IDENTIFYING SPURIOUS BIASES EARLY IN TRAINING THROUGH THE LENS OF SIMPLICITY BIAS

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural networks trained with (stochastic) gradient descent have an inductive bias towards learning simpler solutions. This makes them highly prone to learning simple *spurious* features that are highly correlated with a label instead of the predictive but more complex core features. In this work, we show that, interestingly, the simplicity bias of gradient descent, can be leveraged to identify spurious correlations early in training. We provide theoretical insights on a two-layer neural network that subsets of data points where the spurious features strongly influence the label predictions are separable based on the model's output in the initial training iterations. We further show that if spurious features have a small enough noise-to-signal ratio, the network's output on the majority of examples containing the spurious feature will be almost exclusively determined by the spurious features and will be nearly invariant to the core feature, leading to poor generalization performance for minority groups. Building on these findings, we propose SPARE, which separates groups with spurious features early in training, and utilizes importance sampling to alleviate the spurious correlation by balancing the group sizes. Through rigorous experiments, we first establish SPARE's effectiveness in discovering spurious correlations in Restricted ImageNet dataset. We then demonstrate that SPARE outperforms state-of-the-art methods by up to 5.6% in worst-group accuracy, while being up to 12x faster.

1 INTRODUCTION

The *simplicity bias* of gradient-based training algorithms towards learning simpler solutions has been suggested as a key factor for the superior generalization performance of overparameterized neural networks (Hermann & Lampinen, 2020; Hu et al., 2020; Nakkiran et al., 2019; Neyshabur et al., 2014; Pezeshki et al., 2021; Shah et al., 2020). At the same time, it is conjectured to make neural networks vulnerable to learning *spurious* correlations frequently found in real-world datasets (Sagawa et al., 2019; Sohoni et al., 2020). Neural networks trained with gradient-based methods can exclusively rely on simple spurious features that exist in majority of examples in a class but are not predictive of the class in general (e.g., image background), and remain invariant to the predictive but more complex core features (Shah et al., 2020). This results in learning non-robust solutions that do not generalize well on minority groups of the original data distribution that do not contain the spurious features (Shah et al., 2020; Teney et al., 2022).

To alleviate spurious biases without knowing the group labels, existing methods partition examples in each class into majority and minority groups. This is done by training the model via gradient descent and flagging the minority based on misclassification (Liu et al., 2021), high loss (Nam et al., 2020), or sensitive representations (Creager et al., 2021; Sohoni et al., 2020). A robust model is then trained by upweighting (Sagawa et al., 2019) or upsampling (Liu et al., 2021) the minority to counteract the majority's spurious features. However, existing methods heavily rely on extensive hyperparameter tuning using a group-labeled validation set for group inference, or require to directly train with a group-labeled validation data, which may not be available for real-world datasets. Besides, state-of-the-art methods are computationally expensive for either group inference (Sohoni et al., 2020) or robust training (Taghanaki et al., 2021) or both (Liu et al., 2021; Nam et al., 2021; Zhang et al., 2022) (as evidenced by Table 2), rendering them impractical for even medium-sized datasets.

In this work, we show that the simplicity bias of gradient descent, which leads to learning spurious biases, can be leveraged to provably separate majority and minority groups *early in training*. To the best of our knowledge, this is the first analysis of how the simplicity bias of SGD encourages learning

of easy spurious correlations and inhibits learning of more complex core features. We examine a two-layer fully connected neural network and identify two early training phases. Initially, the spurious feature's contribution to the model's output within a majority group rises linearly with the spurious correlation. Afterward, if the noise-to-signal ratio of a spurious feature is lower than that of the core feature, the model's output for most examples in the class becomes almost solely determined by the spurious feature. We show that the model's output *provably* separates majority and minority groups early in training. Based on these insights, we introduce a method, SPARE (SePArate early and REsample), that clusters model output early in training to accurately identify examples with spurious features, and uses importance sampling to balance the groups to effectively mitigate spurious bias without increasing training time. In contrast to existing methods, our theoretically-grounded method does not require extensive hyperparameter tuning and thus can operate effectively even without a group-labeled validation set. Moreover, SPARE is very lightweight and easily scales to large datasets.

We first apply SPARE to Restricted ImageNet, a setting not studied previously in the group inference literature, to discover spurious correlations in more realistic settings beyond carefully curated benchmark datasets: SPARE identifies up to 7.3% more examples with spurious correlations than the state-of-the-art group inference methods and improves model's accuracy on minority examples by 11.5%, i.e., up to 23.2% higher than the state-of-the-art methods. Then, we confirm that SPARE can achieve up to 5.6% higher worst-group accuracy compared to state-of-the-art baselines on multiple most commonly used benchmark datasets, including CMNIST (Alain et al., 2015), Waterbirds (Sagawa et al., 2019), and CelebA (Liu et al., 2015) while being up to 12x faster. Notably, SPARE performs comparably or even superior to methods requiring ground-truth group information at training time.

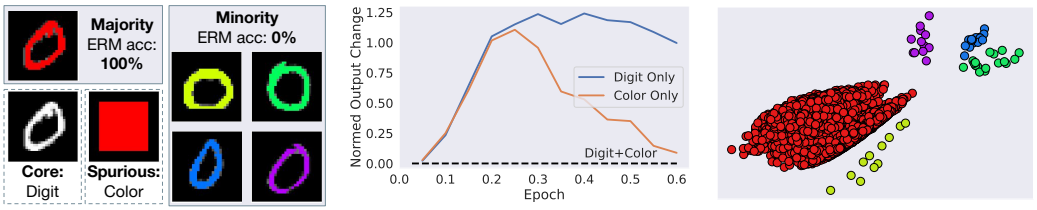
2 RELATED WORK

Mitigating spurious bias. If group labels are available at training time, techniques such as class balancing (He & Garcia, 2009; Cui et al., 2019) and importance weighting (Shimodaira, 2000; Byrd & Lipton, 2019) are used to enhance performance on minority groups. Alternatively, GDRO (Sagawa et al., 2019) focuses on higher-loss groups to minimize the worst group-level error.

Without group labels, existing methods aim to first infer this information for a second round of model training. GEORGE (Sohoni et al., 2020) uses clustering of ERM (Empirical Risk Minimization) representations and then trains the second model with GDRO. LfF (Nam et al., 2020) trains two models simultaneously; the second model upweights examples with high loss from the first. JTT (Liu et al., 2021) and CNC (Zhang et al., 2022) upsample minority groups identified as those misclassified by an initial ERM model and upsample the minority groups. JTT trains the second robust model using ERM, and CNC applies contrastive learning to pull misclassified examples towards their class. EIIL (Creager et al., 2021) and PGI (Ahmed et al., 2020) also split data based on an ERM model by finding an assignment that maximizes the Invariant Risk Minimization (IRM) objective (Arjovsky & Bottou, 2017), i.e., the variance of the model on the two groups. Then, EIIL trains the second robust model with GDRO, and PGI minimizes the KL divergence of softmaxed logits for same-class samples across groups. CIM (Taghanaki et al., 2021) learns input-space transformations of the data to ensure that the transformation preserves task-relevant information. If some group-labeled data is available, SSA (Nam et al., 2021) applies semi-supervised learning with extra group-labeled data to infer the training group labels and then uses GDRO to train a robust model. DFR (Kirichenko et al., 2023) first trains the model with ERM, and then retrains the last layer on the group-balanced data.

State-of-the-art methods heavily rely on an extra group-labeled data (Nam et al., 2021; Kirichenko et al., 2023) to tune their group inference method in a wider range of hyperparameters, or require to directly train on a group-labeled data. Besides, they often significantly increase training time during group-inference or robust training (Liu et al., 2021; Nam et al., 2021; Zhang et al., 2022). In contrast, SPARE can *provably* and accurately separate groups of examples with spurious features *early* in training, thus does not require extensive hyperparameter tuning, and yields a superior performance on minority groups without increasing the training time.

Simplicity Bias. Recent work has revealed the simplicity bias in (stochastic) gradient methods towards learning linear functions early in training, progressing to more complex functions later (Hermann & Lampinen, 2020; Hu et al., 2020; Nakkiran et al., 2019; Neyshabur et al., 2014; Pezeshki et al., 2021; Shah et al., 2020). This is empirically observed in various network architectures, including MobileNetV2, ResNet50, and DenseNet121 (Sandler et al., 2018; He et al., 2016; Shah et al., 2020). Hu et al. (2020) formally proved that initial learning dynamics of a two-layer FC



(a) (b) (c)

Figure 1: Training LeNet-5 on Colored MNIST containing colored handwritten digits. (a) Each digit is a class; the majority of digits in a class have a particular color, and the remaining digits are in 4 other colors. Models trained with ERM learn spurious features (colors) that exist in the majority of examples in a class instead of the core feature (digits) and thus do not perform well on the minority. (b) The network output is almost exclusively indicated by the color of the majority group, early in training. That is, the color alone results in the same prediction as that of the entire image around 0.6 epoch into training, while the digit alone yields a very different prediction. (c) Majority and minority groups are separable based on the network output. Figure 1b in Appendix shows similar results on Waterbirds.

network can be mimicked by a linear model and extended this to multi-layer FC and convolutional networks. Simplicity bias is suggested to explain the good generalization of overparameterized networks but is also *conjectured* to produce models that rely on the simplest feature at the expense of more complex ones, even when the simplest feature has less predictive power (Shah et al., 2020; Teney et al., 2022). However, the exact notion of the simplicity of features and the mechanism by which they are learned remain poorly understood except in certain simplistic settings (Nagarajan et al., 2020; Shah et al., 2020). Here, we build on Hu et al. (2020) and rigorously specify the required conditions and mechanism of learning spurious features by a two-layer FC network.

3 Problem Formulation

Let $\mathcal{D}=\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n \subset \mathbb{R}^d \times \mathbb{R}$ be n training data with features $\boldsymbol{x}_i \in \mathbb{R}^d$, and labels $y_i \in \mathcal{C}=\{1,-1\}$.

Features & Groups. We assume every class $c \in \mathcal{C}$ has a *core* feature $\boldsymbol{v}_c$, which is the invariant feature of the class that appears in both training and test set. Besides, there is a set of *spurious* features $\boldsymbol{v}_s \in \mathcal{A}$ that are shared between classes but may not be present at test time. For example, in the CMNIST dataset containing images of colored hand-written digits (Figure 1a), the digit is the core feature, and its color is the spurious feature. Assuming w.l.o.g. that all $\boldsymbol{v}_c, \boldsymbol{v}_s \in \mathbb{R}^d$ are orthogonal vectors, the feature vector of every example $\boldsymbol{x}_i$ in class c can be written as $\boldsymbol{x}_i = \boldsymbol{v}_c + \boldsymbol{v}_s + \boldsymbol{\xi}_i$, where $\boldsymbol{v}_s \in \mathcal{A}$, and each $\boldsymbol{\xi}_i$ is a noise vector drawn i.i.d. from $\mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_\xi)$. We assume the noise along each feature is independent, and denoted by σ_c^2, σ_s^2 variance of the noise in the directions of $\boldsymbol{v}_c$, $\boldsymbol{v}_s$, respectively. Training examples can be partitioned into groups $g_{c,s}$ based on the combinations of their core and spurious features $(\boldsymbol{v}_c, \boldsymbol{v}_s)$. If a group $g_{c,s}$ contains the majority of examples in class c, it is called a majority group. A class may contain multiple minority groups, corresponding to different spurious features.

Neural Network & Training. We consider a two-layer FC neural network with m hidden neurons:

$$f(\boldsymbol{x}; \boldsymbol{W}, \boldsymbol{z}) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} z_r \phi(\boldsymbol{w}_r^T \boldsymbol{x}/\sqrt{d}) = \frac{1}{\sqrt{m}} \boldsymbol{z}^T \phi(\boldsymbol{W}\boldsymbol{x}/\sqrt{d}), \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^d$ is the input, $\boldsymbol{W} = [\boldsymbol{w}_1, \cdots, \boldsymbol{w}_m]^T \in \mathbb{R}^{m \times d}$ is the weight matrix in the first layer, and $\boldsymbol{z} = [\boldsymbol{z}_1, \cdots, \boldsymbol{z}_m]^T \in \mathbb{R}^m$ is the weight vector in the second layer. Here $\phi : \mathbb{R} \to \mathbb{R}$ is a smooth or piece-wise linear activation function (including ReLU, Leaky ReLU, Erf, Tanh, Sigmoid, Softplus, etc.) that acts entry-wise on vectors or matrices. We consider the following ℓ_2 training loss:

$$\mathcal{L}(\boldsymbol{W}, \boldsymbol{z}) = \frac{1}{2n} \sum_{i=1}^{n} (f(\boldsymbol{x}_i; \boldsymbol{W}, \boldsymbol{z}) - y_i)^2. \tag{2}$$

We train the network by applying gradient descent on the loss (2) starting from random initialization[1]:

$$\boldsymbol{W}_{t+1} = \boldsymbol{W}_t - \eta \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}_t, \boldsymbol{z}_t), \qquad \boldsymbol{z}_{t+1} = \boldsymbol{z}_t - \eta \nabla_{\boldsymbol{z}} \mathcal{L}(\boldsymbol{W}_t, \boldsymbol{z}_t), \tag{3}$$

Worst-group error. We quantify the performance of the model based on its highest test error across groups $\mathcal{G} = \{g_{c,s}\}_{c,s}$ in all classes. Formally, *worst-group* test error is defined as:

$$\text{Err}_{wg} = \max_{g \in \mathcal{G}} \mathop{\mathbb{E}}_{(\boldsymbol{x}_i, y_i) \in g} [y_i \neq y_f(\boldsymbol{x}_i; \boldsymbol{W}, \boldsymbol{z})], \tag{4}$$

[1]Detailed assumptions on the activations, and initialization can be found in Appendix A.2

where $y_f(\boldsymbol{x}_i; \boldsymbol{W}, \boldsymbol{z})$ is the label predicted by the model. In other words, Err_{wg} measures the highest fraction of examples that are incorrectly classified across all groups.

