Just Train Twice: Improving Group Robustness without Training Group Information

Evan Zheran Liu [*1] **Behzad Haghgoo** [*1] **Annie S. Chen** [*1] **Aditi Raghunathan** [1] **Pang Wei Koh** [1]
Shiori Sagawa [1] **Percy Liang** [1] **Chelsea Finn** [1]

Abstract

Standard training via empirical risk minimization (ERM) can produce models that achieve low error on average but high error on certain groups, especially in the presence of spurious correlations between the input and label. Prior approaches that achieve low worst-group error, like group distributionally robust optimization (group DRO) require expensive group annotations for each training point, whereas approaches that do not use such group annotations achieve worse worst-group performance. In this paper, we propose a simple two-stage approach, JTT, that minimizes the loss over a reweighted dataset (second stage) where we upweight training examples that are misclassified at the end of a few steps of standard training (first stage). Intuitively, this upweights points from groups on which standard ERM models perform poorly, leading to improved worst-group performance. On four image classification and natural language processing tasks with spurious correlations, we show that JTT closes 73% of the gap in worst-group accuracy between standard ERM and group DRO, while only requiring group annotations on a small validation set in order to tune hyperparameters.

1. Introduction

The standard approach of empirical risk minimization (ERM)—training machine learning models to minimize average training loss—can produce models that achieve low test error on average but still incur high error on certain groups of examples (Hovy & Søgaard, 2015; Blodgett et al., 2016; Tatman, 2017; Hashimoto et al., 2018; Duchi et al., 2019). These kinds of performance disparities across groups can be especially pronounced in the presence of *spurious correlations*. For example, in the task of classifying whether an online comment is toxic, the training data is often biased so that mentions of particular demographic identities (e.g., certain races or religions) are positively correlated with toxicity. Models trained via ERM then associate demographic mentions with toxicity and thus perform poorly on groups of examples in which the correlation does not hold, such as non-toxic comments mentioning a particular demographic (Borkan et al., 2019). Similar performance disparities due to spurious correlations have been reported in many other applications, including other language tasks, facial recognition, and medical imaging (Gururangan et al., 2018; McCoy et al., 2019; Badgeley et al., 2019; Sagawa et al., 2020a; Oakden-Rayner et al., 2020).

Following prior work, we formalize this setting by considering a set of pre-defined groups (e.g., corresponding to different demographics) and seeking models that have low worst-group error (Sagawa et al., 2020a). Previous approaches to this problem typically require annotations of the group membership of each training example (Sagawa et al., 2020a; Goel et al., 2020; Zhang et al., 2020). While these approaches have been successful at improving worst-group performance, the training group annotations that they require are often expensive to obtain; for example, in the toxicity classification task mentioned above, this would require annotating each comment with the demographic identities that are mentioned.

In this paper, we propose a simple algorithm, JTT (Just Train Twice), for improving the worst-group error without training group annotations, instead only requiring group annotations on a much smaller validation set to tune hyperparameters. JTT is composed of two stages: we first identify training examples that are misclassified by a standard ERM model, and we then train the final model by upweighting the examples identified in the first stage. Intuitively, this procedure exploits the observation that sufficiently-regularized ERM models tend to incur high worst-group training error. This makes selecting misclassified examples an effective heuristic for identifying examples from the worst-performing group, and upweighting such examples can yield

[*]Equal contribution [1]Department of Computer Science, Stanford University. Correspondence to: Evan Zheran Liu <evanliu@cs.stanford.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

models with better worst-group performance (Sagawa et al., 2020a).

We evaluate JTT on two image classification datasets with spurious correlations, Waterbirds (Wah et al., 2011) and CelebA (Liu et al., 2015) and two natural language processing datasets, MultiNLI (Williams et al., 2018) and CivilComments-WILDS (Borkan et al., 2019; Koh et al., 2021). We use the versions of Waterbirds, CelebA, and MultiNLI from Sagawa et al. (2020a), where in Waterbirds, the label *waterbird* or *landbird* spuriously correlates with water in the background; in CelebA, the label *blond* or *non-blond* spuriously correlates with binary gender; in MultiNLI, the label spuriously correlates with the presence of negation words. Our method outperforms ERM on all four datasets, with an average worst-group accuracy improvement of 15.9%, while maintaining competitive average accuracy (only 3.7% worse on average). Furthermore, despite having no group annotations during training, JTT closes 73% of the gap between ERM and group DRO, which uses complete group information on the training data.

We then empirically analyze JTT. First, we analyze the examples identified by JTT and show that JTT upweights groups on which standard ERM models perform poorly, such as waterbirds on land backgrounds of landbirds on water backgrounds. Second, we compare JTT with a related algorithm in the common framework of distributionally robust optimization (DRO) that minimizes the conditional value at risk (CVaR). CVaR DRO aims to train models that are robust to a wide range of potential distribution shifts without specific group annotations of training points by minimizing the worst-case loss over all subsets of the training set of a certain size (Duchi et al., 2019). This objective can be optimized by dynamically upweighting training examples with the highest losses in each minibatch (Levy et al., 2020). Though CVaR DRO and JTT share conceptual similarities—they both upweight training and do not require training group information—JTT empirically substantially outperforms CVaR DRO. We empirically find that one crucial difference between the two is that JTT upweights a static set of examples, while CVaR DRO dynamically re-computes which examples to update.

