving the group robustness of deep learning models is a common deep learning challenge, where models may learn biases during training that lead to poor performance on certain groups. This is a widespread issue presented in contexts ranging from algorithmic fairness to healthcare diagnosis [8, 12, 27]. Several methods exist to improve group robustness. We compare against several recent approaches in Section 5.3. While one effective strategy to improve group robustness is to upweight the error of worst-performing group during training [65, 66], training group labels may be impractical to obtain in practice [55, 71]. We thus consider robustness approaches which aim to work without training group labels. Several approaches involve training two models; one model is first trained with standard ERM to help infer groups, and another trained with a robust objective using these inferred groups. Just Train Twice (JTT) [47] treats samples that the first model misclassifies as inferred minority group samples to upweight. JTT then upweights these samples by a hyperparameter factor, and trains a second model with ERM on this upsampled data. Environment Inference for Invariant Learning (EIIL) [16] infers groups by assigning samples to group under which the ERM model maximally violates an Invariant Risk Minimization [2] principle. It then trains a robust model with Group DRO [65] using the inferred groups, which dynamically upweights the worst-performing groups during training. Correct-N-Contrast (CNC) [79] instead identifies samples with the same class labels but different ERM model predictions, and trains a robust model by using a contrastive loss to learn similar representations between these samples. Spread Spurious Attribute (SSA) [52] specifically trains the first model to predict groups using a small set of group labels, before using Group DRO to train a robust model. Contrastive Input Morphing (CIM) [72] trains a network to transform the input features of an image

to better present class-specific information shared across groups. Idrissi et al. [34] suggest that simply changing the training data by subsampling large classes (SUBY) or balancing the class sampling probabilities (RWY), then training a model with ERM, can also improve group robustness.

# E Additional experimental results

## E.1 Extended main results

In Table 12 and Table 13 we report group robustness results evaluating all methods discussed in Section 5.1 on all group robustness benchmarks. Table 12 contains results for image datasets, using CLIP-RN50 embeddings. Table 13 contains results for text datasets, using GPT-Neo 1.3B embeddings. As in Table 3, we report the worst-group and average accuracies, along with their gap. Higher worst-group accuracy and smaller accuracy gap are indicative of better group robustness. All results are computed over three random seeds, with mean and one standard deviation included (error bars deferred to here from the main paper). Compared to alternative methods, contrastive adapting consistently improves group robustness over zero-shot classification, and obtains highest worst-group accuracy and smallest accuracy gap on datasets where training adapters with ERM fails. On datasets where ERM-trained adapters achieve best group robustness, contrastive adapters are also competitive or closest to ERM-trained adapters among other robustness methods.

## E.2 Contrastive adapter ablations

In this section, in extension to Section 5.2 we ablate different training components of contrastive adapting. We first study how ablating the training objective (cross-entropy and contrastive losses) affects worst-group accuracy across multiple pretrained model embeddings on the Waterbirds dataset. We then study how the number of positives and negatives used in contrastive sampling affects performance. Our results suggest that the full combination of contrastive objective and hard sampling corresponds to best group robustness across models on Waterbirds, and that contrastive adapters also benefit from training with larger batches of positive and negative samples.

### E.2.1 Ablation on loss components in training objective

We first study the importance of the contrastive and cross-entropy components in contrastive adapting. For evaluation, we use the Waterbirds dataset, and run ablations comparing adapters trained on top of CLIP embeddings with (i) no contrastive component (Eq. 6), (ii) no cross-entropy component (Eq. 5), or the default proposed approach. We evaluate across five different CLIP models and three seeds. We keep all other training procedures consistent (*e.g.*, we use the proposed "hard" sampling strategy).

In Table 14, we report worst-group accuracies. We find that both contrastive and cross-entropy components are necessary for best worst-group accuracy. The contrastive objective leads to a substantial improvement over just the resampled cross-entropy loss (+17.9 pp on average). However, we also note that without the cross-entropy objective to learn sample embeddings close to their ground-truth class embeddings, we observe high variance in classification accuracy. We improve +26.9 pp on average using both objectives compared to contrastive alone.

### E.2.2 Effect of contrastive batch size

While one advantage of training adapters is that because we train on embeddings, the memory size of our data inputs during training is much smaller than the traditional alternative (*e.g.*, storing an tensorized image). We can thus train with larger batch sizes. Here we study how contrastive batch size, *i.e.* how many positives and negatives we sample per anchor, affects worst-group accuracy. On the Waterbirds and CelebA datasets and with CLIP RN-50 embeddings, we train a contrastive adapter with varying levels of positives and negatives. For both datasets, the default is 2048 positives and 2048 negatives per batch. We ablate these numbers with the following (positive, negative) combinations: (1, 1), (2, 2), (256, 256), (256, 512), (512, 256), (512, 512), (512, 1024), (1024, 512), (1024, 1024), (1024, 2048), (2048, 1024).

In Figure 8, we plot the effect of smaller batch sizes on worst-group accuracy. We find that larger batch sizes weakly correspond to higher worst-group accuracy on both Waterbirds and CelebA. However, perhaps surprisingly, we still maintain a substantial improvement over zero-shot classification with just a single positive and negative per anchor.