While for simplicity, we consider binary classification with ℓ_2 loss, our analysis generalizes to multi-class classification with CE loss, and other model architectures, as we also confirm experimentally.

4 Investigating Spurious Feature Learning in Neural Networks

We start by investigating how spurious features are learned during training a two-layer fully-connected neural network. Our analysis reveals two phases in early-time learning. First, in the initial training iterations, the contribution of a spurious feature to the network output increases linearly with the amount of the spurious correlation. Interestingly, if the majority group is sufficiently large, majority and minority groups are separable at this phase by the network output. Second, if the noise-to-signal ratio of the spurious feature of the majority group is smaller than that of the core feature, the network's output on the majority of examples in the class will be almost exclusively determined by the spurious feature and will remain mostly invariant to the core feature. Next, we will discuss the two phases in detail.

4.1 Spurious Features are Learned in the Initial Training Iterations

We start by analyzing the effect of spurious features on the learning dynamics of a two-layer FC neural network trained with gradient descent in the initial training iterations. The following theorem shows that if a majority group is sufficiently large, the contribution of the spurious feature of the majority group to the model's output is magnified by the network at every step early in training.

Theorem 4.1. *Let $\alpha \in (0, \frac{1}{4})$ be a fixed constant. Suppose the number of training samples n and the network width m satisfy $n \gtrsim d^{1+\alpha}$ and $m \gtrsim d^{1+\alpha}$. Let n_c be the number of examples in class c, and $n_{c,s} = |g_{c,s}|$ be the size of group $g_{c,s}$ with label c and spurious feature $\boldsymbol{v}_s \in \mathcal{A}$. Then, under the setting of Sec. 3 there exist a constant $\nu_1 > 0$, such that with high probability, for all $0 \le t \le \nu_1 \cdot \sqrt{\frac{d^{1-\alpha}}{\eta}}$, the contribution of the core and spurious features to the network output can be quantified as follows:*

$$f(\boldsymbol{v}_c; \boldsymbol{W}_t, \boldsymbol{z}_t) = \sqrt{\frac{2}{d}} \eta \zeta c \|\boldsymbol{v}_c\|^2 t \left(\frac{n_c}{n} \pm \mathcal{O}(d^{-\Omega(\alpha)}) \right), \tag{5}$$

$$f(\boldsymbol{v}_s; \boldsymbol{W}_t, \boldsymbol{z}_t) = \sqrt{\frac{2}{d}} \eta \zeta c \|\boldsymbol{v}_s\|^2 t \left(\frac{n_{c,s} - n_{c',s}}{n} \pm \mathcal{O}(d^{-\Omega(\alpha)}) \right), \tag{6}$$

where $c' = \mathcal{C} \setminus c$, and ζ is the expected gradient of activation functions at random initialization.

The proof can be found in Appendix B.2. Note that the width requirement in Theorem 4.1 is very mild as it only requires to be larger than $d^{1+\alpha}$ for some small constant α, but can be much smaller than the number of samples. The proof of Theorem 4.1 builds on the bound on the difference between training dynamics of a two-layer fully-connected neural network trained with gradient descent and that of a linear model (Hu et al., 2020) early in training, with a modest generalization that this bound holds for isolated core and spurious features, as we justify in Appendix A.1. At a high level, as the model is nearly linear in the initial $\nu_1 \cdot \frac{d \log d}{\eta}$ iterations, the contribution of the spurious feature $\boldsymbol{v}_s$ to the network output grows almost linearly with $(n_{c,s} - n_{c',s}) \|\boldsymbol{v}_s\|^2$, at *every iteration* in the initial phase of training. Note that $n_{c,s} - n_{c',s}$ is the correlation between the spurious feature and the label c. When $n_{c,s} \gg n_{c',s}$, the spurious feature exists almost exclusively in the majority group of class c, and thus has a high correlation only with class c. In this case, if the magnitude of the spurious feature is significant, the contribution of the spurious feature to the model's output grows very rapidly, early in training. In particular, if $(n_{c,s} - n_{c',s}) \|\boldsymbol{v}_s\|^2 \gg n_c \|\boldsymbol{v}_c\|^2$, the model's output is increasingly determined by the spurious feature, but not the core feature.

Remember from Sec. 3 that every example consists of a core and a spurious feature. As the effect of spurious features of the majority groups is amplified in the network output, the model's output will differ for examples in the majority and minority groups. The following corollary shows that the majority and minority groups are separable based on the network's output early in training. Notably, multiple minority groups with spurious features contained in majority groups of other classes are also separable.

Corollary 4.2 (Separability of majority and minority groups). *Suppose that for all classes, a majority group has at least K examples and a minority group has at most k examples. Then, under*

the assumptions of Theorem 4.1, examples in the majority and minority groups are separable based on the model's output, early in training. That is, for all $0 \leq t \leq \nu_1 \cdot \sqrt{\frac{d^{1-\alpha}}{\eta}}$*, with high probability, the following holds for at least* $1 - \mathcal{O}(d^{-\Omega(\alpha)})$ *fraction of the training examples* $\boldsymbol{x}_i$ *in group* $g_{c,s}$*:*

If $g_{c,s}$ *is in a majority group in class* $c = 1$*:*

$$f(\boldsymbol{x}_i; \boldsymbol{W}_t, \boldsymbol{z}_t) \geq \frac{2\eta\zeta^2 t}{d} \left(\frac{\|\boldsymbol{v}_s\|^2 (K-k)}{n} + \xi \pm O(d^{-\Omega(\alpha)}) \right) + \rho(t, \phi, \Sigma), \tag{7}$$

If $g_{c,s}$ *is in a minority group in class* $c = 1$*, but* $g_{c',s}$ *is a majority group in class* $c' = -1$*:*

$$f(\boldsymbol{x}_i; \boldsymbol{W}_t, \boldsymbol{z}_t) \leq \frac{2\eta\zeta^2 t}{d} \left(-\frac{\|\boldsymbol{v}_s\|^2 (K-k)}{n} + \xi \pm O(d^{-\Omega(\alpha)}) \right) + \rho(t, \phi, \Sigma), \tag{8}$$

where ρ *is constant for all examples in the same class,* $\xi \sim \mathcal{N}(0, \kappa)$ *with* $\kappa = \frac{1}{n}(\sum_c n_c^2 \sigma_c^2 \|\boldsymbol{v}_c\|^2)^{1/2} + \frac{1}{n}(\sum_s (n_{c,s} - n_{c',s})^2 \sigma_s^2 \|\boldsymbol{v}_s\|^2)^{1/2}$ *is the total effect of noise on the model.*

Analogous statements holds for the class $c = -1$ *by changing the sign and direction of the inequality.*

The proof can be found in Appendix B.2. Corollary 4.2 shows that when the majority group is considerably larger than the minority groups ($K \gg k$), the prediction of examples in the majority group move toward their label considerably faster, due to the contribution of the spurious feature. Hence, majority and minority groups can be separated from each other, early in training. Importantly, multiple minority groups can be also separated from each other, if their spurious feature exists in majority groups of other classes. Note that $K > k + |\xi|$ is the minimum requirement for the separation to happen. Separation is more significant when $K \gg k$ and when $\|\boldsymbol{v}_s\|$ is significant.

4.2 Network Relies on Simple Spurious Features for Majority of Examples

Next, we analyze the second phase in early-time learning of a two-layer neural network. In particular, we show that if the noise-to-signal ratio of the spurious feature of the majority group of class c, i.e., $R_s = \sigma_s / \|\boldsymbol{v}_s\|$ is smaller than that of the core feature $R_c = \sigma_c / \|\boldsymbol{v}_c\|$, then the neural network's output is almost exclusively determined by the spurious feature and remain invariant to the core feature at $T = \nu_2 \cdot \frac{d \log d}{\eta}$, even though the core feature is more predictive of the class.

Theorem 4.3. *Under the assumptions of Theorem 4.1, if the classes are balanced, and the total size of the minority groups in class* c *is small, i.e.,* $\mathcal{O}(n^{1-\gamma})$ *for some* $\gamma > 0$*, then there exists a constant* $\nu_2 > 0$ *such that at* $T = \nu_2 \cdot \frac{d \log d}{\eta}$*, for an example* $\boldsymbol{x}_i$ *in a majority group* $g_{c,s}$*, the contribution of the core feature to the model's output is at most:*

$$|f(\boldsymbol{v}_c; \boldsymbol{W}_T, \boldsymbol{z}_T)| \leq \sqrt{d} \frac{R_s}{\zeta R_c} + \mathcal{O}(n^{-\gamma}\sqrt{d}) + \mathcal{O}(d^{-\Omega(\alpha)}). \tag{9}$$

In particular if $\min\{R_c, 1\} \gg R_s$*, then the model's output is mostly indicated by the spurious feature instead of the core feature:*

$$|f(\boldsymbol{v}_s; \boldsymbol{W}_T, \boldsymbol{z}_T)| \geq \frac{\sqrt{d}}{2\zeta} \gg |f(\boldsymbol{v}_c; \boldsymbol{W}_T, \boldsymbol{z}_T)|. \tag{10}$$

The proof can be found in Appendix B.3. The proof of Theorem 4.3 shows that at $T = \nu_2 \cdot \frac{d \log d}{\eta}$ where the linear model that closely mimics early-time learning dynamics of a two-layer FC neural network converges to its optimum parameters, the network has fully learned the spurious feature of the majority groups. At the same time, the contribution of the core feature to the network's output is at most proportional to R_s / R_c. Hence, if $R_s \ll R_c$, the core feature does not considerably contribute to the output of the neural network at T. That is, the network almost exclusively relies on the spurious feature of the majority group instead of the core feature which is more predictive of the class.

We note that our results in Theorem 4.1, Corollary 4.2, and Theorem 4.3 generalize to more than two classes and hold if the classes are imbalanced, as we will confirm by our experiments. Similar results can be shown for multi-layer fully connected and convolutional networks, following (Hu et al., 2020).

Empirical Evidence of Theoretical Results. In Figure 1, we empirically illustrate the above results during early-time training of LeNet-5 (LeCun et al., 1998) on the Colored MNIST (Alain et al., 2015) dataset containing colored handwritten digits. Figure 1b shows that the prediction of the network on the majority group is almost exclusively indicated by the color of the majority group, confirming Theorem 4.3. An analogous result is shown on the Waterbirds dataset in Figure 6 in the Appendix. Figure 1c shows that the majority and minority groups are separable based on the network output, confirming Corollary 4.2.

Finally, note that by only learning the spurious feature, the neural network can shrinks the training loss on the majority of examples in class c to nearly zero and correctly classify them. Hence, the contribution of the spurious feature of the majority group of class c to the model's output is retained throughout the training. On the other hand, if minority groups are small, higher complexity functions that appear later in training overfit the minority groups, as observed by (Sagawa et al., 2020). This results in a small training error but a poor worst-group generalization performance on the minorities.

5 SPARE: ELIMINATING SPURIOUS BIAS EARLY IN TRAINING

Drawing on the theoretical foundations outlined in Section 4, we develop a principled pipeline, SPARE, to discover and mitigate spurious correlations *early in training*.

Discovering Spurious Correlations: Separating the Groups Early in Training. Corollary 4.2 shows that majority and minority groups are separable based on the network's output. To identify the majority and minority groups, we cluster examples V_c in every class $c \in \mathcal{C}$ based on the output of the network, during the first few epochs. We determine the number of clusters via silhouette analysis (Rousseeuw, 1987). In doing so, we can separate majority and minority groups in each class of examples with different spurious features. Any clustering algorithm such as k-means or k-median clustering can be applied to separate the groups.

Algorithm 1 SePArate early and REsample (SPARE)

Input: Network $f(., \boldsymbol{W})$, data $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, loss function $\mathcal{L}$, iteration numbers T_N, T_{init}.
Output: Model f trained without bias
Stage 1: Early Bias Identification
for $t = 0, \cdots, T_{init}$ **do**
  $\boldsymbol{W}_{t+1} \leftarrow \boldsymbol{W}_t - \eta \nabla \mathcal{L}(\boldsymbol{W}_t; \mathcal{D})$
end for
for every class $c \in \mathcal{C}$ with examples V_c **do**
  Identify λ, # of clusters k via Silhouette analysis
  Cluster V_c into $\{V_{c,j}\}_{j=1}^k$ based on $f(\boldsymbol{x}_i; \boldsymbol{W}_t)$
  Weight every $\boldsymbol{x}_i \in V_{c,j}$ by $w_i = 1/|V_{c,j}|$, $p_i = w_i^\lambda / \sum_i w_i^\lambda$
end for
Stage 2: Learning without Bias
for $t = 0, \cdots, T_N$ **do**
  Sample a mini-batch $\mathcal{M}_t = \{(\boldsymbol{x}_i, y_i)\}_i$ with probabilities p_i
  $\boldsymbol{W}_{t+1} = \boldsymbol{W}_t - \eta \nabla \mathcal{L}(\boldsymbol{W}_t; \mathcal{M}_t)$.
end for

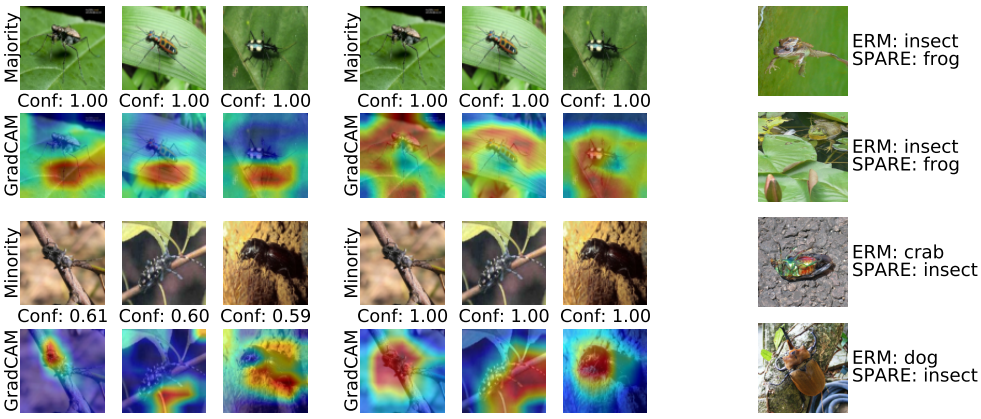
Mitigation after Discovery: Balancing Groups via Importance Sampling. To alleviate the spurious correlations and enable effective learning of the core features, we employ an importance sampling method on examples in each class to upsample examples in the smaller clusters and downsample examples in the larger clusters. To do so, we assign every example $i \in V_{c,j}$ a weight given by the size of the cluster it belongs to, i.e., $w_i = 1/|V_{c,j}|$. Then we sample examples in every mini-batch with probabilities equal to $p_i = w_i^\lambda / \sum_i w_i^\lambda$, where λ can be determined based on the average silhouette score of clusters in each class, *without further tuning*. See Appendix G for a detailed explanation and ablation study on choosing λ. Note that our importance sampling method does not increase the size of the training data, and only changes the data distribution. Hence, it does not increase the training time. The pseudocode is illustrated in Algorithm 1.