2. Related Work

In this paper, we focus on group robustness (i.e., training models that obtain good performance on each of a set of predefined groups in the dataset), though other notions of robustness are also studied, such as adversarial examples (Biggio et al., 2013; Szegedy et al., 2014) or domain generalization (Blanchard et al., 2011; Muandet et al., 2013). Approaches for group robustness fall into the two main categories we discuss below.

Robustness using group information. Several approaches leverage group information during training, either to combat spurious correlations or maintain good performance even when the groups representations change between training and testing. For example, Mohri et al. (2019); Sagawa et al. (2020a); Zhang et al. (2020) minimize the worst-group loss during training; Goel et al. (2020) synthetically expand the minority groups via generative modeling; Shimodaira (2000); Byrd & Lipton (2019); Sagawa et al. (2020b) reweight or subsample the majority and minority groups; and Cao et al. (2019; 2020) impose heavy Lipschitz regularization around minority points. These approaches substantially reduce worst-group error, but obtaining group annotations for the entire training set can be extremely expensive.

Another line of work studies worst-group performance in the context of fairness. Whereas the above works seek to improve the worst-group loss, this line of work explicitly tries to equalize loss across groups (Hardt et al., 2016; Woodworth et al., 2017; Pleiss et al., 2017; Agarwal et al., 2018; Khani et al., 2019).

Robustness without group information. We instead focus on the setting where group annotations are unavailable at training, and potentially only available during validation for hyperparameter tuning. This setting requires significantly lower annotation effort, as validation sets typically contain far fewer examples than training sets, but approaches for this setting achieve lower worst-group performance. Many approaches for this setting fall under the DRO framework, where models are trained to minimize the worst-case loss across all distributions in a ball around the empirical distribution (Ben-Tal et al., 2013; Lam & Zhou, 2015; Duchi et al., 2016; Namkoong & Duchi, 2017; Oren et al., 2019). Pezeshki et al. (2020) modify the dynamics of stochastic gradient descent to avoid learning spurious correlations. Sohoni et al. (2020) automatically identify groups based on clustering and improve robustness via approaches that use this learnt group information. Kim et al. (2019) propose an auditing scheme that searches for high-loss groups defined by a function within a pre-specified complexity class and postprocess the model to minimize discrepancies identified by the auditor. Another approach is to directly *learn* to reweight the training examples either using small amount of metadata (Shu et al., 2019), or automatically via meta learning (Ren et al., 2018).

The closest related approach to our work is Learning from Failure (LfF) (Nam et al., 2020), which simultaneously learns a pair of models. The first model is intentionally biased and tries to identify minority examples where the spurious correlation does not hold. The identified examples are upweighted while training the second model. This approach interleaves the updates of both models and requires

purposely biasing the first model. In contrast, our approach of JTT is simpler, though conceptually similar: we have a two-stage process where we attempt to identify minority points in the first stage and then upweight these points in the second stage without interleaved training. Empirically, despite its simplicity, JTT performs better than LfF.

3. Preliminaries

3.1. Problem Setup

We consider the setting where each example consists of an input $x \in \mathcal{X}$, a label $y \in \mathcal{Y}$, and a group $g \in \mathcal{G}$. We primarily consider the setting where each group $g = (a, y) \in \mathcal{G}$ is defined by the label y and a spurious attribute $a \in \mathcal{A}$ that spuriously correlates with the label (i.e., $\mathcal{G} = \mathcal{A} \times \mathcal{Y}$). For instance, in the Waterbirds dataset (Sagawa et al., 2020a), x is an image of a bird; y is whether the bird is a land or waterbird; a is whether the background is a land or water background; and there are 4 groups corresponding to $\{\text{landbird}, \text{waterbird}\} \times \{\text{land background}, \text{water background}\}$. In that dataset, y is correlated with a: landbirds are more likely to be pictured on land backgrounds, and similarly waterbirds are more likely to be on water backgrounds.

Our goal is to learn a model $f_\theta : \mathcal{X} \to \mathcal{Y}$, parameterized by $\theta \in \Theta$, that minimizes the *worst-group error*:

$$\max_{g \in \mathcal{G}} \mathbb{E}\left[\ell_{0-1}(x, y; \theta) \mid g\right], \tag{1}$$

where $l_{0-1}(x, y; \theta)$ is the 0-1 loss. This objective is one way of encoding the goal of learning a model that does not latch on to the spurious correlation. For example, in the Waterbirds dataset, if a model learns to use the spurious correlation—e.g., if it simply predicts that birds on land backgrounds must be landbirds—then it would do poorly on the groups of waterbirds on land and landbirds on water, and hence suffer high worst-group error.

In this work, we are interested in the setting where the spurious attribute a and consequently the group identity g are not available at training time, as annotating spurious attributes is typically expensive. However, we assume that we have access to a small validation set with annotations of the spurious attribute, which can be used to select model or algorithm hyperparameters.