Table 12: Worst-group (WG) and average (Avg) accuracies (in %) for zero-shot and efficient methods to improve CLIP-RN50 inference. **1st** / 2nd highest WG acc. and **1st** / 2nd smallest accuracy gap **bolded** / underlined respectively. Contrastive adapter (Contrast. Adapter) bolded for visibility.

| Waterbirds | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Acc. | Zero-Shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $49.8 \pm 0.0$ | $55.9 \pm 0.0$ | $7.9 \pm 1.0$ | $60.8 \pm 0.9$ | $49.8 \pm 0.0$ | $51.3 \pm 1.4$ | $63.9 \pm 1.5$ | $\mathbf{83.7 \pm 0.7}$ |
| Avg. | $91.0 \pm 0.0$ | $87.8 \pm 0.0$ | $93.5 \pm 0.1$ | $96.0 \pm 0.1$ | $91.0 \pm 0.0$ | $92.4 \pm 0.1$ | $91.8 \pm 3.1$ | $89.4 \pm 0.9$ |
| Gap | 41.2 | 31.9 | 85.6 | 35.2 | 41.2 | 41.1 | 27.9 | **5.7** |
| CelebA | | | | | | | | |
| Acc. | Zero-Shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $74.0 \pm 0.0$ | 70.8 | $11.9 \pm 0.3$ | $36.1 \pm 1.4$ | $85.6 \pm 0.0$ | $76.9 \pm 1.4$ | $89.6 \pm 0.3$ | $\mathbf{90.0 \pm 0.4}$ |
| Avg. | $81.9 \pm 0.0$ | 82.6 | $94.7 \pm 0.0$ | $94.2 \pm 0.2$ | $88.6 \pm 0.0$ | $92.5 \pm 0.2$ | $91.8 \pm 0.1$ | $90.7 \pm 0.0$ |
| Gap | 7.9 | 11.8 | 82.8 | 58.1 | 3.0 | 15.6 | 2.2 | **0.7** |
| BREEDS Living-17 | | | | | | | | |
| Acc. | Zero-Shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $6.0 \pm 0.0$ | $30.0 \pm 0.0$ | $53.3 \pm 0.9$ | $\mathbf{70.7 \pm 0.9}$ | $53.3 \pm 0.9$ | $46.7 \pm 3.4$ | $44.0 \pm 0.0$ | $62.0 \pm 1.6$ |
| Avg | $86.7 \pm 0.0$ | $90.6 \pm 0.0$ | $90.8 \pm 0.0$ | $94.0 \pm 0.1$ | $90.8 \pm 0.0$ | $89.3 \pm 0.3$ | $86.4 \pm 0.0$ | $90.9 \pm 0.3$ |
| Gap | 80.7 | 60.6 | 37.5 | **23.2** | 37.5 | 42.6 | 42.4 | 28.9 |
| BREEDS Nonliving-26 | | | | | | | | |
| Acc. | Zero-Shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $6.0 \pm 0.0$ | $56.0 \pm 0.0$ | $32.0 \pm 0.0$ | $\mathbf{61.3 \pm 1.9}$ | $36.7 \pm 0.9$ | $29.3 \pm 1.9$ | $30.0 \pm 4.1$ | $55.3 \pm 4.2$ |
| Avg | $72.3 \pm 0.0$ | $87.1 \pm 0.0$ | $82.3 \pm 0.1$ | $92.1 \pm 0.2$ | $83.6 \pm 0.1$ | $80.6 \pm 0.1$ | $83.6 \pm 0.0$ | $88.1 \pm 0.6$ |
| Gap | 66.3 | 31.1 | 50.3 | **30.8** | 46.9 | 51.3 | 53.6 | 32.8 |
| CIFAR-10.001 | | | | | | | | |
| Acc. | Zero-Shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $31.4 \pm 0.0$ | N/A | $44.0 \pm 1.4$ | $\mathbf{68.2 \pm 3.5}$ | $53.3 \pm 0.0$ | $18.1 \pm 4.3$ | $45.0 \pm 1.6$ | $59.7 \pm 4.1$ |
| Avg | $69.8 \pm 0.0$ | N/A | $75.2 \pm 0.2$ | $87.3 \pm 0.3$ | $81.1 \pm 0.0$ | $58.7 \pm 1.7$ | $78.3 \pm 0.1$ | $82.0 \pm 0.1$ |
| Gap | 38.4 | N/A | 31.2 | **19.1** | 27.8 | 40.6 | 33.3 | 22.3 |
| CIFAR-10.02 | | | | | | | | |
| Acc. | Zero-Shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $39.1 \pm 0.0$ | N/A | $51.3 \pm 0.2$ | $\mathbf{68.8 \pm 0.5}$ | $58.2 \pm 0.2$ | $45.0 \pm 0.8$ | $38.5 \pm 2.1$ | $60.7 \pm 1.7$ |
| Avg | $69.9 \pm 0.0$ | N/A | $77.7 \pm 0.1$ | $86.0 \pm 0.5$ | $79.1 \pm 0.0$ | $75.0 \pm 0.3$ | $77.9 \pm 0.5$ | $80.9 \pm 0.2$ |
| Gap | 48 | N/A | 26.4 | **17.2** | 20.9 | 30.0 | 39.4 | 20.2 |
| FMoW-WILDS | | | | | | | | |
| Acc. | Zero-shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $10.5 \pm 0.0$ | - | $21.6 \pm 0.1$ | $\mathbf{41.3 \pm 0.5}$ | $21.6 \pm 0.1$ | $6.8 \pm 0.6$ | $27.0 \pm 0.2$ | $39.2 \pm 0.7$ |
| Avg | $13.2 \pm 0.0$ | - | $24.1 \pm 0.1$ | $43.6 \pm 0.5$ | $24.1 \pm 0.1$ | $10.2 \pm 0.5$ | $28.7 \pm 0.2$ | $41.9 \pm 0.1$ |
| Gap | 2.7 | - | 2.5 | 2.3 | 2.5 | 3.4 | **1.7** | 2.7 |