6 EXPERIMENTS

In this section, we first demonstrate that SPARE effectively discovers and mitigates naturally existing spurious correlations early in training, on Restricted ImageNet—a realistic dataset *not previously studied for spurious correlations*. Then, we confirm that SPARE outperforms state-of-the-art baselines in inferring and mitigating spurious correlations across multiple curated benchmark datasets.

6.1 DISCOVERING NATURAL SPURIOUS CORRELATIONS IN RESTRICTED IMAGENET

We first show the applicability of SPARE to discover and mitigate spurious correlations with Restricted ImageNet (Taghanaki et al., 2021), a 9-superclass subset of ImageNet, to train ResNet-50 from scratch.



(a) Insects in ImageNet, Epoch 8. (b) Insects in ImageNet, End. (c) SPARE corrects spurious correlation.

Figure 2: SPARE-discovered spurious correlation between "green leaf" & "insect" in Restricted ImageNet. (a) SPARE can separate the majority groups whose predictions are dominated by spurious features early in training. (b) According to the GradCAM, the model's outputs rely heavily on spurious features if present, as suggested by Theorem 4.3. (c) Models trained with SPARE do not learn the spurious correlations that would otherwise learned with ERM.

Table 1: Mitigating spurious correlations in Restricted ImageNet. SPARE infers groups more accurately and improves the model's performance on both minority groups over ERM.

	Test Acc	Insect Minority Acc	Frog Minority Acc	Spurious Recall
ERM	96.0%	91.7%	80.8%	-
CB	95.9%	**93.7%** ↑	80.8%−	-
JTT	92.8%	75.0% ↓	**92.3%** ↑	77.9%
EIIL	93.1%	88.3% ↓	69.2% ↓	75.8%
SPARE	95.4%	**92.9%** ↑	**92.3%** ↑	**83.1%**

We applied SPARE to cluster the model's output every 2 epochs in the first 10 epochs and inspected the clusters as described below. See Appendix F for more details on the dataset and experiment.

SPARE Discovers Spurious Correlations in Insect and Frog Classes Early in Training. By inspecting the clusters with the highest fraction of misclassified examples to another class, we find that many Frog images are misclassified as Insects. Figure 2a shows examples from the two groups SPARE finds for the Insect class at epoch 8, where clusters with spurious feature are visually evident (we visually inspected epochs 4, 6, 8)[2]. GradCAM reveals an obvious spurious correlation between "green leaf" and the insect class that is maintained until the end of the training, as illustrated in Figure 2b. We also observe a large gap between the confidence of examples in the two groups. This indicates that the model has learned the spurious feature early in training.

SPARE Discovers Spurious Correlations Without Reliance on Group-labled Validation. We compare SPARE with state-of-the-art group inference baselines. However, these methods heavily rely on a group-labeled validation set to identify the time of group inference during training with ERM. This covers a wide range from epoch 1 to 60 for Waterbirds and CelebA datasets. While SPARE can also benefit from a group-labeled validation, this is not essential. In fact, our theoretical results limit the range for inference time to the initial epochs. This sets SPARE apart as a more generally applicable method for discovering and mitigating spurious correlations, even in the absence of a validation set.

SPARE Achieves State-of-the-art Accuracy on Minority Groups. Based on the spurious correlations SPARE discovered, we manually labeled the background of both training and test data for the *insect* and *frog* classes. We used these group labels to tune the baseline group inference methods. Table 1 shows SPARE separates the insect majority group with the spurious correlation better than other group inference methods and improves both insect and frog minority accuracy by 1.2% and 11.5% respectively, with only a minor drop in total accuracy. CB only improves insect minority accuracy. JTT decreases the model's accuracy on the insect minority a lot while improving

[2]Since the model is not pretrained, it is expected that the spurious clusters form slightly later. For pretrained models, spurious clusters form very early, as we will confirm in Table 2

Table 2: Worst-group and average accuracy (%) of training with SPARE vs. state-of-the-art algorithms, on datasets with spurious correlations. CB, GB indicate balancing classes and groups, respectively. SPARE achieves a superior performance much faster. The range provided for the training cost encompasses all three datasets, which accounts not only for the number of epochs during which the group identification model was trained, but also for the number of training examples involved in robust training (excluding tuning cost). Baseline results are from (Zhang et al., 2022). ♦ marks using group-labeled validation set for tuning group inference, and △ means the validation set is needed for robust training. (E#) shows the early group inference epoch for SPARE. SPARE is the only method that does not heavily rely on a validation set (◊) or incur additional training costs.

	Group labels required	Train cost	CMNIST		Waterbirds		CelebA	
			Worst-group	Average	Worst-group	Average	Worst-group	Average
ERM	- -	1x	$0.0_{\pm 0.0}$	$20.1_{\pm 0.2}$	$62.6_{\pm 0.3}$	$97.3_{\pm 1.0}$	$47.7_{\pm 2.1}$	$94.9_{\pm 0.3}$
CB	- -	1x	$0.0_{\pm 0.0}$	$23.7_{\pm 3.1}$	$62.8_{\pm 1.6}$	$97.1_{\pm 0.1}$	$46.1_{\pm 1.5}$	$95.2_{\pm 0.4}$
PGI	♦–	1x	$73.5_{\pm 1.8}$	$88.5_{\pm 1.4}$	$79.5_{\pm 1.9}$	$95.5_{\pm 0.8}$	$85.3_{\pm 0.3}$	$87.3_{\pm 0.1}$
EIIL	♦–	1x	$72.8_{\pm 6.8}$	$90.7_{\pm 0.9}$	$83.5_{\pm 2.8}$	$94.2_{\pm 1.3}$	$81.7_{\pm 0.8}$	$85.7_{\pm 0.1}$
GEORGE	- -	2x	$76.4_{\pm 2.3}$	$89.5_{\pm 0.3}$	$76.2_{\pm 2.0}$	$95.7_{\pm 0.5}$	$54.9_{\pm 1.9}$	$94.6_{\pm 0.2}$
LfF	♦△	2x	$0.0_{\pm 0.0}$	$25.0_{\pm 0.5}$	$78.0_{N/A}$	$91.2_{N/A}$	$77.2_{N/A}$	$85.1_{N/A}$
CIM	♦△	2x	$0.0_{\pm 0.0}$	$36.8_{\pm 1.3}$	$77.2_{N/A}$	$95.6_{N/A}$	$83.6_{N/A}$	$90.6_{N/A}$
JTT	♦△	5x-6x	$74.5_{\pm 2.4}$	$90.2_{\pm 0.8}$	$83.1_{\pm 3.5}$	$90.6_{\pm 0.3}$	$81.5_{\pm 1.7}$	$88.1_{\pm 0.3}$
CnC	♦△	2x-12x	$77.4_{\pm 3.0}$	$90.9_{\pm 0.6}$	$88.5_{\pm 0.3}$	$90.9_{\pm 0.1}$	$88.8_{\pm 0.9}$	$89.9_{\pm 0.5}$
SPARE	◊–	**1x**	**(E2) 83.0$_{\pm 1.7}$**	$91.8_{\pm 0.7}$	**(E2) 91.6$_{\pm 0.8}$**	$96.2_{\pm 0.6}$	**(E1) 90.3$_{\pm 0.3}$**	$91.1_{\pm 0.1}$
SSA	validation	1.5x-5x	$0.0_{\pm 0.0}$	$47.9_{\pm 14.4}$	$89.0_{\pm 0.6}$	$92.2_{\pm 0.9}$	$89.8_{\pm 1.3}$	$92.8_{\pm 0.1}$
DFR	training sub.	1x	-	-	$90.4_{\pm 1.5}$	$94.1_{\pm 0.5}$	$80.1_{\pm 1.1}$	$89.7_{\pm 0.4}$
GB	training full	1x	$82.2_{\pm 1.0}$	$91.7_{\pm 0.6}$	$86.3_{\pm 0.3}$	$93.0_{\pm 1.5}$	$85.0_{\pm 1.1}$	$92.7_{\pm 0.1}$
GDRO	training full	1x	$78.5_{\pm 4.5}$	$90.6_{\pm 0.1}$	$89.9_{\pm 0.6}$	$92.0_{\pm 0.6}$	$88.9_{\pm 1.3}$	$93.9_{\pm 0.1}$

the frog minority. EIIL decreases both minority and total accuracy as it finds the least majority. Unlike the baselines, SPARE effectively balances groups, mitigating spurious correlations.

6.2 MITIGATING CURATED SPURIOUS CORRELATIONS IN BENCHMARK DATASETS

Next, we evaluate the effectiveness of SPARE in alleviating spurious correlations on spurious benchmarks. The reported results are averaged over three runs with different model initializations.

Benchmark Datasets & Models. (1) CMNIST (Alain et al., 2015) contains colored handwritten digits derived from MNIST (LeCun et al., 1998). We follow the challenging 5-class setting in Zhang et al. (2022) where every two digits form one class and 99.5% of training examples in each class are spuriously correlated with a distinct color. We use a 5-layer CNN (LeNet-5 (LeCun et al., 1998)) for CMNIST. (2) Waterbirds (Sagawa et al., 2019) contains two classes (landbird vs. waterbird) and the background (land or water) is the spurious feature. Majority groups are (waterbird, water) and (landbird, land). (3) CelebA (Liu et al., 2015) is another most commonly used benchmark for spurious correlations. Following Sagawa et al. (2019), we consider the hair color (blond vs. non-blond) as the class labels and gender (male or female) as the spurious feature. The majority groups are (blond, female) and (non-blond male). For both Waterbirds and CelebA, we follow the standard settings used in the previous work to train a ResNet-50 model (He et al., 2016) pretrained on ImageNet provided by the Pytorch library (Paszke et al., 2019). More details about the datasets and the experimental settings are in Appendix D.

Baselines. We compare SPARE with the state-of-the-art methods for eliminating spurious correlations in Table 2, in terms of both worst-group accuracy, i.e., the minimum accuracy across all groups, and average accuracy. We use adjusted average accuracy for Waterbirds, i.e., the average accuracy over groups weighted by their size. This is consistent with prior work, and is done because the validation and test sets are group-balanced while the training set is skewed. GB (Group Balancing) and GDRO (Sagawa et al., 2019) use the group label of all training examples, and SSA (Nam et al., 2021) uses the group labels of the validation data. DFR (Kirichenko et al., 2023) uses a group-balanced data drawn from training data. The rest of the methods infer the group labels without using such information.

SPARE outperforms SOTA algorithms, including those that require group information. Table 2 shows that compared to baselines that do not use the group labels, SPARE obtains the *highest* worst-group accuracy, while maintaining high average accuracy. In particular, SPARE consistently outperforms the best baselines, CnC (Zhang et al., 2022) and JTT (Liu et al., 2021), on worst-group and average accuracy while having up to 12x lower computational cost (k-means/total wall-clock runtimes are in Appendix Table 7 and Table 8). Notably, SPARE performs comparably to those that use the group information, and even achieves a better worst-group accuracy on CMNIST and CelebA

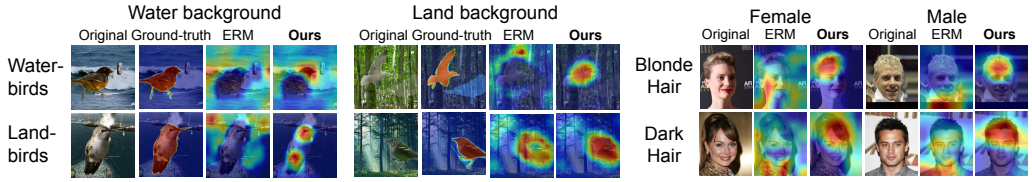


Figure 3: GradCAM Visualization. Warmer colors correspond to the pixels that are weighed more in making the final classification. SPARE allows learning the core features instead of spurious ones.

and has a comparable worst-group but higher average accuracy on the Waterbirds. As group labels are unavailable in real-world datasets, methods that do not rely on group labels are more practical. Among such methods, SPARE has a superior performance and easily scales to large datasets. Notably, SPARE finds the groups, at *epoch 2* for CMNIST and Waterbirds, and at *epoch 1* for CelebA.

Ablation on Two Stages. We conducted an ablation study to examine (1) group inference and (2) robust training with SPARE vs. other techniques. Hyperparameters were tuned similarly for all variants, as detailed in Appendix D.2. Both components of SPARE are essential for its superior performance and using groups found by SPARE usually leads to better performance even when combined with other robust training methods. The second best group inference method, EIIL decides when to find groups by hyperparameter tuning while SPARE finds groups early guided by our theory in Section 4. Ablation study on using the silhouette score to determine λ can be found in Appendix G.

Table 3: Ablation of using different combinations of group inference and robust training methods on the Waterbirds benchmark.