3.2. Baseline Algorithms

Here, we describe three baseline algorithms that we use as comparisons in this paper. The first, empirical risk minimization, is the standard approach for training machine learning models, which seems to minimize the average error. The second, a distributionally robust optimization (DRO) method for minimizing the conditional value at risk (CVaR), seeks to minimize error over all "large enough" groups (Duchi et al., 2019), and is a natural approach to training models with low worst-group error; we will discuss later the relation between CVaR DRO and our proposed method, JTT. Finally, we also consider models trained with group DRO (Sagawa et al., 2020a), which—unlike ERM and CVaR DRO—uses training group annotations, and can therefore be considered as an oracle method that upper bounds the performance we might expect from methods that do not use training group annotations.

Empirical risk minimization (ERM). Given n training points $\{(x_1, y_1), \ldots (x_n, y_n)\}$, empirical risk minimization seeks to minimize the average training loss

$$J_{\text{ERM}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(x_i, y_i; \theta), \tag{2}$$

where $\ell(x, y; \theta) : \mathcal{X} \times \mathcal{Y} \times \Theta \to \mathbb{R}_+$ is a loss function (e.g., logistic loss).

Distributionally robust optimization of the conditional value at risk (CVaR DRO). Instead of minimizing the expected loss over the training distribution $\hat{P} = \frac{1}{n} \sum_{i=1}^{n} \delta_{(x_i, y_i)}$, as in ERM, distributionally robust learning algorithms define an uncertainty set $U(\hat{P})$ over distributions that are within some distance of $\hat{P}$, and then minimize the expected loss over the worst-case distribution in this uncertainty set (Duchi et al., 2019).

In this paper, we study a classic instance of this type of worst-case loss known as the conditional value at risk (CVaR) at level $\alpha \in (0, 1]$, which corresponds to an uncertainty set that contains all α-sized subpopulations of the training distribution (Rockafellar & Uryasev, 2000). The idea is that the worst loss over α-sized subpopulations upper bounds the worst-group loss for α similar to the size of each group. In practice, we treat α as a hyperparameter.

Concretely, the CVaR objective can be written as

$$J_{\text{CVaR}}(\theta, \alpha) = \sup_{q \in \Delta^n} \left\{ \sum_{i=1}^{n} q_i \ell(x_i, y_i; \theta) \text{ s.t. } \|q\|_\infty \leq \frac{1}{\alpha n} \right\}, \tag{3}$$

where Δ^n is the probability simplex in $\mathbb{R}^n$. It can also be expressed in terms of the inverse CDF of the loss $\ell(x, y; \theta)$ under the empirical training distribution $\hat{P}$,

$$J_{\text{CVaR}}(\theta, \alpha) = \frac{1}{\alpha} \int_{1-\alpha}^{1} \hat{F}^{-1}(u)\, du, \tag{4}$$

where $\hat{F}^{-1}(u)$ is the inverse CDF of $\ell(x, y; \theta)$ under $\hat{P}$. In other words, the CVaR objective is the average loss incurred by the α-fraction of training points that have the highest loss.

Group distributionally robust optimization (Group DRO). Group DRO (Sagawa et al., 2020a) uses training group annotations to define the uncertainty set as all possible mixtures of the group distributions $\hat{P}_g$, where $\hat{P}_g$ is the empirical distribution of training examples associated with a particular group $g \in \mathcal{G}$. Concretely, assume that we observe n training points $\{(x_1, y_1, g_1), \dots (x_n, y_n, g_n)\}$. The group DRO objective can then be written as

$$J_{\text{group-DRO}}(\theta) = \max_{g \in \mathcal{G}} \frac{1}{n_g} \sum_{g_i = g} \ell(x_i, y_i; \theta) \tag{5}$$

where n_g is the number of training points with group $g_i = g$. Note that training models via group DRO requires group annotations g_i over the training data (though the models do not need to see g at test time). In contrast, the other methods discussed in this paper focus on the setting where we do not have access to training group annotations.

4. JTT: Just Train Twice

We now present JTT, a simple two-stage approach that does not require group annotations at training time. In the first stage, we train an initial model and identify examples with high training loss. Then, in the second stage, we train a final model while upweighting these examples.

Stage 1 (identification). The key empirical observation that JTT builds on is that sufficiently low complexity ERM models tend to fit groups with easy-to-learn spurious correlations (e.g., landbirds on land and waterbirds on water in the Waterbirds dataset), but not groups that do not exhibit the same correlation (e.g., waterbirds on land) (Sagawa et al., 2020a). We therefore use the simple heuristic of first training an *identification model* $\hat{f}_{\text{id}}$ via ERM and then identifying an *error set* E of training examples that $\hat{f}_{\text{id}}$ misclassifies:

$$E = \{(x_i, y_i) \text{ s.t. } \hat{f}_{\text{id}}(x_i) \neq y_i\}. \tag{6}$$

Stage 2 (upweighting). Next, we train a final model $\hat{f}_{\text{final}}$ by upweighting the points in the error set E identified in step one:

$$J_{\text{up-ERM}}(\theta, E) = \left(\lambda_{\text{up}} \sum_{(x,y) \in E} \ell(x, y; \theta) + \sum_{(x,y) \notin E} \ell(x, y; \theta) \right), \tag{7}$$

where $\lambda_{\text{up}} \in \mathbb{R}_+$ is a hyperparameter. The hope is that if the examples in the error sets come from challenging groups, such as those where the spurious correlation does not hold, then upweighting them will lead to better worst-group performance.