Table 13: Worst-group (WG) and average (Avg) accuracies (in %) for zero-shot and efficient methods to improve GPT-Neo 1.3B inference. **1st** / 2nd highest WG acc. and **1st** / 2nd smallest accuracy gap **bolded** / underlined respectively. Contrastive adapter (Contrast. Adapter) bolded for visability.

| Amazon-WILDS | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Acc. | Zero-shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $79.4 \pm 0.0$ | N/A | $87.2 \pm 0.3$ | $87.2 \pm 0.3$ | $87.2 \pm 0.3$ | $87.2 \pm 0.3$ | $85.4 \pm 0.8$ | $\mathbf{87.9 \pm 1.1}$ |
| Avg | $86.7 \pm 0.0$ | N/A | $93.3 \pm 0.2$ | $93.6 \pm 0.1$ | $93.3 \pm 0.2$ | $93.2 \pm 0.3$ | $92.7 \pm 0.7$ | $92.6 \pm 0.8$ |
| Gap | 7.3 | N/A | 6.1 | 6.4 | 6.1 | 6.0 | 7.3 | **4.7** |
| CivilComments-WILDS | | | | | | | | |
| Acc. | Zero-shot | Group Prompt | ERM LP | ERM Adapter | WiSE-FT | DFR (Sub) | DFR (Up) | **Contrast. Adapter** |
| WG | $16.0 \pm 0.0$ | N/A | $46.7 \pm 2.0$ | $32.1 \pm 1.5$ | $46.7 \pm 2.0$ | $47.4 \pm 0.9$ | $48.2 \pm 1.3$ | $\mathbf{50.1 \pm 1.5}$ |
| Avg | $74.8 \pm 0.0$ | N/A | $51.2 \pm 0.26$ | $37.7 \pm 0.7$ | $51.2 \pm 0.26$ | $51.9 \pm 0.8$ | $52.1 \pm 1.3$ | $54.2 \pm 0.5$ |
| Gap | 58.8 | N/A | 4.5 | 5.6 | 4.5 | 4.5 | **3.9** | 4.1 |

### E.3 Comparison to TIP-adapter and training sample nearest-neighbors lookup

On representative benchmarks, we perform further comparison to the nearest-neighbor look-up approach employed by TIP Adapter [80]. Instead of learning transformed representations of pretrained

Table 14: Contrastive adapter training objective ablation. For five CLIP models, we report the worst-group accuracy (%) on Waterbirds (mean, std. dev. over three seeds). Having both contrastive (Eq. 6) and cross-entropy (Eq. 5) objectives lead to best worst-group accuracy. Removing Eq. 5 (which keeps sample embeddings close to class embeddings), also leads to higher performance variance.

| Adapter Method Ablation | RN-50 | RN-101 | ViT-B/32 | ViT-B/16 | ViT-L/14 |
|---|---|---|---|---|---|
| No contrastive (Eq. 6) | $56.3 \pm 1.5$ | $68.8 \pm 2.2$ | $56.7 \pm 2.4$ | $70.2 \pm 1.4$ | $75.1 \pm 1.0$ |
| No cross-entropy (Eq. 5) | $60.7 \pm 8.3$ | $37.8 \pm 12.0$ | $23.1 \pm 10.5$ | $77.7 \pm 2.9$ | $82.4 \pm 2.0$ |
| Default | $\mathbf{83.7 \pm 0.7}$ | $\mathbf{82.0 \pm 1.3}$ | $\mathbf{80.7 \pm 1.4}$ | $\mathbf{83.1 \pm 2.1}$ | $\mathbf{86.9 \pm 1.6}$ |

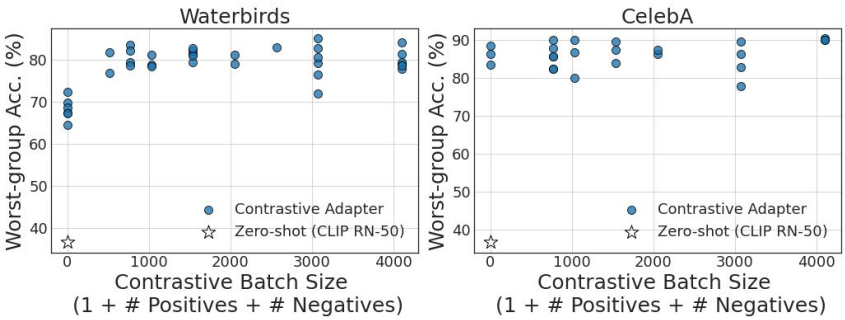


Figure 8: Effect of contrastive batch size on worst-group accuracy. With CLIP RN-50 embeddings, training contrastive adapters with larger batch sizes (greater number of positives and negatives) tends to help worst-group accuracy. However, even training with one positive and negative per batch leads to substantially greater worst-group accuracy than zero-shot classification.