Group inference	**Robust training**	Worst-group	Avg Acc
JTT	SPARE	80.1 ± 1.3	94.6 ± 0.4
GEORGE/CnC	SPARE	84.4 ± 2.2	87.1 ± 1.0
SPARE	JTT	86.2 ± 3.6	92.0 ± 0.8
SPARE	GDRO(/GEORGE/EIIL)	87.6 ± 0.8	89.4 ± 1.3
EIIL	SPARE	88.6 ± 0.1	95.2 ± 0.1
SPARE	SPARE	$\mathbf{91.6 \pm 0.8}$	$\mathbf{96.2 \pm 0.6}$

GradCAM visualizations: SPARE helps the learning of core features. Fig. 3 compares GradCAM (Selvaraju et al., 2017) visualizations depicting saliency maps for samples from Waterbirds with water and land backgrounds (left), and from CelebA with different genders (right), when ResNet50 is trained by ERM vs. SPARE. Warmer colors indicate the pixels that the model considered more important for making the final classification, based on gradient activations. We see that training with SPARE allows the model to learn the core feature, instead of the spurious features.

Noise-to-signal Ratio To study the effect of noise-to-signal ratio on the performance of SPARE (*c.f.* Theorem 4.3), we conduct experiments on a version of CMNIST where we added color patches to the background instead of the digits which allows us to control the noise-to-signal ratio better via the size and locations of the patch. Results in Table 4 and Table 5 in the Appendix demonstrate the effectiveness of SPARE in handling spurious correlations in different scenarios. We see that ERM easily learns a spurious feature with a small variance and/or a large signal and obtains a poor worst-group accuracy. Under large spurious noise, JTT cannot infer the groups well and performs poorly. Besides, EIIL performs poorly when the spurious signal is large. In all cases, SPARE archives state-of-the-art worst-group accuracy, and outperforms the other group-inference methods. Notably, SPARE performs better or comparable to GB and GDRO that use the group labels and thus are not affected by noise-to-signal ratio during group inference.

7 CONCLUSION

In this work, we studied how neural networks trained with gradient methods learn simple spurious features. In particular, we analyzed a two-layer fully-connected neural network and showed that spurious features can be identified early in training based on model output. If these features have a low noise-to-signal ratio, they dominate the network's output, overshadowing core features. Based on the above theoretical insights, we proposed SPARE, which separates majority and minority groups by clustering the model output early in training. Then, it applies importance sampling based on the cluster sizes to make the groups relatively balanced. Importantly, unlike existing group inference methods, SPARE does not require extensive hyperparameter tuning and hence can discover spurious correlations in realistic scenarios like Restricted ImageNet early in training. In also outperforms state-of-the-art methods in worst-group accuracy on benchmark datasets with carefully curated spurious correlations. SPARE is also highly scalable, making it suitable for large-scale applications.

Limitations. To our knowledge, simplicity bias has been mainly studied for vision models, and applicability of our method to other data modalities requires further investigations.

REFERENCES

Faruk Ahmed, Yoshua Bengio, Harm van Seijen, and Aaron Courville. Systematic generalisation with group invariant predictions. In *International Conference on Learning Representations*, 2020.

Guillaume Alain, Alex Lamb, Chinnadhurai Sankar, Aaron Courville, and Yoshua Bengio. Variance reduction in sgd by distributed importance sampling. *arXiv preprint arXiv:1511.06481*, 2015.

Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*, 2017.

Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. Streaming submodular maximization: Massive data summarization on the fly. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 671–680, 2014.

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

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

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*, pp. 9268–9277, 2019.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

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

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*, pp. 770–778, 2016.

Katherine Hermann and Andrew Lampinen. What shapes feature representations? exploring datasets, architectures, and training. *Advances in Neural Information Processing Systems*, 33:9995–10006, 2020.

Wei Hu, Lechao Xiao, Ben Adlam, and Jeffrey Pennington. The surprising simplicity of the early-time learning dynamics of neural networks. *Advances in Neural Information Processing Systems*, 33: 17116–17128, 2020.

Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Zb6c8A-Fghk.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

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*, pp. 6781–6792. PMLR, 2021.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Baharan Mirzasoleiman, Amin Karbasi, Rik Sarkar, and Andreas Krause. Distributed submodular maximization: Identifying representative elements in massive data. In *Advances in Neural Information Processing Systems*, pp. 2049–2057, 2013.

Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. Lazier than lazy greedy. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. *arXiv preprint arXiv:2010.15775*, 2020.

Preetum Nakkiran, Gal Kaplun, Dimitris Kalimeris, Tristan Yang, Benjamin L Edelman, Fred Zhang, and Boaz Barak. Sgd on neural networks learns functions of increasing complexity. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 3496–3506, 2019.

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

Junhyun Nam, Jaehyung Kim, Jaeho Lee, and Jinwoo Shin. Spread spurious attribute: Improving worst-group accuracy with spurious attribute estimation. In *International Conference on Learning Representations*, 2021.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *arXiv preprint arXiv:1412.6614*, 2014.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. 2019.

Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34:1256–1272, 2021.

Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2019.

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

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33:9573–9585, 2020.

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.

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

Saeid A Taghanaki, Kristy Choi, Amir Hosein Khasahmadi, and Anirudh Goyal. Robust representation learning via perceptual similarity metrics. In *International Conference on Machine Learning*, pp. 10043–10053. PMLR, 2021.

Damien Teney, Ehsan Abbasnejad, Simon Lucey, and Anton Van den Hengel. Evading the simplicity bias: Training a diverse set of models discovers solutions with superior ood generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16761–16772, 2022.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SyxAb30cY7.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Laurence A Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393, 1982.

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.

A APPENDIX

A.1 SIMPLICITY BIAS

A recent body of work revealed that the neural network trained with (stochastic) gradient methods can be approximated on the training data by a linear function early in training (Hermann & Lampinen, 2020; Hu et al., 2020; Nakkiran et al., 2019; Neyshabur et al., 2014; Pezeshki et al., 2021; Shah et al., 2020). We hypothesize that a slightly stronger statement holds, namely the approximation still holds if we isolate a core or spurious feature from an example and input it to the model.

Assumption A.1 (simplicity bias on core and spurious features, informal)**.** Suppose that f^{lin} is a linear function that closely approximates $f(\boldsymbol{x}; \boldsymbol{W}, \boldsymbol{z})$ on the training data. Then f^{lin} also approximates f on input either a core feature or a spurious feature corresponding to a majority group in some class, that is

$$f^{lin}(\boldsymbol{v}_c) \approx f(\boldsymbol{v}_c; \boldsymbol{W}, \boldsymbol{z}) \qquad \forall c \in \mathcal{C}$$
$$f^{lin}(\boldsymbol{v}_s) \approx f(\boldsymbol{v}_s; \boldsymbol{W}, \boldsymbol{z}) \qquad \forall s \in \mathcal{A}$$

Intuitively, every core feature and every spurious feature corresponding to a majority group is well represented in the training dataset, and since it is known that the linear model and the full neural network agree on the training dataset, we can expect them to agree on such features as well. Note that spurious features that do not appear in majority groups may not be well represented in the training dataset, hence we do not require that the linear model approximates the neural network well on such features.

Moreover, we verify assumption A.1 empirically on CMNIST in Figure 5, which shows that a two layer neural network and the approximating linear model are close even when isolating a core or spurious feature.

The formal statement is provided below as Assumption A.6.

A.2 SETTING

We now introduce the formal mathematical setting for the theory.

Let $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n \subset \mathbb{R}^d \times \mathbb{R}$, be a dataset with covariance $\boldsymbol{\Sigma}$. Define the data matrix $\boldsymbol{X} = [\boldsymbol{x}_1 \quad \dots \quad \boldsymbol{x}_n]^\top$ and the label vector $\boldsymbol{y} = [y_1 \quad \dots \quad y_n]^\top$. We use $\|\cdot\|$ to refer to the Euclidean norm of a vector or the spectral norm of the data.

Following Hu et al. 2020, we make the following assumptions:

Assumption A.2 (input distribution)**.** The data has the following properties (with high probability):

$$\frac{\|\boldsymbol{x}_i\|^2}{d} = 1 \pm O(\sqrt{\frac{\log n}{d}}), \forall i \in [n]$$
$$\frac{|\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle|}{d} = O(\sqrt{\frac{\log n}{d}}), \forall i, j \in [n], i \neq j$$
$$\|\boldsymbol{X}\boldsymbol{X}^\top\| = \Theta(n)$$

Assumption A.3 (activation function)**.** The activation $\phi(\cdot)$ satisfies either of the following:

- smooth activation: ϕ has bounded first and second derivative
- piecewise linear activation:

$$\phi(z) = \begin{cases} z & z \geq 0 \\ az & z < 0 \end{cases}$$

Assumption A.4 (initialization)**.** The weights $(\boldsymbol{W}, \boldsymbol{v})$ are initialized using symmetric initialization:

$$\boldsymbol{w}_1, \dots, \boldsymbol{w}_{\frac{m}{2}} \sim \mathcal{N}(\boldsymbol{0}_d, \boldsymbol{I}_d), \qquad \boldsymbol{w}_{i+\frac{m}{2}} = \boldsymbol{w}_i (\forall i \in 1, \dots, \frac{m}{2})$$
$$v_1, \dots, v_{\frac{m}{2}} \sim \text{Unif}(\{-1, 1\}), \qquad v_{i+\frac{m}{2}} = -v_i (\forall i \in 1, \dots, \frac{m}{2})$$

It is not hard to check that the concrete scenario we choose in our analysis satisfies the above assumptions. Now, given the following assumptions, we leverage the result of Hu et al. (2020):

Theorem A.5 (Hu et al. 2020)**.** *Let $\alpha \in (0, 1/4)$ be a fixed constant. Suppose d is the input dimensionality, $\frac{\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle}{d} = \mathbb{1}_{i=j} \pm O(\sqrt{\frac{\log n}{d}}), \forall i, j \in [n]$, the data matrix $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^n$ has spectral norm $\left\|\boldsymbol{X}\boldsymbol{X}^\top\right\| = \Theta(n)$, and for the labels we have $|y_i| \leq 1\ \forall y_i$. Assume the number of training samples n and the network width m satisfy $n, m = \Omega(d^{1+\alpha}), n, m \leq d^{\mathcal{O}(1)}$, and the learning rate $\eta \ll d$. Then, there exist a universal constant C, such that with high probability for all $0 \leq t \leq T = C \cdot \frac{d \log d}{\eta}$, the network $f(\boldsymbol{w}_t, \boldsymbol{X})$ trained with GD is very close to a linear function $f^{lin}(\boldsymbol{\beta}, \boldsymbol{X})$:*

$$\frac{1}{n}\sum_{i=1}^{n}(f^{lin}(\boldsymbol{\beta}_t, \boldsymbol{X}) - f(\boldsymbol{w}_t, \boldsymbol{X}))^2 \leq \frac{\eta^2 t^2}{d^{2+\Omega(\alpha)}} \leq \frac{1}{d^{\Omega(\alpha)}}. \tag{11}$$

In particular, the linear model $f^{lin}(\boldsymbol{\beta}, \boldsymbol{X})$ operates on the transformed data $\boldsymbol{\psi}(\boldsymbol{x})$, where

$$\begin{aligned}
\boldsymbol{\psi}(\boldsymbol{x}) &= \begin{bmatrix} \sqrt{\frac{2}{d}}\zeta \boldsymbol{x} \\ \sqrt{\frac{3}{2d}}\nu \\ \vartheta_0 + \vartheta_1(\frac{\|\boldsymbol{x}\|}{\sqrt{d}} - 1) + \vartheta_2(\frac{\|\boldsymbol{x}\|}{\sqrt{d}} - 1)^2 \end{bmatrix} \\
\zeta &= \mathop{\mathbb{E}}_{g \sim \mathcal{N}(0,1)}[\phi'(g)] \\
\nu &= \mathop{\mathbb{E}}_{g \sim \mathcal{N}(0,1)}[g\phi'(g)]\sqrt{\frac{\mathrm{Tr}[\boldsymbol{\Sigma}^2]}{d}} \\
\vartheta_0 &= \mathop{\mathbb{E}}_{g \sim \mathcal{N}(0,1)}[g] \\
\vartheta_1 &= \mathop{\mathbb{E}}_{g \sim \mathcal{N}(0,1)}[g\phi'(g)] \\
\vartheta_2 &= \mathop{\mathbb{E}}_{g \sim \mathcal{N}(0,1)}[(\frac{1}{2}g^3 - g)\phi'(g)]
\end{aligned}$$

Note that $\boldsymbol{\psi}(\boldsymbol{x})$ consists of a scaled version of the data, a bias term, and a term that depends on the norm of the example. We will adopt the notation $f(\boldsymbol{\psi}; \boldsymbol{\beta}) = \boldsymbol{\psi}^\top \boldsymbol{\beta}$ for the linear model.

We can now formally state A.1:

Assumption A.6 (formal version of A.1)**.** Suppose that Theorem A.5 holds. Then with high probability, for all such t the following also holds for all $c \in \mathcal{C}$ and for all $s \in \mathcal{A}$:

$$\begin{aligned}
|f^{lin}(\boldsymbol{\beta}_t, \boldsymbol{v}_c) - f(\boldsymbol{w}_t, \boldsymbol{v}_c)| &\leq \frac{\eta t}{d^{1+\Omega(\alpha)}}, \\
|f^{lin}(\boldsymbol{\beta}_t, \boldsymbol{v}_s) - f(\boldsymbol{w}_t, \boldsymbol{v}_s)| &\leq \frac{\eta t}{d^{1+\Omega(\alpha)}}.
\end{aligned}$$

We will assume the former holds in the proof of the following theorems, although as we will see the assumption is unnecessary for Theorem. 4.2.

B Proof for Theorems

B.1 Notation

For the analysis, we split $\boldsymbol{\beta}$ into its components corresponding to the data, bias and norm parts of $\boldsymbol{\psi}$; that is $\boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{\beta}' \\ \beta_{bias} \\ \beta_{norm} \end{pmatrix}$ for $\boldsymbol{\beta}' \in \mathbb{R}^d, \beta_{bias} \in \mathbb{R}, \beta_{norm} \in \mathbb{R}$. We use the inner product between $\boldsymbol{\beta}'$ and a feature $\boldsymbol{v}$ to understand how well the linear model learns a feature $\boldsymbol{v} \in \mathbb{R}^d$. With slight abuse of notation, we will simply write $\langle \boldsymbol{\beta}, \boldsymbol{v} \rangle$ to mean $\langle \boldsymbol{\beta}', \boldsymbol{v} \rangle$.