Algorithm 1 JTT training

Input: Training set $\mathcal{D}$ and hyperparameters T and λ_{up}.
Stage one: identification
1. Train $\hat{f}_{\text{id}}$ on $\mathcal{D}$ via ERM for T steps (Equation 2).
2. Construct the error set E of training examples misclassified by $\hat{f}_{\text{id}}$ (Equation 6).
Stage two: upweighting identified points
3. Construct upsampled dataset $\mathcal{D}_{\text{up}}$ containing examples in the error set λ_{up} times and all other examples once.
4. Train final model $\hat{f}_{\text{final}}$ on $\mathcal{D}_{\text{up}}$ via ERM (Equation 2).

Practical implementation. Overall, training JTT is summarized in Algorithm 1. In practice, to restrict the capacity of the identification model, we only train it for T steps, where T is a hyperparameter in line 1. This prevents it from potentially overfitting the training data and yielding an empty error set. To implement the upweighted objective (7), we simply upsample the examples from the error set by λ_{up} (line 3) and train the final model on the upsampled data (line 4). Specifically, in each epoch of training, we sample each example from the error set λ_{up} times and all other examples only once.

We tune the algorithm hyperparameters (the number of training epochs T for the identification model $\hat{f}_{\text{id}}$, and the upweight factor λ_{up}) and both identification and final model hyperparameters (e.g., the learning rate and ℓ_2 regularization strength) based on the worst-group error of the final model $\hat{f}_{\text{final}}$ on the validation set. In our experiments, we share the same hyperparameters and architecture between the identification and final models, outside of the early stopping T of the identification model, and we sometimes find it helpful to learn them with different optimizers. Note that setting the upweight factor λ_{up} to 1 recovers ERM, so JTT should perform at least as well as ERM, given a sufficiently large validation set. We describe full training details in Appendix A.

5. Experiments

In our experiments, we first demonstrate that JTT substantially improves worst-group performance compared to standard ERM models (Section 5.2). We also show that it recovers a significant fraction of the performance gains yielded by group DRO, which, as discussed in Section 3, is an oracle that relies on group annotations on training examples. We then present empirical analysis of JTT, including the analysis of the error set (Section 5.3), exploration on the role of the validation set (Section 5.4), and comparison with CVaR DRO (Section 5.5).

Figure 1. Examples from the tasks we evaluate on. The spurious attribute a is correlated with the label y on the training data.

Method	Group labels in train set?	Waterbirds		CelebA		MultiNLI		CivilComments-WILDS	
		Avg Acc.	Worst-group Acc.	Avg Acc.	Worst-group Acc.	Avg Acc.	Worst-group Acc.	Avg Acc.	Worst-group Acc.
ERM	No	97.3%	72.6%	95.6%	47.2%	82.4%	67.9%	92.6%	57.4%
CVaR DRO (Levy et al., 2020)	No	96.5%	69.5%	82.4%	64.4%	82.0%	68.0%	92.5%	60.5%
LfF (Nam et al., 2020)	No	97.5 %	75.2%	86.0%	70.6%	80.8%	70.2%	92.5%	58.8%
JTT (Ours)	No	93.6%	**86.0%**	88.0%	**81.1%**	80.4%	**72.3%**	91.1%	**69.3%**
Group DRO (Sagawa et al., 2020a)	Yes	93.5%	91.4%	92.9%	88.9%	81.4%	77.7%	88.9%	69.9%

Table 1. Average and worst-group test accuracies of models trained via JTT and baselines. JTT substantially improves worst-group accuracy relative to ERM and CVaR DRO and outperforms LfF (Nam et al., 2020), a recently proposed algorithm for improving worst-group accuracy without group annotations. We also compare with group DRO, an oracle that assumes group annotations. JTT recovers a significant fraction of the gap in worst-group accuracy between ERM and group DRO.

5.1. Setup

We study four datasets in which prior work has observed poor worst-group performance due to spurious correlations (Figure 1). Full details about these datasets are in Appendix B.

- **Waterbirds** (Wah et al., 2011; Sagawa et al., 2020a): The task is to classify images of birds as "waterbird" or "landbird", and the label is spuriously correlated with the image background, which is either "land" or "water."
- **CelebA** (Liu et al., 2015): We consider the task from Sagawa et al. (2020a) of classifying the hair color of celebrities as "blond" or "not blond." The label is spuriously correlated with gender, which is either "male" or "female."
- **MultiNLI** (Williams et al., 2018): Given a pair of sentences, the task is to classify whether the second sentence is entailed by, neutral with, or contradicts the first sentence. We use the spurious attribute from Sagawa et al. (2020a), which is the presence of negation words in the second sentence; due to the artifacts from the data collection process, contradiction examples often include negation words.
- **CivilComments-WILDS** (Borkan et al., 2019; Koh et al., 2021): The task is to classify whether an online comment is toxic or non-toxic, and the label is spuriously correlated with mentions of certain demographic identities (male, female, White, Black, LGBTQ, Muslim, Christian, and other religion). We use the evaluation metric from Koh et al. (2021), which defines 16 overlapping groups $(a, \textit{toxic})$ and $(a, \textit{non-toxic})$ for each of the above 8 demographic identities a, and report the worst-group performance over these groups.