Table 15: Group robustness comparison to nearest training sample look-up / TIP Adapter [80]. Across representative benchmarks, on average contrastive adapting achieves 30.4 pp higher worst-group accuracy than the nearest training sample look-up employed by TIP-adapter. This supports learning non-linear transformations of pretrained embeddings to better classify samples.

| | Waterbirds | | | CelebA | | | BREEDS Living-17 | | | CIFAR-10.02 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Acc (%) | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap |
| Zero-shot (ZS) | 36.6 | 92.2 | 55.6 | 74.0 | 81.9 | 7.9 | 6.0 | 86.7 | 80.7 | 39.1 | 69.9 | 30.8 |
| TIP Adapter | 39.9 | 93.9 | 54.0 | 19.4 | 91.1 | 71.7 | **64.0** | 90.7 | **26.7** | 51.5 | 75.4 | 23.9 |
| Contrastive Adapter | **83.7** | 89.4 | **5.7** | **90.0** | 90.7 | **0.7** | 62.0 | 90.9 | 28.9 | **60.7** | 80.9 | **20.2** |

embeddings, another approach to better classify a test sample is to use the class of its nearest training sample. Under the assumption that the training and test data are sampled from the same broader distribution and share the same groups, then test samples in a given group should embed closest to training samples in the same group. The training sample ground-truth class should then apply to the test sample. TIP adapter operates accordingly, keeping a cache of training sample embeddings available at test-time. One advantage is this allows for potentially more accurate classification *without any* training. To test how well this idea fares for group robust classification, for each test sample we perform a look-up with *all* training samples, using cosine similarity to identify nearest neighbors.

In Table 15, we compare TIP-adapter with zero-shot classification and contrastive adapting on the Waterbirds, CelebA, BREEDS Living-17, and CIFAR-10.02 group robustness benchmarks. For all methods, we use CLIP RN-50 pretrained embeddings. We find that TIP adapter improves worst-group accuracy over zero-shot classification on 3 out of 4 datasets, and notably achieves best worst-group accuracy on BREEDS Living-17 without training any additional parameters. However, the improvements are more marginal on Waterbirds and CIFAR-10.02. Contrastive adapting still achieves 30.4 pp higher worst-group accuracy over TIP adapter on average. This may suggest that learning a nonlinear transformation of the pretrained embeddings can still be helpful for better "presenting" class-specific information to classify samples by.

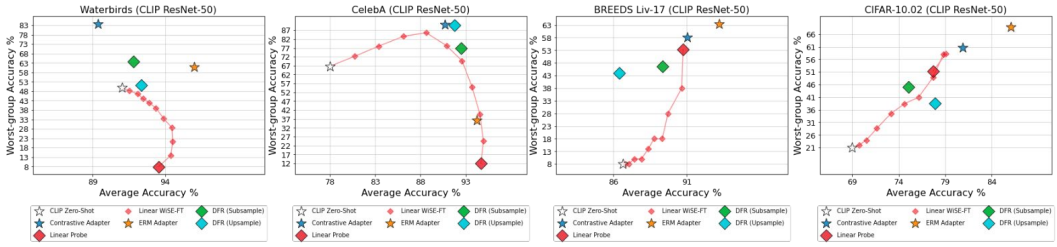


Figure 9: Plotting worst-group versus average accuracy trade-off against WiSE-FT ensemble (traced out) on representative datasets (Table 3). Contrastive adapters (dark blue stars) consistently achieve higher worst-group accuracy than weight-space ensembles.

### E.4 Evaluation with respect to weight-space ensembling trade-off

To provide additional perspective on how different embedding-only methods trade-off worst-group and average accuracy, we compare how these methods perform with respect to the accuracy trade-off traced out by weight-space ensembles. Wortsman et al. [76] show an interesting phenomenon where simply taking a weighted average of a trained linear probe and the original foundation model (either over the weights, or the outputs) can result in a "pareto frontier" of accuracy metrics. They specifically show that a weight-space ensemble can often achieve better OOD performance without sacrificing too much IID performance. compared to a single linear probe. In this context, we see how this trade-off occurs over average accuracy and worst-group accuracy across our representative set of group-robustness datasets. We also evaluate how other approaches (ERM adapters, DFR [39], contrastive adapters) fare along this trade-off.

In Figure 9, we plot the accuracies of these methods run on CLIP RN-50 embeddings. We note several observations. Weight-space ensembles (WiSE-FT) achieve the desired effect on certain datasets but not others. On CelebA and CIFAR-10.02, we find that an ensemble can obtain a better worst-group accuracy versus average accuracy trade-off than either zero-shot classification or linear probes. However, the single linear probe does at least as well as any ensemble in BREEDS Living-17, while the zero-shot classification does at least as well as any ensemble in Waterbirds.

We also find that among other methods, contrastive adapting is the only evaluated approach that consistently achieves higher worst-group accuracy than any weight-space ensemble. While contrastive adapting places "above" the trade-off curve traced out by WiSE-FT on 3 out of 4 datasets (CelebA, BREEDS Living-17, and CIFAR-10.02), it tends to degrade average performance in favor of higher worst-group performance compared to other approaches. Further work can improve on how to raise worst-group performance without sacrificing any average performance when compared to zero-shot classification or ERM-trained adapters.