We also define the matrix $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1 \quad \dots \quad \boldsymbol{\phi}_n]^\top$.

B.2 PROOF OF THEOREM 4.1 AND 4.2

Theorem 4.1. *Let $\alpha \in (0, \frac{1}{4})$ be a fixed constant. Suppose the number of training samples n and the network width m satisfy $n \gtrsim d^{1+\alpha}$ and $m \gtrsim d^{1+\alpha}$. Let n_c be the number of examples in class c, and $n_{c,s} = |g_{c,s}|$ be the size of group $g_{c,s}$ with label c and spurious feature $\boldsymbol{v}_s \in \mathcal{A}$. Then, under the setting of Sec. 3 there exist a constant $\nu_1 > 0$, such that with high probability, for all $0 \le t \le \nu_1 \cdot \sqrt{\frac{d^{1-\alpha}}{\eta}}$, the contribution of the core and spurious features to the network output can be quantified as follows:*

$$f(\boldsymbol{v}_c; \boldsymbol{W}_t, \boldsymbol{z}_t) = \sqrt{\frac{2}{d}}\eta\zeta c\|\boldsymbol{v}_c\|^2 t\left(\frac{n_c}{n} \pm \mathcal{O}(d^{-\Omega(\alpha)})\right), \tag{5}$$

$$f(\boldsymbol{v}_s; \boldsymbol{W}_t, \boldsymbol{z}_t) = \sqrt{\frac{2}{d}}\eta\zeta c\|\boldsymbol{v}_s\|^2 t\left(\frac{n_{c,s} - n_{c',s}}{n} \pm \mathcal{O}(d^{-\Omega(\alpha)})\right), \tag{6}$$

where $c' = \mathcal{C} \setminus c$, and ζ is the expected gradient of activation functions at random initialization.

Corollary 4.2 (Separability of majority and minority groups). *Suppose that for all classes, a majority group has at least K examples and a minority group has at most k examples. Then, under the assumptions of Theorem 4.1, examples in the majority and minority groups are separable based on the model's output, early in training. That is, for all $0 \le t \le \nu_1 \cdot \sqrt{\frac{d^{1-\alpha}}{\eta}}$, with high probability, the following holds for at least $1 - \mathcal{O}(d^{-\Omega(\alpha)})$ fraction of the training examples $\boldsymbol{x}_i$ in group $g_{c,s}$:*

If $g_{c,s}$ is in a majority group in class $c = 1$:

$$f(\boldsymbol{x}_i; \boldsymbol{W}_t, \boldsymbol{z}_t) \ge \frac{2\eta\zeta^2 t}{d}\left(\frac{\|\boldsymbol{v}_s\|^2(K-k)}{n} + \xi \pm O(d^{-\Omega(\alpha)})\right) + \rho(t, \phi, \Sigma), \tag{7}$$

If $g_{c,s}$ is in a minority group in class $c = 1$, but $g_{c',s}$ is a majority group in class $c' = -1$:

$$f(\boldsymbol{x}_i; \boldsymbol{W}_t, \boldsymbol{z}_t) \le \frac{2\eta\zeta^2 t}{d}\left(-\frac{\|\boldsymbol{v}_s\|^2(K-k)}{n} + \xi \pm O(d^{-\Omega(\alpha)})\right) + \rho(t, \phi, \Sigma), \tag{8}$$

where ρ is constant for all examples in the same class, $\xi \sim \mathcal{N}(0, \kappa)$ with $\kappa = \frac{1}{n}(\sum_c n_c^2\sigma_c^2\|\boldsymbol{v}_c\|^2)^{1/2} + \frac{1}{n}(\sum_s (n_{c,s} - n_{c',s})^2\sigma_s^2\|\boldsymbol{v}_s\|^2)^{1/2}$ is the total effect of noise on the model.

Analogous statements holds for the class $c = -1$ by changing the sign and direction of the inequality.

As in Hu et al. (2020), we will conduct our analysis under the high probability events that $\|\boldsymbol{\Psi}^\top\boldsymbol{\Psi}\| = O(\frac{n}{d})$ and for all training data $\boldsymbol{x}$, $\frac{\|\boldsymbol{x}\|}{\sqrt{d}} = 1 \pm O(\sqrt{\frac{\log n}{d}})$.

Starting from the rule of gradient descent

$$\begin{aligned}\boldsymbol{\beta}(t+1) &= \boldsymbol{\beta}(t) - \frac{\eta}{n}\boldsymbol{\Psi}^\top(\boldsymbol{\Psi}\boldsymbol{\beta}(t) - \boldsymbol{y}) \\ &= \left(I - \frac{\eta}{n}\boldsymbol{\Psi}^\top\boldsymbol{\Psi}\right)\boldsymbol{\beta}(t) + \frac{\eta}{n}\boldsymbol{\Psi}^\top\boldsymbol{y}\end{aligned}$$

Let $\boldsymbol{A} = \boldsymbol{I} - \frac{\eta}{n}\boldsymbol{\Psi}^\top\boldsymbol{\Psi}, \boldsymbol{b} = \frac{\eta}{n}\boldsymbol{\Psi}^\top\boldsymbol{y}$. Also, $\boldsymbol{A}$ can be diagonalized as $\boldsymbol{A} = \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}^\top$. Since $\|\boldsymbol{\Psi}^\top\boldsymbol{\Psi}\| = O(\frac{n}{d})$, the eigenvalues of $\boldsymbol{A}$, call them $\lambda_1, \ldots, \lambda_d$, are of order $1 - O(\frac{\eta}{d})$. For $t \ge 1$, the previous recurrence relation admits the solution

$$\begin{aligned}\boldsymbol{\beta}(t) &= (\boldsymbol{I} + \boldsymbol{A} + \cdots + \boldsymbol{A}^{t-1})\boldsymbol{b} \\ &= \boldsymbol{V}(\boldsymbol{I} + \boldsymbol{D} + \cdots + \boldsymbol{D}^{t-1})\boldsymbol{V}^\top\boldsymbol{b}\end{aligned}$$

When $t = O(\sqrt{\frac{d^{1-\alpha}}{\eta}})$, the eigenvalues of $\boldsymbol{I} + \boldsymbol{D} + \cdots + \boldsymbol{D}^{t-1}$ are on the order of

$$\begin{aligned}1 + \lambda_i + \cdots + \lambda_i^{t-1} &= \frac{1 - \lambda_i^t}{1 - \lambda_i} \\ &= 1 + O(d^{-\frac{\alpha}{2}})\end{aligned}$$

Thus we can approximate $\boldsymbol{I} + \boldsymbol{D} + \cdots + \boldsymbol{D}^{t-1} = t\boldsymbol{I} + \boldsymbol{\Delta}$, where $\|\boldsymbol{\Delta}\| = O(d^{-\frac{\alpha}{2}})$. Then

$$\beta(t) = \boldsymbol{V}(t\boldsymbol{I} + \boldsymbol{\Delta})\boldsymbol{V}^\top \boldsymbol{b} = t\boldsymbol{b} + \boldsymbol{\Delta}_1 \boldsymbol{b}$$

where $\boldsymbol{\Delta}_1 = \boldsymbol{V}\boldsymbol{\Delta}\boldsymbol{V}^\top$ also satisfies $\|\boldsymbol{\Delta}\| = O(d^{-\frac{\alpha}{2}})$.

From here we may calculate the following: the alignment of $\boldsymbol{\beta}$ with a core feature $\boldsymbol{v}_c$ is

$$\langle \boldsymbol{v}_c, \boldsymbol{\beta} \rangle = \sqrt{\frac{2}{d}} \frac{\eta\zeta c \|\boldsymbol{v}_c\|}{n} (t \pm O(d^{-\frac{\alpha}{2}}))(\|\boldsymbol{v}_c\| n_c \pm O(\sigma_c \sqrt{n})) \tag{12}$$

$$= \sqrt{\frac{2}{d}} \eta\zeta c \|\boldsymbol{v}_c\|^2 t \left(\frac{n_c}{n} \pm O(d^{-\Omega(\alpha)}) \right) \tag{13}$$

and the alignment with a spurious feature $\boldsymbol{v}_s$ is

$$\langle \boldsymbol{v}_s, \boldsymbol{\beta} \rangle = \sqrt{\frac{2}{d}} \frac{\eta\zeta c \|\boldsymbol{v}_s\|}{n} (t \pm O(d^{-\frac{\alpha}{2}}))(\|\boldsymbol{v}_s\| (n_{c,s} - n_{c',s}) \pm O(\sigma_s \sqrt{n})) \tag{14}$$

$$= \sqrt{\frac{2}{d}} \eta\zeta c \|\boldsymbol{v}_s\|^2 t \left(\frac{n_{c,s} - n_{c',s}}{n} \pm O(d^{-\Omega(\alpha)}) \right) \tag{15}$$

The effect of the noise is captured by the $O(\sigma\sqrt{n})$ terms, following standard concentration inequalities, and we used the fact that $\frac{1}{\sqrt{n}} = O(d^{-\Omega(\alpha)})$. The result transfers to the full neural network under assumption A.6, namely

$$f(\boldsymbol{v}_c; \boldsymbol{W}_t, \boldsymbol{z}_t) = \sqrt{\frac{2}{d}} \eta\zeta c \|\boldsymbol{v}_c\|^2 t \left(\frac{n_c}{n} \pm \mathcal{O}(d^{-\Omega(\alpha)}) \right), \tag{16}$$

$$f(\boldsymbol{v}_s; \boldsymbol{W}_t, \boldsymbol{z}_t) = \sqrt{\frac{2}{d}} \eta\zeta c \|\boldsymbol{v}_s\|^2 t \left(\frac{n_{c,s} - n_{c',s}}{n} \pm \mathcal{O}(d^{-\Omega(\alpha)}) \right), \tag{17}$$

This proves Theorem 4.1.

In addition, we calculate that

$$\beta_{norm}(t) = (tI + \boldsymbol{\Delta}_1) \sum_{i=1}^{n} y_i \left(\vartheta_0 + \vartheta_1 (\frac{\|\boldsymbol{x}_i\|}{\sqrt{d}} - 1) + \vartheta_2 (\frac{\|\boldsymbol{x}_i\|}{\sqrt{d}} - 1)^2 \right) = O(\frac{\eta t}{\sqrt{n}})$$

Then for the predictions at time t for an example in class $c = 1$, group $g_{1,s}$:

$$\boldsymbol{\psi}(\boldsymbol{x})^\top \boldsymbol{\beta}(t) = \sqrt{\frac{2}{d}} \zeta \boldsymbol{x}^\top \boldsymbol{\beta}' + \sqrt{\frac{3}{2d}} \nu \beta_{bias}(t) + \beta_{norm}(t) \left(\vartheta_0 + \vartheta_1 (\frac{\|\boldsymbol{x}\|}{\sqrt{d}} - 1) + \vartheta_2 (\frac{\|\boldsymbol{x}\|}{\sqrt{d}} - 1)^2 \right)$$

$$= \sqrt{\frac{2}{d}} \zeta (\boldsymbol{v}_1 + \boldsymbol{v}_s + \boldsymbol{\xi})^\top \boldsymbol{\beta}' + \sqrt{\frac{3}{2d}} \nu \beta_{bias}(t) + \vartheta_0 \beta_{norm}(t) \pm O\left(\eta t \sqrt{\frac{\log n}{nd}} \right)$$

We have a few cases

1. $g_{1,k}$ is a majority group. In this case

$$\boldsymbol{\psi}(\boldsymbol{x})^\top \boldsymbol{\beta}(t) \geq \frac{2\eta\zeta^2 t}{d} \left(\frac{n_1 \|\boldsymbol{v}_c\|^2}{n} + \frac{\|\boldsymbol{v}_s\|^2 (K-k)}{n} + \left\langle \boldsymbol{\xi}, \frac{1}{n} \boldsymbol{X}^\top \boldsymbol{y} \right\rangle \pm O(d^{-\Omega(\alpha)}) \right)$$
$$+ \sqrt{\frac{3}{2d}} \nu \beta_{bias}(t) + \vartheta_0 \beta_{norm}(t) \pm O\left(\eta t \sqrt{\frac{\log n}{nd}} \right)$$

2. $g_{1,k}$ is a minority group and $g_{-1,k}$ is a majority group. In this case

$$\boldsymbol{\psi}(\boldsymbol{x})^\top \boldsymbol{\beta}(t) \leq \frac{2\eta\zeta^2 t}{d} \left(\frac{n_1 \|\boldsymbol{v}_c\|^2}{n} - \frac{\|\boldsymbol{v}_s\|^2 (K-k)}{n} + \left\langle \boldsymbol{\xi}, \frac{1}{n} \boldsymbol{X}^\top \boldsymbol{y} \right\rangle \pm O(d^{-\Omega(\alpha)}) \right)$$
$$+ \sqrt{\frac{3}{2d}} \nu \beta_{bias}(t) + \vartheta_0 \beta_{norm}(t) \pm O\left(\eta t \sqrt{\frac{\log n}{nd}} \right)$$

3. $g_{1,k}$ is such that no majority groups have the spurious feature. In this case

$$\boldsymbol{\psi}(\boldsymbol{x})^\top \boldsymbol{\beta}(t) = \frac{2\eta\zeta^2 t}{d}\left(\frac{n_1\|\boldsymbol{v}_c\|^2}{n} + \frac{\|\boldsymbol{v}_s\|^2\tilde{k}}{n} + \left\langle \boldsymbol{\xi}, \frac{1}{n}\boldsymbol{X}^\top \boldsymbol{y}\right\rangle \pm O(d^{-\Omega(\alpha)})\right) + \sqrt{\frac{3}{2d}}\nu\beta_{bias}(t) + \vartheta_0\beta_{norm}(t) \pm O\left(\eta t\sqrt{\frac{\log n}{nd}}\right), \qquad |\tilde{k}| \le k$$