Points of comparison. We aim to answer two main questions: (1) How does JTT compare with other approaches that also do not use training group information? (2) How does JTT compare with approaches that *do* use training group information?

To answer the first question, we compare JTT with ERM, CVaR DRO (see Section 3), and a recently proposed approach called Learning from Failure (LfF) (Nam et al., 2020). To answer the second question, we compare JTT with group DRO (Sagawa et al., 2020a), an oracle that uses training group annotations. Note that on CivilComments, group DRO cannot be directly applied on the 16 defined groups, since it is not designed for overlapping groups. Instead, our group DRO minimizes worst-group loss over 4 groups (a, y), where the spurious attribute a is a binary indicator of whether any demographic identity is mentioned and the label y is *toxic* or *non-toxic*. We tune the hyperparameters of all approaches based on worst-group performance on a small validation set with group annotations.

Dataset	Worst-group Recall	Worst-group Precision	Worst-group Empirical Rate
Waterbirds	87.5%	19.1%	1.2%
CelebA	94.7%	9.4%	0.9%
MultiNLI	67.1%	2.2%	1.0%
CivilComments	96.9%	7.8%	0.9%

Table 2. The precision and recall of the worst-group examples (i.e., the group with lowest validation accuracy) belonging to JTT's error set. The error set includes a high fraction of the worst-group examples and includes them at a much higher rate than they occur in the training data.

Group	Enrichment	ERM test acc.
(land background, waterbird)	15.92x	72.6%
(water background, landbird)	6.97x	73.3%
(water background, waterbird)	2.40x	96.3%
(land background, landbird)	0.02x	99.3%

Table 3. Waterbirds error set breakdowns.

5.2. Main Results

Table 4 reports the average and worst-group accuracies of all approaches. Compared to other approaches that do not use training group information, JTT consistently achieves higher worst-group accuracy on all 4 datasets. Additionally, JTT performs well even relative to approaches that use training group information. In particular, JTT recovers a significant portion of the gap in worst-group accuracy between ERM and group DRO, closing 73% of the gap on average. As a caveat, we note that simple label balancing also achieves comparably worst-group accuracy to group DRO on CivilComments.

JTT's worst-group accuracy improvements come at only a modest drop in average accuracy, averaging only 3.8% worse than the highest average accuracy on each dataset. This drop is consistent with Sagawa et al. (2020a), which observes a tradeoff between average and worst-group accuracies.

5.3. Error set analysis

We find it surprising that just a small amount of group information on the validation set can allow JTT to achieve high worst-group accuracy with no knowledge of the groups on the training set. We now probe into how JTT achieves such high worst-group accuracy. In order to perform this analysis, we use the group annotations on the training data to closely examine what examples are upweighted in the error set identified in the first step of JTT.

To start, we define the worst group as the group on which the standard ERM model achieves the lowest test accuracy, when tuned for worst-group validation accuracy. We analyze how well the error set captures this worst-case group.

Group	Enrichment	ERM test acc.
(male, blond)	10.44x	47.2%
(female, blond)	5.42x	89.1%
(male, non-blond)	0.32x	99.3%
(female, non-blond)	0.01x	95.1%

Table 4. CelebA error set breakdowns.

Group	Enrichment	ERM test acc.
(negation, neutral)	2.2x	67.9%
(no negation, contradiction)	1.35x	77.0%
(negation, entailment)	1.14x	80.4%
(no negation, neutral)	1.07x	81.8%
(no negation, entailment)	0.73x	86.1%
(negation, contradiction)	0.19x	94.5%

Table 5. MultiNLI error set breakdowns.

To do this, we measure *precision*, the fraction of examples in the error set that belong to the worst-case group, *recall*, the fraction of the worst-case group examples that are included in the error set, and the *empirical rate*, the rate at which the worst-case group examples appear in the training data. As reported in Table 2, we observe that the error set contains worst-group examples at a much higher rate (precision) than they appear in the training dataset (empirical rate). Worst-group examples appear in the error set 2.2x to 15.9x more frequently in the error set than in the training data, across the 4 datasets. In other words, the worst-case group is significantly *enriched* in the error set compared to the training dataset, which may explain why JTT has much better worst-group performance over ERM. Additionally, the error set has high worst-group recall, ranging from 67.1% to 96.9% and averaging to 86.4% across the 4 datasets. Together, these results indicate that the worst-case group is included in the error set at both reasonable precision and recall.

Empirically, ERM performs poorly on several groups, not just on a single worst group. We therefore next examine what other groups the examples in the error set belong to, beyond the worst group. For each group, we compute two metrics: (i) *enrichment* defined as how much more frequently examples from a group appear in the error set than in the training data (i.e., the precision of the group divided by the empirical rate of the group); (ii) the test accuracy that ERM achieves on this group, when tuned for worst-group validation accuracy.

Tables 3 to 5 and Table 7 in Appendix B.4 report these results for Waterbirds, CelebA, MultiNLI, and CivilComments respectively. We observe that the enrichment roughly inversely correlates with ERM's test accuracy on that group: examples from low performance groups are included at high rates in the error set relative to the empirical rate. This may help JTT to perform better across all groups that ERM

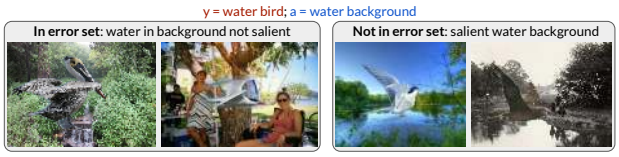

Figure 2. Randomly selected examples from the Waterbirds majority group with label water bird and spurious attribute water background. The spurious attribute (i.e., the water in the background) in examples in JTT's error set is far less salient than in examples not in the error set.

performs poorly on, which in turn improves worst-group accuracy.