### E.5 Comparison to recent robustness methods adapted for the pretrained embedding setting

The scope of this paper focused on identifying the poor group robustness of foundation models and determining how to efficiently improve this robustness. Thus, to judge the effectiveness of our proposed approach, our main evaluation considered existing methods for efficiently improving pretrained model inference, *i.e.*, with no model retraining and only access to pretrained embeddings.

Here, in an initial study on how we can adapt more robustness techniques for efficiently improving group robustness, we consider how recent robustness methods proposed for standard model training (as introduced in Section 5.3) can *transfer* to the pretrained model setting. We adapt methods such as Just Train Twice (JTT) [47], Correct-N-Contrast (CNC) [79], and Just Mix Once (JM1) [24] to train adapters, and compare these methods to the proposed contrastive adapting on the representative benchmarks in the main results (Section 5.1). In Section E.5.1, we describe how we adapt the method to the pretrained model setting. In Section E.5.2, we discuss key hyperparameters and our sweeps. Finally, in Section E.5.3 we report results and comparison takeaways.

Table 16: Hyperparameters for robustness methods adapted to the pretrained embedding setting

| Dataset | Waterbirds | | | CelebA | | | BREEDS Liv-17 | | | CIFAR-10.02 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | JTT | CNC | JM1 | JTT | CNC | JM1 | JTT | CNC | JM1 | JTT | CNC | JM1 |
| LR | 1e-3 | 1e-3 | 1e-3 | 1e-3 | 1e-3 | 1e-3 | 1e-3 | 1e-3 | 1e-3 | 1e-3 | 1e-3 | 1e-3 |
| Weight Decay | 5e-5 | 5e-5 | 1e-1 | 5e-5 | 5e-5 | 1e-1 | 5e-5 | 5e-5 | 5e-5 | 5e-5 | 5e-5 | 5e-5 |
| Momentum | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 |
| Batch Size | 128 | 128 | 128 | 128 | 128 | 128 | 128 | 128 | 128 | 128 | 128 | 128 |
| Stage 1 Epochs | 0 | 5 | 0 | 0 | 0 | 0 | 5 | 0 | 0 | 2 | 1 | 0 |

### E.5.1 Method overview for adapters

- **Just Train Twice (JTT)** [47]: Following the two-stage process in JTT, we first train an initial ERM adapter (via the cross-entropy loss in Eq. 5) for a few epochs and upsample the samples that the adapter gets incorrect. We then train a new adapter with the same objective but on the resampled data. Alternatively, we can also use the zero-shot pretrained model predictions for sampling.
- **Correct-N-Contrast (CNC)** [79]: Like JTT, CNC also employs a two-stage process. We first train an initial ERM adapter like in JTT, but instead of upsampling the incorrect samples, use the first ERM adapter's predictions to set up a contrastive learning dataset in the second stage. Following [79], we obtain contrastive batches by sampling anchors and positives as samples with the same class, but different initial ERM adapter predictions. We identify anchors and negatives as samples with different classes, but the same initial ERM adapter predictions. We then train a second adapter with the contrastive loss in Eq. 6 on this training set. Alternatively, we can also use the zero-shot pretrained model predictions for sampling.
- **Just Mix Once (JM1)** [24]: We similarly first identify samples with the same class but different predictions, using predictions from either training an initial ERM adapter for a few epochs, or using the pretrained model zero-shot predictions. Then to train a robust adapter, during each training epoch we compute a set of interpolated sample embeddings by interpolating embeddings for samples in the same class but different predictions (*i.e.* $u = \alpha u_1 + (1-\alpha)u_2$ if $u_1, u_2$ are pretrained sample embeddings that fit the criteria). We follow the proposed strategy [24] for randomly sampling $\alpha$, alternating between a Uniform(0, 1) and Beta(2, 5) distribution. As the embeddings have the same class, we use the original class label as the interpolated sample label. We train another adapter on the interpolated embeddings using the cross-entropy loss (Eq. 5).

### E.5.2 Training details and hyperparameters

We evaluate the group robustness after training adapters on CLIP ResNet-50 pretrained embeddings with each method. For all methods, we sweep over the same adapter hyperparameter set in Section C.1. As each method trains an initial "stage 1" ERM model, before using this model's predictions to setup a robust training stage, we also try this. In addition to using the pretrained model's zero-shot predictions, as a hyperparameter we try sourcing predictions from an initial ERM adapter after training for 1, 2, 5, or 10 epochs. In Table 16, we list the optimal hyperparameters for each method and dataset. We use the same learning rate, weight decay, momemtum, and optimizer for both adapters. We select models with early stopping based on best worst-group validation accuracy. When the zero-shot predictions lead to best validation set accuracy, we denote this by Stage 1 Epochs = 0.

Following recommendations for fair group robustness comparison [26], we try to keep the total size of all hyperparameter combinations consistent across methods. Thus for JTT, we do not tune the upsampling ratio; we upsample samples by the class-specific ratios that equate the number of correct and incorrect samples per class (intuitively balancing groups to improve group robustness). For CNC, we pick the same number of positives and negatives for each dataset as in Table 10.