Now

$$\left\langle \boldsymbol{\xi}, \frac{1}{n}\boldsymbol{X}^\top \boldsymbol{y}\right\rangle = \sum_{c\in\{\pm 1\}} \frac{\|\boldsymbol{v}_c\|n_c}{n}\langle \boldsymbol{\xi}, \boldsymbol{v}_c\rangle + \sum_s \frac{\|\boldsymbol{v}_s\|(n_{1,s} - n_{-1,s})}{n}\langle \boldsymbol{\xi}, \boldsymbol{v}_s\rangle + \left\langle \boldsymbol{\xi}, \frac{1}{n}\sum_{i=1}^n \boldsymbol{\xi}_i y_i\right\rangle \tag{18}$$

$$= \sum_{c\in\{\pm 1\}} \frac{\|\boldsymbol{v}_c\|n_c}{n}\langle \boldsymbol{\xi}, \boldsymbol{v}_c\rangle + \sum_s \frac{\|\boldsymbol{v}_s\|(n_{1,s} - n_{-1,s})}{n}\langle \boldsymbol{\xi}, \boldsymbol{v}_s\rangle \pm O\left(\sqrt{\frac{d}{n}}\right) \tag{19}$$

$$\sim \mathcal{N}(0, \kappa) \pm O(d^{-\Omega(\alpha)}) \tag{20}$$

Finally, observe that $O\left(\eta t\sqrt{\frac{\log n}{nd}}\right) = O(d^{-1-\Omega(\alpha)})$. Combining all these results and setting $\rho_1 = \frac{2\eta\zeta^2 ct}{d}, \rho_2 = \frac{\rho_1 n_1\|\boldsymbol{v}_c\|^2}{n} + \sqrt{\frac{3}{2d}}\nu\beta_{bias}(t) + \vartheta_0\beta_{norm}(t)$ shows Theorem 4.2 when looking at the prediction of the linear model. Recall that Hu et al. (2020) showed that the average squared error in predictions between the linear model and the full neural network is $O(\frac{\eta^2 t^2}{d^{2+\Omega(\alpha)}})$. Then by Markov's inequality, we can guarantee that the predictions of the linear model differ by at most $O(\frac{\eta t}{d^{1+\Omega(\alpha)}})$ for at least $1 - O(d^{-\Omega(\alpha)})$ proportion of the examples. This error can be factored into the existing error term. Hence the result holds for the full neural network.

We can apply the same argument for the class c'. Thus Theorem 4.2 is proven.

Notably, Theorem 4.2 only depends on the closeness of the neural network and the initial linear model on the training data, hence does not rely on assumption A.6.

B.3 Proof of Theorem 4.3

Theorem 4.3. *Under the assumptions of Theorem 4.1, if the classes are balanced, and the total size of the minority groups in class c is small, i.e., $\mathcal{O}(n^{1-\gamma})$ for some $\gamma > 0$, then there exists a constant $\nu_2 > 0$ such that at $T = \nu_2 \cdot \frac{d\log d}{\eta}$, for an example $\boldsymbol{x}_i$ in a majority group $g_{c,s}$, the contribution of the core feature to the model's output is at most:*

$$|f(\boldsymbol{v}_c; \boldsymbol{W}_T, \boldsymbol{z}_T)| \le \sqrt{d}\frac{R_s}{\zeta R_c} + \mathcal{O}(n^{-\gamma}\sqrt{d}) + \mathcal{O}(d^{-\Omega(\alpha)}). \tag{9}$$

In particular if $\min\{R_c, 1\} \gg R_s$, then the model's output is mostly indicated by the spurious feature instead of the core feature:

$$|f(\boldsymbol{v}_s; \boldsymbol{W}_T, \boldsymbol{z}_T)| \ge \frac{\sqrt{d}}{2\zeta} \gg |f(\boldsymbol{v}_c; \boldsymbol{W}_T, \boldsymbol{z}_T)|. \tag{10}$$

Let g_{maj} be the total number of majority groups among all classes. Note that by the definition of majority groups, g_{maj} is at most the number of classes, namely 2 in the given analysis.

Since the classes are balanced with labels ± 1, it is not hard to see that the bias term in the weights will always be zero, hence we may as well assume that we do not have the bias term. Abusing notation, we will still denote quantities by the same symbol, even though now the bias term has been removed.

First consider a model $\tilde{f} = \boldsymbol{\psi}^\top \tilde{\boldsymbol{\beta}}$ trained on the dataset $\mathcal{D}_{\text{maj}}$, which only contains examples from the majority groups. Further, assume $\mathcal{D}_{\text{maj}}$ has infinitely many examples so that the noise perfectly matches the underlying distribution. We prove the results in this simplified setting then extend the result using matrix perturbations.

We have

$$\mathcal{L} = \frac{1}{2} \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}} [(\boldsymbol{\psi}_i^\top \tilde{\boldsymbol{\beta}} - y_i)^2]$$
$$\nabla \mathcal{L} = \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}} [(\boldsymbol{\psi}_i^\top \tilde{\boldsymbol{\beta}} - y_i)\boldsymbol{\psi}_i]$$

and the optimal $\tilde{\boldsymbol{\beta}}_*$ satisfies

$$\tilde{\boldsymbol{\beta}}_* = \left(\mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}} [\boldsymbol{\psi}_i \boldsymbol{\psi}_i^\top] \right)^\dagger \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}} [y_i \boldsymbol{\psi}_i]$$

where † represents the Moore-Penrose pseudo-inverse.

Since the noise is symmetrical with respect to the classes, the bias and norm terms of $\boldsymbol{\beta}$ must be zero. Thus the loss becomes

$$\mathcal{L} = \frac{1}{2} \mathop{\mathbb{E}}_{(\boldsymbol{x}_i, y_i) \sim \mathcal{D}_{\text{maj}}} \left[(\sqrt{\frac{2}{d}} \zeta \boldsymbol{x}_i^\top \tilde{\boldsymbol{\beta}}' - y_i)^2 \right] \tag{21}$$
$$= \frac{1}{2} \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}} \left[(\sqrt{\frac{2}{d}} \zeta (\boldsymbol{v}_{c_i} + \boldsymbol{v}_{s_i} + \boldsymbol{\xi}_i)^\top \tilde{\boldsymbol{\beta}}' - y_i)^2 \right] \tag{22}$$
$$= \frac{1}{2} \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}} \left[(\sqrt{\frac{2}{d}} \zeta (\boldsymbol{v}_{c_i} + \boldsymbol{v}_{s_i})^\top \tilde{\boldsymbol{\beta}}' - y_i)^2 + (\sqrt{\frac{2}{d}} \zeta \boldsymbol{\xi}_i^\top \tilde{\boldsymbol{\beta}}')^2 \right] \tag{23}$$
$$= \frac{1}{2} \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}} \left[(\sqrt{\frac{2}{d}} \zeta (\boldsymbol{v}_{c_i} + \boldsymbol{v}_{s_i})^\top \tilde{\boldsymbol{\beta}}' - y_i)^2 \right] + \frac{\zeta^2}{d} \tilde{\boldsymbol{\beta}}'^\top \boldsymbol{\Sigma}_\xi \tilde{\boldsymbol{\beta}}' \tag{24}$$

Consider the model $\boldsymbol{\beta}_s$ which only learns the spurious features of majority groups

$$\boldsymbol{\beta}'_s = \sqrt{\frac{d}{2}} \frac{1}{\zeta} \sum_{g_{c,s} \text{is a majority group}} \frac{c \boldsymbol{v}_s}{\|\boldsymbol{v}_s\|^2}.$$

Note that for any example in a majority group, $(\boldsymbol{v}_{c_i} + \boldsymbol{v}_{s_i})^\top \boldsymbol{\beta}'_s - y_i = 0$. Thus

$$\mathcal{L} = \frac{\zeta^2}{d} \tilde{\boldsymbol{\beta}}'^\top \boldsymbol{\Sigma}_\xi \tilde{\boldsymbol{\beta}}'$$
$$= \sum_{\boldsymbol{v}_s \text{is spurious}} \frac{\sigma_s^2}{2\|\boldsymbol{v}_s\|^2}$$
$$\leq \frac{g_{\text{maj}} R^2}{2}$$

The loss for the optimal model must be smaller. But the loss due to the last term in equation 24 along a core feature alone is

$$\frac{\zeta^2 \sigma_c^2}{\|\boldsymbol{v}_c\|^2 d} \langle \boldsymbol{v}_c, \boldsymbol{\beta}'_* \rangle^2 \leq \frac{g_{\text{maj}} R^2}{2}$$

Rearranging gives

$$\langle \boldsymbol{v}_c, \boldsymbol{\beta}'_* \rangle^2 \leq \frac{d g_{\text{maj}} R^2 \|\boldsymbol{v}_c\|^2}{2\zeta^2 \sigma_c^2} \tag{25}$$

Now consider the loss from the first term in equation 24 due to a majority group. It must be at least

$$\frac{K}{n} \left(1 - \sqrt{\frac{2}{d}} \zeta \langle \boldsymbol{v}_s, \boldsymbol{\beta}'_* \rangle - \frac{\sqrt{g_{\text{maj}}} R \|\boldsymbol{v}_c\|}{\sigma_c} \right)^2 \leq \frac{g_{\text{maj}} R^2}{2}$$
$$1 - \sqrt{\frac{2}{d}} \zeta \langle \boldsymbol{v}_s, \boldsymbol{\beta}'_* \rangle - \frac{\sqrt{g_{\text{maj}}} R \|\boldsymbol{v}_c\|}{\sigma_c} \leq \sqrt{\frac{n g_{\text{maj}} R^2}{2K}}$$
$$1 - \sqrt{g_{\text{maj}}} R \left(\frac{\|\boldsymbol{v}_c\|}{\sigma_c} + \sqrt{\frac{n}{2K}} \right) \leq \sqrt{\frac{2}{d}} \zeta \langle \boldsymbol{v}_s, \boldsymbol{\beta}'_* \rangle$$

Note that $\sqrt{\frac{n}{2K}} \leq \sqrt{\frac{g_{maj}}{2}}$. Now if we have R sufficiently smaller than $\frac{\sigma_c}{\sqrt{g_{maj}}\|\boldsymbol{v}_c\|}$ and $\frac{2}{g_{\text{maj}}}$, we can guarantee that the RHS is at least some constant less than 1, say $\frac{1}{\sqrt{2}}$. In this case, we have

$$\langle \boldsymbol{v}_s, \boldsymbol{\beta}_* \rangle^2 \geq \frac{d}{4\zeta^2} \tag{26}$$

Under these assumptions it is clear from equation 25 that we will also have

$$\frac{d}{4\zeta^2} \gg \langle \boldsymbol{v}_c, \boldsymbol{\beta}_* \rangle^2 \tag{27}$$

Now we return to the original dataset, which contains minority groups and only a finite number of examples. Again, we have

$$\boldsymbol{\beta}_* = (\boldsymbol{\Psi}^\top \boldsymbol{\Psi})^\dagger \boldsymbol{\Psi}^\top \boldsymbol{y}$$

Since we have removed the bias term, it is not hard to show that the matrix $\frac{1}{n}\boldsymbol{\Psi}^\top\boldsymbol{\Psi}$ has all eigenvalues of order $\Theta(\frac{1}{d})$. Now consider the difference between $\|\frac{1}{n}\boldsymbol{\Psi}^\top\boldsymbol{\Psi}\|$ and $\|\mathbb{E}_{\mathcal{D}_{\text{maj}}}[\boldsymbol{\psi}_i\boldsymbol{\psi}_i^\top]\|$. With high probability it will be of order $O(\frac{n_{\text{mino}}}{nd} + \frac{1}{d\sqrt{n}}) = O(\frac{n^{-\gamma}}{d})$, where the first term corresponds to the inclusion of minority groups and the second term corresponds having a finite sample size. It follows that

$$\begin{aligned}\left\|(\frac{1}{n}\boldsymbol{\Psi}^\top\boldsymbol{\Psi})^\dagger - (\mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}}[\boldsymbol{\psi}_i\boldsymbol{\psi}_i^\top])^\dagger\right\| &= O\left(d - \frac{d}{d - O(n^{-\gamma})}\right) \\ &= O(dn^{-\gamma})\end{aligned}$$

A similar argument shows that

$$\|\boldsymbol{\Psi}^\top\boldsymbol{y} - \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}}[y_i\boldsymbol{\psi}_i]\| = O(d^{-\frac{1}{2}}n^{-\gamma})$$

Thus the change in alignment with a feature $\boldsymbol{v}$ is

$$\begin{aligned}\left\|\left\langle\tilde{\boldsymbol{\beta}}_*, \boldsymbol{v}\right\rangle - \langle\boldsymbol{\beta}_*, \boldsymbol{v}\rangle\right\| &= \left\|(\boldsymbol{\Psi}^\top\boldsymbol{\Psi})^\dagger\boldsymbol{\Psi}^\top\boldsymbol{y} - \left(\mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}}[\boldsymbol{\psi}_i\boldsymbol{\psi}_i^\top]\right)^\dagger \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}}[y_i\boldsymbol{\psi}_i]\right\| \|\boldsymbol{v}\| \\ &\leq \Bigg\| \left((\boldsymbol{\Psi}^\top\boldsymbol{\Psi})^\dagger - \left(\mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}}[\boldsymbol{\psi}_i\boldsymbol{\psi}_i^\top]\right)^\dagger\right)\boldsymbol{\Psi}^\top\boldsymbol{y} \\ &\qquad + \left(\mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}}[\boldsymbol{\psi}_i\boldsymbol{\psi}_i^\top]\right)^\dagger(\boldsymbol{\Psi}^\top\boldsymbol{y} - \mathop{\mathbb{E}}_{\mathcal{D}_{\text{maj}}}[y_i\boldsymbol{\psi}_i])\Bigg\|\|\boldsymbol{v}\| \\ &\leq O\left((dn^{-\gamma})(d^{-\frac{1}{2}}) + d(d^{-\frac{1}{2}}n^{-\gamma})\right) \\ &\leq O(n^{-\gamma}\sqrt{d})\end{aligned}$$

Replacing g_{maj} with 2,and combining equations 25, 26 28, and ASsumption A.6, we get

$$|f(\boldsymbol{v}_s; \boldsymbol{W}_T, \boldsymbol{z}_T)| \geq \frac{\sqrt{d}}{2\zeta} \gg \sqrt{d}\frac{R_s}{\zeta R_c} + \mathcal{O}(n^{-\gamma}\sqrt{d}) + \mathcal{O}(d^{-\Omega(\alpha)}) \geq |f(\boldsymbol{v}_c; \boldsymbol{W}_T, \boldsymbol{z}_T)|. \tag{28}$$

which proves the theorem.