While groups that ERM performs well on typically have low enrichments, the waterbirds on water backgrounds group has both high ERM test accuracy and high enrichment. We find the high enrichment particularly counter-intuitive, since the spurious correlation between the background (the spurious attribute) and the label seems to hold for this group. To understand why this occurs, we visually examine randomly selected examples from this group, both in and not in the error set, shown in Figure 2. We observe that the examples in the error set which are annotated to have water background actually have very little water, and are more reminiscent of land backgrounds. In contrast, the random examples *not* in the error set tend to have backgrounds in which water is highly prominent. In other words, the error set seems to include examples that resemble the worst-case group (waterbirds on land backgrounds), which is likely to be helpful for learning *not* to rely on the spurious correlation, and hence, for improving worst-group performance.

5.4. Hyperparameter Tuning and the Role of the Validation Set

In all the experiments, we tune the algorithm hyperparameters and model hyperparamters (such as early stopping) based on the worst-group accuracy on the validation set. In general, across all methods, we found hyperparameter tuning in this fashion to be critical, and this requires group annotations on the validation set. This is consistent with prior work showing that reweighting methods like JTT, LfF, and Group DRO require appropriate capacity control via techniques like early stopping or strong ℓ_2 regularization (Byrd & Lipton, 2019; Sagawa et al., 2020a). Note that if we instead tuned hyperparameters based on average validation accuracy (without requiring group annotations), typically, we would end up with large models with very low complexity control as such models tend to work best for average accuracy (Zhang et al., 2017).

In addition to early stopping and ℓ_2-regularization strength, JTT has two algorithm hyperparameters: the number of epochs to train the identification model T and the upweight factor λ_{up}. These hyperparameters can have a significant effect on the performance of JTT. For example, if T is too high and the identification model is otherwise unregularized, it could perfectly fit the training set, yielding an empty set and making JTT identical to ERM. Figure 3 shows how the worst-group accuracy of JTT's final model changes as we vary T between 20 and 100 epochs on Waterbirds. Worst-group accuracy is high when T is between 40 and 60, but drops when T is too small or too large.

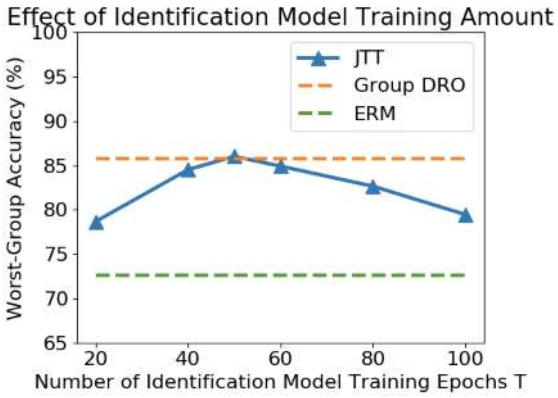

Figure 3. Effect of number of epochs of identification model training in JTT. Worst-group test accuracy is high for T between 40 and 60 epochs, but degrades when T becomes too small or large, which results in less informative error sets.

Reducing the size of the validation set. In our experiments, we used the default validation sets provided with each of the datasets. We ran additional preliminary experiments that suggest it could be possible to achieve similar results using much smaller validation sets, which would reduce the cost of obtaining group annotations on those sets. For example, cutting the validation set to 10% of its size for Waterbirds and CelebA gave almost identical worst-group performance for JTT, with an important caveat: we only used this smaller validation set for selecting model hyperparameters and early stopping, but not for selecting the JTT hyperparameters T and λ_{up}. For convenience, we used the same value of T as selected from the full validation set, and we heuristically set the upweight factor λ_{up} to weight the error set equally to the original training dataset. This heuristic seemed to be effective, but further work will be needed to rigorously establish the size of the validation set needed for hyperparameter tuning, as well as the effectiveness of any similar heuristics for selecting the algorithm hyperparameters.

5.5. Comparison with CVaR DRO

In this section, we explore the relation between JTT and the CVaR objective in Equation 3. Recall that the CVaR objective is the average loss incurred by the α-fraction of training

examples with the highest loss. We can view minimizing this objective as effectively upweighting this α-fraction of examples while ignoring the remaining examples. In this way, JTT is conceptually similar to CVaR DRO: both algorithms upweight training points with high loss, without requiring group annotations of training points. However, their empirical performance is widely different: CVaR DRO offers almost no gains in worst-case accuracy over ERM, while JTT offers substantial gains. One key difference is that in JTT, the set of points that get upweighted E is computed once during stage 1, and then held fixed. In contrast, minimizing the CVaR objective involves dynamically computing the α-subset of points with the highest loss at each step, upweighting them and updating the model, and then repeating to update the α-subset. As we show next, ablating this key difference from JTT substantially degrades worst-group performance.