### E.5.3 Results

In Table 17 we report results on representative group robustness benchmarks, comparing contrastive adapting to recent robustness techniques applied to training adapters. We find that among methods, the proposed contrastive adapting strategy obtains best or competitive group robustness on all datasets. For three out of four datasets (Waterbirds, CelebA, and BREEDS Living-17), contrastive adapting obtains best worst-group accuracy, and on average across all datasets outperforms the next best

Table 17: Group robustness of recent robustness methods adapted for training adapters on top of pretrained foundation model embeddings. Worst-group (WG), Average (Avg) and worst-group versus average accuracy gap (Gap) reported over three seeds. Contrastive adapting (Ours) achieves best worst-group (WG) accuracy on three of four benchmarks, and competitive worst-group versus average accuracy gap on all datasets. On average, this results in a 1.7 pp increase in worst-group accuracy, and a 0.5 pp decrease in the worst-group versus average accuracy gap, over the next best method.

| | Waterbirds | | | CelebA | | | BREEDS Liv-17 | | | CIFAR-10.02 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Acc. (%) | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap | WG | Avg | Gap |
| JTT | 69.2 ± 2.3 | 85.1 ± 0.4 | 15.9 | 86.3 ± 0.9 | 88.2 ± 0.4 | 1.9 | 61.4 ± 2.5 | 91.4 ± 0.3 | 30.0 | 63.5 ± 1.3 | 81.7 ± 0.2 | 18.2 |
| CNC | 82.7 ± 2.0 | 87.0 ± 1.1 | **4.3** | 86.9 ± 1.4 | 91.8 ± 0.4 | 4.9 | 60.7 ± 2.5 | 88.1 ± 0.7 | **27.4** | 59.2 ± 2.5 | 80.4 ± 0.2 | 21.2 |
| JM1 | 74.2 ± 3.1 | 80.4 ± 1.0 | 6.2 | 87.1 ± 1.1 | 91.6 ± 1.0 | 4.5 | 61.7 ± 2.0 | 91.1 ± 0.1 | 29.4 | **65.2 ± 0.9** | 82.6 ± 0.2 | **17.4** |
| Ours | **83.7 ± 0.7** | 89.4 ± 0.9 | 5.7 | **90.0 ± 0.4** | 90.7 ± 0.4 | **0.7** | **62.0 ± 1.6** | 90.9 ± 0.3 | 28.9 | 60.7 ± 1.7 | 80.9 ± 0.2 | 20.2 |

method by 1.7 pp. Contrastive adapting also obtains competitive gaps between worst-group and average accuracy. Notably, while the application of CNC to adapters also involves supervised contrastive learning over sample embeddings, contrastive adapting outperforms CNC on all datasets. We preliminarily hypothesize that the negative sampling procedure in contrastive adapting, which focuses on the *nearest neighbors* of each anchor, and not just incorrectly predicted samples with the same class (as in CNC), samples "harder" negatives that the initial pretrained model embeddings more erroneously embeds closer to the anchors. By focusing on "pulling" these sample embeddings apart, contrastive adapting may thus more effectively transform the pretrained representations for better group robust classification, as discussed in Section 4.1.

Building on these results and our initial proposed method, we believe (1) a more comprehensive evaluation of how well existing robust training methods can transfer to efficiently improving foundation model inference, and (2) a deeper exploration into how method differences impact performance for training on large pretrained model embeddings, are interesting grounds for future work.

### E.6 Additional study on how dataset properties impact adapter robustness

In this section, we try to further understand what properties of the representative benchmarks in the main results (Section 5.1) may explain why ERM adapters achieve better group robustness on BREEDS Living-17 and CIFAR-10.02 than contrastive adapters (c.f. Table 3). Our study also helps understand why contrastive adapters work significantly better on Waterbirds and CelebA.

We consider two hypotheses for the relative performance improvement of ERM adapters on CIFAR-10.02 and BREEDS Living-17, but not Waterbirds and CelebA. First, in Appendix E.6.1, we discuss how the lack of group shift in CIFAR-10.02 may result in ERM adapters and linear probes performing best among all methods. Second in Appendix E.6.2, we explore how the group balance in the BREEDS Living-17 training data may also enable ERM adapters to best improve group robustness.

In each subsection, we provide empirical validation supporting or opposing these hypotheses. Our findings suggest that rather than these settings being challenging for contrastive adapting, the CIFAR-10.02 and BREEDS Living-17 datasets as-is are more optimal for ERM training to achieve good group-robustness. When we make the BREEDS Living-17 task harder for group-robust classification, we find that contrastive adapting outperforms ERM training. Furthermore, this harder BREEDS task exhibits the same combination of significant group shift and group imbalance present in Waterbirds and CelebA (where contrastive adapting outperforms ERM training by a large margin), suggesting that contrastive adapters work particularly well in these harder group robustness settings.

#### E.6.1 Lack of group shift in CIFAR-10.02

First, for CIFAR-10.02, we hypothesize that ERM adapters achieve high worst-group *and* average accuracy on CIFAR-10.02 because there are not significant distribution shifts between individual groups in each class. While we curated benchmarks to intentionally include different groups and induced minority groups in CIFAR-10.02, if there are not significant group shifts between different groups, then training via ERM on all of the data may still allow adapters to learn generalizable correlations from the majority groups that apply to all groups. Thus using all of the available data as in ERM can be optimal. Meanwhile, the contrastive adapter protocol may restrict training to a certain subset of the available data via the hard sampling strategy, relatively hurting performance.