C SIGNAL-TO-NOISE RATIO

The noise-to-signal ratio of the spurious affects the group inference to a great extent. Methods such as GDRO and GB rely on the underlying group information and a larger noise-to-signal ratio of the spurious feature (which makes it much harder to infer the groups) does not affect their group information at all. They only provide an upper bound on robust learning **with group information**.

Table 4: Effect of spurious variance (noise) on worst-group accuracy. Learning a spurious feature with a small variance is easy and yields a poor worst-group accuracy for ERM. Under large spurious noise, JTT cannot infer the groups well and performs poorly. In both cases, SPARE *achieves the SOTA worst-group accuracy, and outperforms not only the other group-inference methods, but also GDRO and GB. This shows the remarkable performance of* SPARE *in inferring the groups.*

	Noise Small		**Noise Large**	
	Worst-group	Average	Worst-group	Average
ERM	44.6 ± 4.1	99.8 ± 0.1	86.4 ± 0.4	95.1 ± 1.2
GDRO	84.3 ± 1.7	98.3 ± 1.2	92.3 ± 0.6	97.2 ± 0.1
Group Balancing	86.3 ± 2.2	97.3 ± 1.2	92.2 ± 0.5	96.2 ± 0.6
JTT	82.7 ± 7.8	97.4 ± 1.1	85.7 ± 16.3	97.3 ± 0.9
EIIL	81.8 ± 1.6	94.8 ± 1.5	92.7 ± 3.5	97.1 ± 0.8
SPARE	86.3 ± 2.9	97.9 ± 0.4	94.1 ± 1.6	97.7 ± 0.3

Table 5: Effect of spurious magnitude (signal) on worst-group accuracy. Learning a spurious feature with a larger magnitude is easy and yields a poor worst-group accuracy for ERM. For large spurious signal, EIIL cannot infer the groups well and performs poorly. In both cases, SPARE *archives the SOTA worst-group accuracy, and outperforms not only the other group-inference methods, but also GB and is comparable to GDRO. Again, this shows the remarkable performance of* SPARE *in inferring the groups.*

	Signal Large		**Signal Small**	
	Worst-group	Average	Worst-group	Average
ERM	0.0 ± 0.0	99.5 ± 0.0	97.0 ± 0.5	98.7 ± 0.2
GDRO	74.8 ± 2.8	98.8 ± 0.3	96.6 ± 0.4	98.7 ± 0.3
Group Balancing	76.8 ± 2.8	97.3 ± 0.3	95.8 ± 0.6	98.2 ± 0.4
JTT	76.4 ± 6.1	97.6 ± 1.3	94.8 ± 1.0	98.4 ± 0.4
EIIL	58.8 ± 4.9	97.1 ± 4.9	89.1 ± 4.0	94.7 ± 2.4
SPARE	78.6 ± 4.6	97.2 ± 0.6	95.5 ± 1.0	97.9 ± 0.4

Only JTT, SPARE and EIIL infer the groups, and hence are affected by noise-to-signal ratio during their group inference.

Table 4 shows the results for small and large noise (variance) of the spurious feature. We see that SPARE outperforms JTT and EIIL in terms of worst-group accuracy by up to 8.4% and 4.5% with a better average accuracy. Table 5 shows the results for small and large magnitude of the spurious feature. We see that SPARE outperforms JTT and EIIL in terms of worst-group accuracy by up to 2.2% and 19.8% with a better average accuracy.

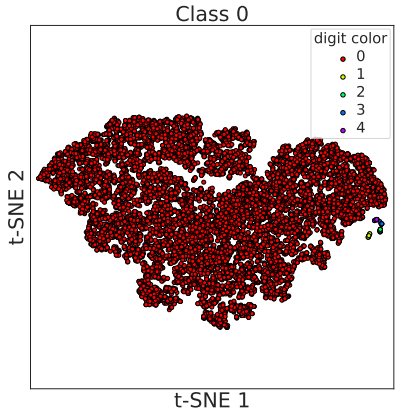
The very similar or sometimes even better worst-group performance of SPARE over GDRO and GB in both tables, confirms how effective SPARE is in inferring the underlying groups for spurious features with various learning difficulties.

D EXPERIMENTATION DETAILS

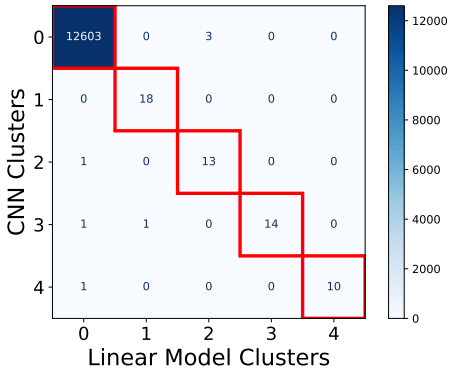
D.1 DATASETS

CMNIST We created a colored MNIST dataset with spurious correlations by using colors as spurious attributes following the settings in Zhang et al. (2022). First, we defined an image classification task with 5 classes by grouping consecutive digits (0 and 1, 2 and 3, 4 and 5, 6 and 7, 8 and 9) into the same class. From the train split, we randomly selected 50,000 examples as the training set, while the remaining 10,000 samples were used as the validation set. The test split follows the official test split of MNIST.

For each class y_i, we assigned a color $\boldsymbol{v}_s$ from a set of colors $\mathcal{A}$={`#ff0000`, `#85ff00`, `#00fff3`, `#6e00ff`, `#ff0018`} as the spurious attribute that highly correlates with this class, represented



(a) t-SNE visualization of the linear model outputs on class 0. Different groups are separable based on the outputs of the linear model.



(b) Groups found by SPARE with the outputs of the neural networks are highly overlapped with the groups found with the outputs of the linear model, demonstrated by near zero values off the diagonal.

Figure 4: Comparing a linear model and a neural network at early stage of training on CMNIST.

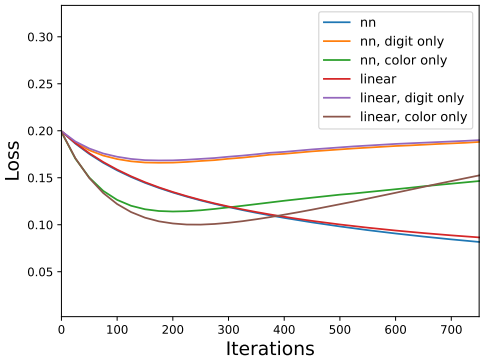


Figure 5: A comparison between the losses of a two-layer network and a simple linear model on the training set, spurious features (color only), and core feature (digit only).

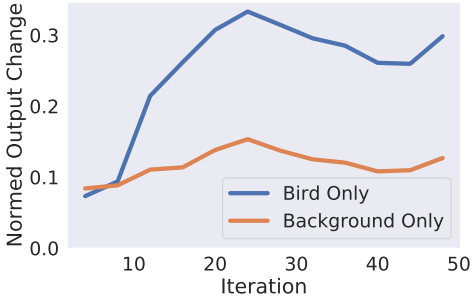


Figure 6: Replicate of Figure 1b on Waterbirds. Inputting only the background (orange line) does not change the model output much (indicating that the background is learned by the model) while inputting only the bird changes the output to a large extent (indicating that the bird is not learned by the model).

by their hex codes, to the foreground of a fraction p_{corr} of the training examples. This fraction represents the majority group for class y_i. The stronger the spurious correlation between class y_i and the spurious attribute $\boldsymbol{v}_s$, the higher the value of p_{corr}. The remaining $1 - p_{corr}$ training examples were randomly colored using a color selected from $\mathcal{A} \setminus \boldsymbol{v}_s$. In our experiments, we set $p_{corr} = 0.995$ to establish significant spurious correlations within the dataset.

Waterbirds is introduced by Sagawa et al. (2019) to study the spurious correlation between the background (land/water) and the foreground (landbird/waterbird) in image recognition. Species in Caltech-UCSD Birds-200-2011 (CUB-200-2011) dataset (Wah et al., 2011) are grouped into two classes, waterbirds and landbirds. All birds are then cut and pasted onto new background images, with waterbirds more likely to appear on water and landbirds having a higher probability on land. There are 4795 training examples in total, 3498 for landbirds with land background, 184 for landbirds with water background, 56 for waterbirds with land background, and 1057 for waterbirds with water background.

CelebA is a large-scale face attribute dataset comprised of photos of celebrities. Each image is annotated with 40 binary attributes, in which “blond hair" and “male" are commonly used for studying

spurious correlations. Specifically, gender is considered a spurious feature for hair color classification. The smallest group is blond male.

D.2 HYPERPARAMETERS

We used SGD as the optimization algorithm to maintain consistency with the existing literature. The hyperparameters employed in our experiments on spurious benchmarks are detailed in Table 6. For the Waterbirds and CelebA datasets, we tuned the learning rate within the range of {1e-4, 1e-5} and weight decay within the range of {1e-1, 1e-0}. These ranges were determined based on the ranges of optimal hyperparameters used by the current state-of-the-art algorithms Creager et al. (2021); Liu et al. (2021); Sagawa et al. (2019); Nam et al. (2021); Zhang et al. (2022). The batch sizes and total training epochs remained consistent with those used in these prior studies. To determine the epoch for separating groups, we performed clustering on the validation set while training the model on the training set to maximize the minimum recall of SPARE's clusters with the groups in the validation set. As mentioned in Section 5, we decided the number of clusters and adjusted the sampling power for each class based on Silhouette scores. Specifically, when the Silhouette score was below 0.9, a sampling power of 2 or 3 was applied, while a sampling power of 1 was used otherwise. It is important to note that other algorithms tuned hyperparameters, such as epochs to separate groups and upweighting factors, by maximizing the worst-group accuracy of fully trained models on the validation set, which is more computationally demanding than the hyperparameter tuning of SPARE.

Table 6: Hyperparameters used for the reported results on different datasets.

DATASET	CMNIST	WATERBIRDS	CELEBA
LEARNING RATE	1E-3	1E-4	1E-5
WEIGHT DECAY	1E-3	1E-1	1E-0
BATCH SIZE	32	128	128
TRAINING EPOCHS	20	300	50
GROUP SEPARATION EPOCH	2	2	1
SILHOUETTE SCORES	[0.997,0.978,0.996,0.991,0.996]	[0.886,0.758]	[0.924,0.757]
SAMPLING POWER	[1,1,1,1,1]	[3,3]	[1,2]

D.3 CHOICES OF MODEL OUTPUTS

In our experiments, we found the worst-group accuracy gets the most improvement when SPARE uses the outputs of the last linear layer to separate the majority from the minority for CMNIST and Waterbirds and use the second to last layer (i.e., the feature embeddings inputted to the last linear layer) to identify groups in CelebA. We speculate that this phenomenon can be attributed to the increased complexity of the CelebA dataset compared to the other two datasets, as employing a higher output dimension help identify groups more effectively.

D.4 DEPENDENCY ON THE CLUSTERING ALGORITHM

The performance of SPARE is not sensitive to the clustering algorithm. The key to SPARE is **clustering the entire model output early in training**. While k-means easily scales to medium-sized datasets, k-median is more suitable for very large datasets, as it can be formulated as a submodular maximization problem (Wolsey, 1982) for which fast and scalable distributed (Mirzasoleiman et al., 2013; 2015) and streaming (Badanidiyuru et al., 2014) algorithms are available.

D.5 CLUSTERING DETAILS

Clustering was performed on all data samples within the same class. It's important to note that k-means doesn't require loading all the data into memory and operates in a streaming manner. As an alternative, we also discussed the possibility of using the k-medoids clustering algorithm and its distributed implementation which uses submodular optimization and easily scales to millions of examples in Section 5. In Table 7, we present the wall-clock times for k-means clustering on

Table 7: Wall-clock times for k-means clustering on Waterbirds, CelebA, CMNIST, and Restricted ImageNet datasets.

CMNIST	Celeba	Waterbirds	Restricted ImageNet
0.46s	31.8s	0.07s	2s

Table 8: Wall-clock runtime comparison of SPARE and SOTA 2-stage algorithms.

ERM	JTT	CnC	SSA	SPARE
1h12m	9h5m	4h25m	2h15m	1h16m

Waterbirds, CelebA, CMNIST, and Restricted ImageNet. It shows that the cost of clustering is negligible when compared to the cost of training.

D.6 TRAINING COST

Table 8 shows a all-clock runtime comparison of SPARE and SOTA 2-stage algorithms. JTT initially trains the identification model for a specific number of epochs and then upsamples misclassified examples by a substantial factor to train the robust model. As a result, the training cost is influenced not just by the training of the identification model but also by the considerable volume of upsampled training data used in the robust model's training. For instance, in the case of CelebA, JTT trains the identification model for just one epoch but then upsamples all misclassified examples (approximately 1/10 of the training set) by a factor of 50. This leads to a training set roughly six times the original size. In this scenario, the large volume of upsampled training data significantly increases the training cost, while the training time for the identification model is almost negligible.

E SPARE REACHES SOTA PERFORMANCE UNDER EXTREME GROUP IMBALANCE.