Dynamically computing the error set in JTT lowers accuracy. We start by observing that the performance of JTT drops when we dynamically recompute the error set E, instead of only computing E once using the identification model. Concretely, we study a variant of JTT on the Waterbirds dataset: as usual, we first train an identification model for $T = 50$ epochs, but then while training the final model, every K epochs, we dynamically update the error set E as the errors of the final model over the training set. Setting K to be ∞—which means that we only compute the error set E once after training the identification model for T epochs—recovers standard JTT. On the other hand, lowering K makes the algorithm more similar to minimizing CVaR, since this more frequently updates the upweighted set to be the examples with higher loss under the current model, instead of the examples with higher loss under the static identification model.

Table 6 shows the results as we vary K between 10, 20, 30, and 50 epochs on Waterbirds. At high values of K, where the error set remains fixed for many epochs, both average and worst-group accuracies are high. However, as K decreases, the average and worst-group accuracies drop sharply. It is unclear why the accuracies at lower K are significantly worse than ERM and CVaR DRO performance, though this could be due to other differences between minimizing CVaR and this variant of JTT: for example, JTT only upweights misclassified training examples, the number of which can vary from epoch to epoch, whereas CVaR DRO always focuses on an α-fraction of training examples. These results show that at least on Waterbirds, holding the error set fixed appears to be critical for JTT.

Which examples does CVaR upweight? The analysis above suggests that the relatively poor worst-group performance of CVaR DRO might stem from how it dynamically computes which examples to upweight. We further study the behavior of CVaR DRO by analyzing the examples that it upweights. Concretely, throughout CVaR DRO training, we periodically identify the α-fraction of training examples with the highest loss and measure the worst-group precision and recall, where the worst group is defined as the group on which ERM achieves the lowest test accuracy.

	Waterbirds	
Epochs per update (K)	Average Acc.	Worst-group Acc.
10	82.3%	25.7%
20	94.1%	57.8%
30	93.3%	88.3%
50	94.5%	86.5%

Table 6. Effect of dynamically computing JTT's error set on Waterbirds. We first train the identification model for $T = 50$ epochs, as usual. Then, we dynamically update the error set using the final model after every K epochs of training the final model. Lower values of K have significantly lower accuracies.

Figure 4 shows the results using the value of α achieving the highest worst-group validation accuracy: $\alpha = 0.1$ on Waterbirds, $\alpha = 0.00852$ on CelebA, and $\alpha = 0.5$ on MultiNLI. On Waterbirds, the worst-group examples (which comprise approximately 1% of the training set) make up 19% of the error set for JTT, whereas they oscillate between 2% and 10% of the worst-α fraction for CVaR DRO. As a result, JTT consistently upweights nearly 90% of the worst-group examples, whereas CVaR DRO oscillates between upweighting the worst group and the other groups, upweighting as little as 20% of the examples at some points during training. On CelebA, CVaR DRO upweights the worst-group examples with slightly higher precision than JTT, but α is much smaller than the size of the error set; as a result, JTT upweights nearly 95% of the worst-group examples, whereas CVaR DRO only upweights 13% of them. On MultiNLI, the worst group steadily gets less and less upweighted for CVaR DRO, whereas JTT upweights it at a higher rate, though it still only comprises 2% of the error set for JTT.

These results suggest that the CVaR objective might be overly conservative where the α-fraction of examples with highest loss often include many examples from other groups. Furthermore, the set of examples varies widely across different iterations of training. In contrast, JTT upweights a fixed set of points. Empirically, we find that this allows JTT to successfully use the worst-group accuracy on a small validation set to identify error sets that improve accuracy on groups we care about.

6. Discussion

In this work, we presented Just Train Twice (JTT), a simple algorithm that can substantially improves worst-group