Table 18: To evaluate if datasets exhibit significant distribution shifts between groups, we report the worst-group (WG) and average (Avg) test accuracy of ERM adapters after training on training data with the minority groups removed (No Minority Group). For comparison, we include performance with default train data (Default Train). High worst-group accuracy (in blue, c.f. **CIFAR-10.02**) suggests even after only training on majority groups, ERM adapters still generalize to unseen test minority groups, pointing to little group-shift between groups (*i.e.*, where ERM is expected to suffice). All other datasets exhibit significant group shifts, via drops in WG acc. to below zero-shot (in red).

| | Waterbirds | | CelebA | | BREEDS Liv-17 | | CIFAR-10.02 | |
|---|---|---|---|---|---|---|---|---|
| Acc. (%) | WG | Avg | WG | Avg | WG | Avg | WG | Avg |
| Default Train | 60.8 ± 0.9 | 96.0 ± 0.1 | 36.1 ± 1.4 | 94.2 ± 0.2 | 70.7 ± 0.9 | 94.0 ± 0.1 | 68.8 ± 0.5 | 86.0 ± 0.5 |
| No Minority Group | 28.7 ± 1.0 | 95.3 ± 0.1 | 30.6 ± 1.1 | 94.7 ± 0.2 | 6.5 ± 0.9 | 85.7 ± 0.3 | 62.5 ± 1.4 | 85.7 ± 0.0 |
| Acc. Difference | **-32.1** | -0.7 | -5.5 | +0.5 | **-64.2** | -8.3 | -6.3 | -0.3 |

To test this hypothesis, we study how well adapters trained via ERM on datasets *with only* majority groups transfer to minority groups. We subsample training sets such that 100% of all samples belong to majority groups (or the source group in BREEDS Living-17), such that models never see the other groups during training. We evaluate these models on the regular test sets, and report average and worst-group accuracy. If models still obtain high worst-group accuracy, then this suggests that there is little group shift between majority and minority groups. The datasets are thus "easier" from a group-robustness standpoint.

In Table 18, we compare results training ERM adapters on the subsampled datasets and the original splits (all trained on CLIP ResNet-50 pretrained embeddings). Notably, on CIFAR-10.02, despite never seeing certain groups during training, worst-group accuracy only drops 6.3 pp, and is still significantly higher than chance (62.5% vs. 10.0% given 10 possible classes). This suggests there is little group-shift, as training on only the majority groups still enables sufficient transfer to minority groups. Meanwhile, on Waterbirds, CelebA, and BREEDS Living-17, we see much larger drops in performance. This suggests these datasets exhibit more significant distribution shifts between groups.

#### E.6.2 Balanced groups in BREEDS Living-17

Next, for BREEDS Living-17, we hypothesize that the natural group balance in the training set allows ERM adapters to achieve best or competitive worst-group and average accuracy. Recall that the same groups are encountered during train and test (for all splits we combine `source` and `target` splits from the original benchmark [67]). This amounts to roughly evenly sized groups (c.f. Table 7). Thus if there are no actual minority groups in the training data, then training adapters with ERM to minimize the average empirical error across samples can also lead to good generalization across all groups at test time. We note that if the embeddings actually do contain the necessary information to classify each group, then this can still happen despite the initial pretrained embeddings performing poorly (*e.g.*, by learning nonlinear transformations with the adapters). Meanwhile, the sampling procedure with contrastive adapting may restrict the training sample size or focus on certain groups, leading to poorer overall generalization.

To test this hypothesis, we introduce group *imbalance* by subsampling certain groups in the training data. We downsample the training data such that there exist groups that only makes up 5% of the samples in their classes. If group balance is a key factor for the robust performance of ERM adapters on BREEDS Living-17, then we expect the worst-group accuracy for ERM adapters to drop substantially. By introducing group imbalance, but keeping the group shifts, we arguably also make group-robust classification more difficult on the BREEDS Living-17 dataset.

In Table 19, using CLIP ResNet-50 and CLIP ViT-L/14 pretrained embeddings, we evaluate ERM adapters, linear probes, and contrastive adapters on the group-imbalanced BREEDS Living-17 training set, and compare their worst-group accuracy to that achieved after training on the default group-balanced training set. As hypothesized, we find that ERM adapters (and linear probes) perform significantly worse after training with the group-imbalanced data. Meanwhile, the contrastive adapters are more robust to this setting. They now outperform the other ERM approaches by 4.0 pp and 2.0 pp on worst-group accuracy for CLIP ResNet-50 and ViT-L/14 models respectively. In contrast to the results in Table 3, contrastive adapting can outperform ERM training in settings with multiple

Table 19: Evaluating whether group balanced training data explains ERM training performance. On the **BREEDS Living-17** dataset, we compare worst-group accuracy for ERM linear probes (ERM LP), ERM adapters, and contrastive adapters (Contrast. Adapter) after training on the default dataset (Group Balanced) and a harder group-imbalanced version (Group Imbalanced). Compared to ERM, contrastive adapting is more robust to group-imbalanced training data, with lowest drop in performance. Contrastive adapting also outperforms ERM in the harder group-imbalanced setting.