Many state-of-the-art algorithms that can successfully eliminate spurious correlations in the Waterbirds and CelebA, severely fail on CMNIST, by providing as low as 0% worst-group accuracy. In CMNIST, every class has a very large majority and *four* very small minority groups, and there is a very strong spurious correlation between the color of the majority group and the corresponding class. Here, the small size of the minority groups makes it difficult to infer the groups based on loss (LfF (Nam et al., 2020)), data augmentation (CIM (Taghanaki et al., 2021)), or semi-supervised learning (SSA (Nam et al., 2021)). Besides, state-of-the-art methods that partition every class into only two groups, namely EIIL (Creager et al., 2021), PGI (Ahmed et al., 2020), and JTT (Liu et al., 2021), fail to balance the minority groups. This is because the minority groups need to be extensively upweighted or upsampled to make a balance with the majority group due to their small sizes, and extensive upweighting or upsampling them as a whole exaggerates the small differences between the original size of the minority groups and makes them imbalanced w.r.t. each other. This yields an inferior worst-group accuracy. In contrast, SPARE finds multiple minority clusters (see Figure 1c). By importance sampling from each cluster based on its size, SPARE can successfully balance the groups and achieve state-of-the-art worst-group and average accuracy.

F DISCOVERING SPURIOUS FEATURES

F.1 RESTRICTED IMAGENET

We use Restricted ImageNet proposed in Tsipras et al. (2019) which contains 9 superclasses of ImageNet. The classes and the corresponding ImageNet class ranges are shown in Table 9.

Table 9: Classes included in Restricted ImageNet and their corresponding ImageNet class ranges.

Restricted ImageNet Class	ImageNet class range
dog	151-268
cat	281-285
frog	30-32
turtle	33-37
bird	80-100
primate	365-382
fish	389-397
crab	118-121
insect	300-319

F.2 Experimental Settings

When training on Restricted ImageNet, we use ResNet50 He et al. (2016) from the PyTorch library Paszke et al. (2019) with randomly initialized weights instead of pretrained weights. We followed the hyperparameters specified in Goyal et al. (2017): the model was trained for 90 epochs, with an initial learning rate of 0.1. The learning rate was reduced by a factor of 0.1 at the 30th, 60th, and 80th epochs. During training, we employed Nesterov momentum of 0.9 and applied a weight decay of 0.0001.

F.3 Investigation on Groups Identified by EIIL vs. SPARE

Evaluation setup. As no group-labeled validation set is available to tune the epoch in which the groups are separated, we tried separating groups using ERM models trained for various numbers of epochs. Since both EIIL and SPARE identify the groups early (EIIL infers groups on models trained with ERM for 1 epoch for both Waterbirds and CelebA, as shown in Table 14 and Table 13, and 5 epochs for CMNIST; the group separation epochs for SPARE are epoch 1 or 2 for the three datasets, as shown in Table 6), we tuned the epoch to separate groups in the range of {2,4,6,8} for both algorithms. This tuning was based on the average test accuracy achieved by the final model, as the worst-group accuracy is undefined without group labels. Interestingly, while SPARE did not show sensitivity to the initial epochs on Restricted ImageNet, EIIL achieved the highest average test accuracy when the initial models were trained for 4 epochs using ERM. We manually labeled examples with their groups for test data.

EIIL finds groups of misclassified examples while SPARE finds groups with spurious features. We observed that EIIL effectively separates examples that have 0% classification accuracy as the minority group, as demonstrated in Table 10. This separation is analogous to the error-splitting strategy employed by JTT Liu et al. (2021) when applied to the same initial model. This similarity in behavior is also discussed in Creager et al. (2021). Instead of focusing on misclassified examples, SPARE separates the examples that are learned early in training. Table 11 shows that the first cluster found by SPARE have almost 100% accuracy, indicating that the spurious feature is learned for such examples. Downweighting examples that are learned early allows for effectively mitigating the spurious correlation.

SPARE upweights outliers less than EIIL. Heavily upweighting misclassified examples can be problematic for this more realistic dataset than the spurious benchmarks as the misclassified ones are likely to be outliers, noisy-labeled or contain non-generalizable information. Table 10 shows that groups inferred by EIIL are more imbalanced, which makes EIIL upweights misclassified examples more than SPARE. As shown in Table 1, this heavier upweighting of misclassified examples with EIIL drops accuracy not only for the minority groups but also for the overall accuracy. Therefore, we anticipate that this effect would persist or become even more pronounced for methods like JTT, which directly identify misclassified examples as the minority group. In contrast, SPARE separates groups based on the spurious feature that is learned early, and upweights the misclassified examples less than other methods due to the more balanced size of the clusters. This allows SPARE to more effectively mitigate spurious correlations than others.

Table 10: Accuracy (%) of training examples in different classes of Restricted ImageNet in the two environments inferred by EIIL. EIIL trains models with Group DRO on the inferred environments, resulting in up-weighting misclassified examples in Env 2.

Class	dog	cat	frog	turtle	bird	primate	fish	crab	insect
Env 1 ERM acc	98	37	26	62	76	78	78	71	90
Env 2 ERM acc	0	0	0	0	0	0	0	0	0
Env 1 size	144378	488	457	2875	17157	11233	6817	2172	21112
Env 2 size	3495	6012	3443	3625	9984	12167	4417	3028	4888

Table 11: Accuracy (%) of training examples in different classes of Restricted ImageNet in the two groups inferred by SPARE at epoch 8.

Class	dog	cat	frog	turtle	bird	primate	fish	crab	insect
Cluster 1 ERM acc	100	100	100	100	100	99	100	100	100
Cluster 2 ERM acc	64	9	11	14	28	13	27	16	36
Cluster 1 size	130541	3236	1578	2684	18870	12158	7331	2566	18974
Cluster 2 size	17332	3264	2322	3816	8271	11242	3903	2634	7026

G ABLATION STUDIES

Importance Sampling Power (λ). Next, we explain how we determine the importance of different clusters using silhouette scores. A higher average silhouette score indicates that clusters are more separated. In this case, groups can be accurately identified and we can balance the groups using $\lambda = 1$. However, when clusters are not well separated (lower silhouette score), some examples from the majority group are spread in smaller clusters. In this case, sampling less from the large clusters is enough to balance the groups, as the majority groups are sampled when we upsample the small clusters. Here, we can balance the groups using $\lambda \geq 2$. Empirically, we found that $\lambda = 1/2/3$ is enough to effectively mitigate the spurious correlation in all our experiments.

Table 12 presents the average silhouette score for each class in different datasets. A higher average silhouette score indicates that clusters are well separated, such as in CMNIST and the female class in CelebA. This means we can accurately identify both the majority and minority groups. However, when clusters are not clearly separated (lower silhouette scores), some examples from the majority group get mixed up with the smaller clusters. As a result, we sample even fewer examples from the larger clusters. When clusters are well separated, we use $\lambda = 1$ to ensure equal treatment of groups. However, for less separable clusters, using $\lambda \geq 2$ helps achieve group balance.

H COMPARING INFERRED WITH GROUND-TRUTH GROUPS

In Table 13 and Table 14, we compare the clusters found by SPARE vs. (1) misclassified examples found by JTT, (2) environments inferred by EIIL, and (3) pseudo-labels learned by SSA.

Table 12: Average Silhouette scores of clusters in different classes, and the corresponding importance sampling power (λ) used for each class.

Dataset	Silhouette score	Sampling power (λ)
CMNIST	between 0.991-0.997	[1, 1, 1, 1, 1]
Waterbirds	[0.886, 0.758]	[3, 3]
CelebA	[0.924, 0.757]	[1, 2]

H.1 IMPLEMENTATION OF BASELINES

Both JTT Liu et al. (2021) and EIIL Creager et al. (2021) require training an ERM model to identify groups of examples for upweighting or downweighting. For clarity, we will refer to this ERM model as the *reference model*, which is equivalent to the *identification models* defined in Liu et al. (2021).

JTT. We train the reference model from ImageNet-pre-trained weights with ERM based on the optimal hyperparameters reported in Liu et al. (2021) and upsample training examples misclassified by the identification models. For Waterbirds, we train the identification model for 60 epochs with a learning rate 1e-5 and weight decay 1. For CelebA, the identification model is trained for 1 epoch with a learning rate 1e-5 and weight decay 0.1.

EIIL. For Waterbirds, we follow the environment inference steps explained in Creager et al. (2021): we use an ERM model trained for 1 epoch as the reference model and optimize the EI objective of EIIL with learning rate 0.01 for 20, 000 steps using the Adam optimizer. As no experiment was conducted on CelebA in the original paper Creager et al. (2021), we follow the proposal in Nam et al. (2021), which took the same EI procedure for CelebA as for Waterbirds.

SSA. We implement SSA based on the pseudo-code and experimental details explained in Nam et al. (2021). Please refer to Nam et al. (2021) for details of the setups. As the pseudo-attribute predictor shares the same architecture as the robust model but is trained on the validation set, to make the inference cost comparable across all methods, we report the inference cost of SSA by converting the number of training-on-validation steps for the pseudo-attribute predictor to the number of training-on-train epochs that involve the same total number of gradient backward steps.

H.2 COMPARISON OF GROUPS.

CelebA. We start from the CelebA dataset, where we observed more significant disparities among the groups identified by different algorithms, as demonstrated in Table 13. JTT simply upweights the smaller class (i.e., blond hair), as most examples from that class are misclassified due to the strong class imbalance. Similarly, EIIL assigns higher weights to more examples from the smaller class.

On the other hand, when examining the confusion matrices, we found that both SSA and SPARE successfully discover groups that closely align with the ground-truth groups in CelebA. Note that SPARE requires much less training than SSA. However, upon visualizing the samples, we noticed that the upweighted examples identified by SSA exhibit some characteristics learned from the validation set that are more correlated with a certain gender. For instance, 11.4% of the upweighted examples of blonde females and only 1.2% of the downweighted examples wear sunglasses, which is a feature that is correlated more with males in the validation set (13.5% of males vs. only 2.3% of females in the validation set wear sunglasses). Importantly, when examining the correlation between hair colors (actual class labels) and sunglasses, we observe a milder correlation between non-blond hair and sunglasses: 7.3% of non-blond haired wear sunglasses compared to only 1.7% of those with blond hair. Therefore, the pseudo-attribute predictor has likely learned to correlate blond males with sunglasses, resulting in the potential to amplify other (potentially spurious) correlations learned from the validation set while mitigating the targeted spurious correlations.

Waterbirds. In line with our observations on CelebA, as shown in Table 14, the groups identified by JTT are similar to those identified by EIIL, and the groups identified by SSA share similarities with the groups identified by SPARE, which requires less training. Specifically, JTT and EIIL focus on upweighting noisy and outlier examples, SSA upweights examples that may possess certain (spurious) features (i.e., yellow feathers), and SPARE prioritizes upweighting minority groups that do not share the spurious features with the majority groups.

I REPRODUCIBILITY

Each experiment was conducted on one of the following GPUs: NVIDIA A40 with 45G memory, NVIDIA RTX A6000 with 48G memory, and NVIDIA RTX A5000 with 24G memory.

Table 13: Comparison of groups found by different methods for CelebA.

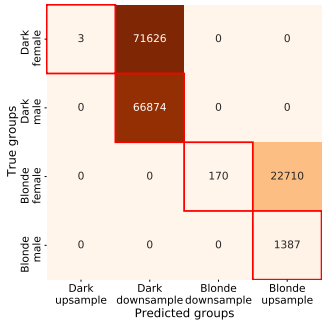
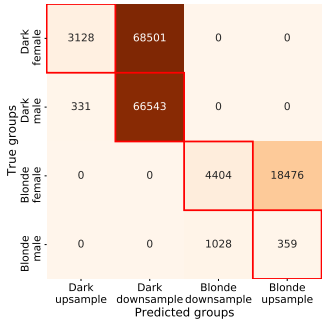
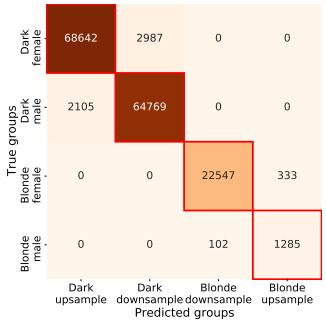
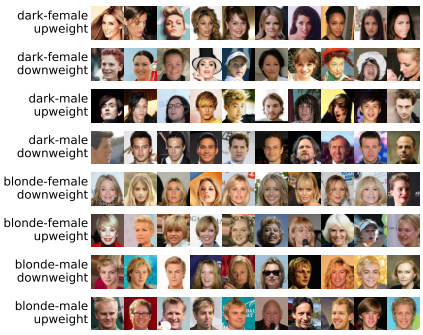
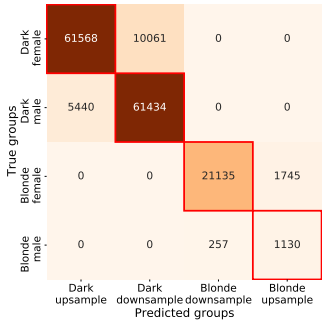
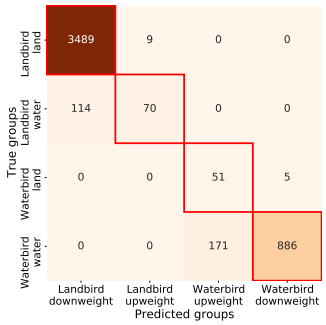
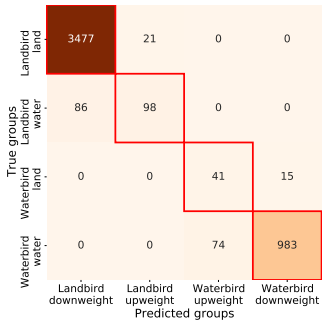
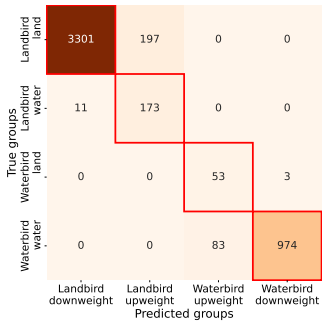
Inference (Cost)	Method	Samples	Confusion Matrix
	JTT (1 epoch)		
	EIIL (1 epoch)		
	SSA (53 epochs)		
	SPARE (1 epoch)		

Table 14: Comparison of groups found by different methods for Waterbirds.

Inference (Cost)	Method	Samples	Confusion Matrix
	JTT (60 epochs)		
	EIIL (1 epoch)		
	SSA (40 epochs)		
	SPARE (1 epoch)