| | CLIP RN-50 | | | CLIP ViT-L/14 | | |
|---|---|---|---|---|---|---|
| Worst-group Acc. (%) | ERM LP | ERM Adapter | Contrast. Adapter | ERM LP | ERM Adapter | Contrast. Adapter |
| Group Balanced | 53.3 ± 0.9 | **70.7 ± 0.9** | 62.0 ± 1.6 | **84.0 ± 0.9** | 82.8 ± 0.9 | 80.0 ± 1.6 |
| Group Imbalanced | 8.0 ± 0.0 | 56.0 ± 1.6 | **60.0 ± 2.3** | 52.7 ± 1.9 | 70.7 ± 0.9 | **72.7 ± 1.9** |
| Acc. Difference | -45.3 | -14.7 | **-2.0** | -31.3 | -12.1 | **-7.3** |

classes and subclass group shift. Notably, from Appendix E.6.2, we saw that the BREEDS Living-17 dataset exhibits significant group shifts, and this improvement over ERM occurs when we make the BREEDS Living-17 dataset harder with group-imbalanced training data. This is also consistent with the improved performance of contrastive adapting over ERM on Waterbirds and CelebA, which exhibit significant group shift (c.f. Table 18) and group imbalance (c.f. Table 7). Thus, contrastive adapting may enable group robust classification in harder settings where ERM training is insufficient.

# F Limitations and societal impact

**Method limitations.** While in this work, we demonstrated that we can substantially improve the group robustness of foundation model classification without any finetuning of the original model, several limitations still exist. First, this does not imply that we can get desirable performance in general without additional retraining. To obtain high worst-group performance in general, we are upper-bounded by whether the pretrained embeddings do contain the information needed to classify all groups. While our study suggests that in many cases they do carry this information—which can surprising given that the zero-shot classification with the same embeddings results in poor group robustness—in other situations the pretrained embeddings may lack this information. For example, if downstream task data is very different in distribution from the pretraining data, then the pretrained foundation model embeddings may not be sufficient to work with. While more efficient ways to improve robustness can democratize foundation model use, further finetuning may still be needed.

We also emphasize that our approach is a simple first-step method to improving the group robustness over existing baseline approaches. This is motivated by our observation that foundation model zero-shot classification may not be group robust, and that we would like to both (i) improve performance of these models when we realize they fail in certain aspects, and (ii) do so efficiently, such that fixing their failures is not bound by who can conduct costly retraining procedures, and when they can do so. We are excited for future work and expect further improvements as this thread of how to efficiently improve FM performance with limited access (*e.g.*, only pretrained embeddings) is further explored.

**Societal impact and related limitations.** Finally, we note that it is important to carefully study the learned biases of foundation models, and to devise appropriate solutions evaluated outside of just computational metrics. Due to their promise of widespread and effective downstream transfer, foundation models may have a particularly strong impact on various parts of society. Individuals may get the sense that they can successfully apply these pretrained models to their desired downstream tasks "out-of-the-box". However, doing so also risks applying any learned biases of the model. Our work raises this issue with respect to group robustness as motivation for our problem setting, also noting that additional evaluation beyond average accuracy can shed light on the negative qualities of existing models (*e.g.*, zero-shot FM classification may perform very well on average, but very poorly on certain groups, c.f. Figure 6). However we note the limitations of purely computational solutions to addressing group performance disparities in society. We also note the need to better understand these large pretrained models and their potential uses in broader socio-technical systems [9].

Table 20: Class prompt templates tried for each dataset. Each vertical section denotes prompt templates used for every dataset in that section, *e.g.*, Waterbirds, CelebA, BREEDS, and CIFAR-10 datasets used the same prompt templates. For Amazon-WILDS and CivilComments-WILDS, we tried different ways to convey the class (*e.g.*, negative versus positive review for Amazon-WILDS), and list the prompts used for the "negative review" and "not toxic" classes respectively.

| Dataset | Prompt template (for a class) |
|---|---|
| Waterbirds | "A photo of a {}." |
| CelebA | "A picture of a {}." |
| BREEDS Living-17 | "A {}" |
| BREEDS Nonliving-26 | "This is a photo of a {}." |
| CIFAR-10.001 | "This is a picture of a {}." |
| CIFAR-10.02 | "a {}" |
| FMoW-WILDS | "satellite imagery of a {}." |
| | "aerial imagery of a {}." |
| | "satellite photo of a {}." |
| | "aerial photo of a {}." |
| | "satellite view of a {}." |
| | "aerial view of a {}." |
| | "satellite imagery of the {}." |
| | "aerial imagery of the {}." |
| | "satellite photo of the {}." |
| | "aerial photo of the {}." |
| | "satellite view of the {}." |
| | "aerial view of the {}." |
| Amazon-WILDS | "negative" |
| | "negative review" |
| | "negative." |
| | "negative review." |
| | "Negative" |
| | "Negative review" |
| | "Negative." |
| | "Negative review." |
| | "This is a negative review" |
| | "This is a negative review." |
| | "this is a negative review" |
| CivilComments-WILDS | "This comment is not toxic" |
| | "This comment is not toxic." |
| | "this comment is not toxic" |
| | "this comment is not toxic." |
| | "This is not a toxic comment." |
| | "this is not a toxic comment." |
| | "this is not a toxic comment" |
| | "Not toxic." |
| | "Not toxic." |
| | "Not toxic" |
| | "not toxic" |
| | "not toxic." |
| | "positive" |