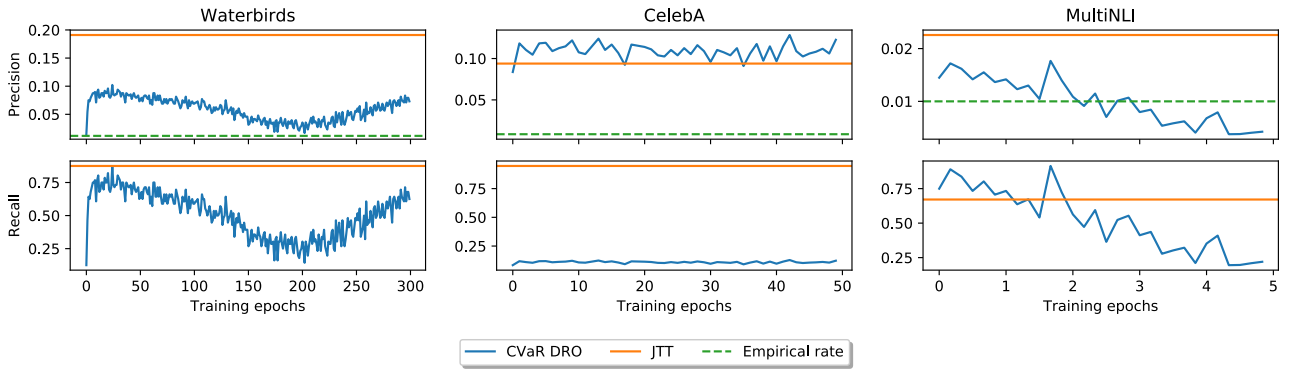

Figure 4. The composition of the CVaR set (the α-fraction of training examples with the highest loss) as training progresses for CVaR DRO models. In these plots, the worst group is defined as the group with the lowest test accuracy under the ERM model. For each dataset, the top plot shows the worst-group precision: the fraction of the CVaR set that belongs to the worst group (blue), with the analogous fraction of the JTT error set (orange) and the overall training data (green) provided for comparison. The bottom plot shows the worst-group recall: the fraction of total worst-group examples that are in the respective sets. For Waterbirds and MultiNLI, the CVaR set is less enriched for the worst group compared to JTT. For CelebA, it is slightly more enriched, but α is much smaller than the size of the JTT error set, so it only upweights a small fraction of the worst group.

performance without requiring expensive group annotations during training. We conclude by discussing several directions for future work.

First, a better theoretical understanding of when and why JTT works would help us to refine and further develop methods for training models that are less susceptible to spurious correlations. For example, it would be useful to understand why early-stopped ERM models (as in the identification models used by JTT) seem to consistently latch onto the spurious correlations in our datasets, and why it seems to be important to fix the upweighted set instead of dynamically recomputing it, as in CVaR DRO.

Second, JTT and many prior methods on robustness without group information all rely on a validation set that is representative of the distribution shift or annotated with group information. While these annotations are significantly cheaper that labeling the entire training set, it still requires the practitioner to be aware of any spurious correlations and define groups accordingly. Doing so may be notably difficult in real-world applications. Therefore, this leaves open the question of whether methods can perform well with mis-specified groups or no group annotations whatsoever.

Finally, while our experiments focus on group robustness in the presence of spurious correlations, JTT is not specifically tailored to spurious correlations. Given JTT's simplicity, it would be straightforward to experiment with JTT to see if it might improve performance under different types of distribution shifts, such as in domain generalization settings (Blanchard et al., 2011; Muandet et al., 2013).

Reproducibility. Our code is publicly available at https://github.com/anniesch/jtt.

Acknowledgements

This work was supported by NSF Award Grant No. 1805310 and in part by Google. EL is supported by a National Science Foundation Graduate Research Fellowship under Grant No. DGE-1656518. AR is supported by a Google PhD Fellowship and Open Philanthropy Project AI Fellowship. SS is supported by a Herbert Kunzel Stanford Graduate Fellowship.

References

Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., and Wallach, H. A reductions approach to fair classification. In *International Conference on Machine Learning (ICML)*, pp. 60–69, 2018.

Badgeley, M. A., Zech, J. R., Oakden-Rayner, L., Glicksberg, B. S., Liu, M., Gale, W., McConnell, M. V., Percha, B., Snyder, T. M., and Dudley, J. T. Deep learning predicts hip fracture using confounding patient and healthcare variables. *npj Digital Medicine*, 2, 2019.

Ben-Tal, A., den Hertog, D., Waegenaere, A. D., Melenberg, B., and Rennen, G. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59:341–357, 2013.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. In *Joint European*

conference on machine learning and knowledge discovery in databases, pp. 387–402, 2013.

Blanchard, G., Lee, G., and Scott, C. Generalizing from several related classification tasks to a new unlabeled sample. In *Advances in neural information processing systems*, pp. 2178–2186, 2011.

Blodgett, S. L., Green, L., and O'Connor, B. Demographic dialectal variation in social media: A case study of African-American English. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1119–1130, 2016.

Borkan, D., Dixon, L., Sorensen, J., Thain, N., and Vasserman, L. Nuanced metrics for measuring unintended bias with real data for text classification. In *World Wide Web (WWW)*, pp. 491–500, 2019.

Byrd, J. and Lipton, Z. What is the effect of importance weighting in deep learning? In *International Conference on Machine Learning (ICML)*, pp. 872–881, 2019.

Cao, K., Wei, C., Gaidon, A., Arechiga, N., and Ma, T. Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Cao, K., Chen, Y., Lu, J., Arechiga, N., Gaidon, A., and Ma, T. Heteroskedastic and imbalanced deep learning with adaptive regularization. *arXiv preprint arXiv:2006.15766*, 2020.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Association for Computational Linguistics (ACL)*, pp. 4171–4186, 2019.

Duchi, J., Glynn, P., and Namkoong, H. Statistics of robust optimization: A generalized empirical likelihood approach. *arXiv*, 2016.

Duchi, J., Hashimoto, T., and Namkoong, H. Distributionally robust losses against mixture covariate shifts. https://cs.stanford.edu/~thashim/assets/publications/condrisk.pdf, 2019.

Goel, K., Gu, A., Li, Y., and Ré, C. Model patching: Closing the subgroup performance gap with data augmentation. *arXiv preprint arXiv:2008.06775*, 2020.

Gururangan, S., Swayamdipta, S., Levy, O., Schwartz, R., Bowman, S., and Smith, N. A. Annotation artifacts in natural language inference data. In *Association for Computational Linguistics (ACL)*, pp. 107–112, 2018.

Hardt, M., Price, E., and Srebo, N. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3315–3323, 2016.

Hashimoto, T. B., Srivastava, M., Namkoong, H., and Liang, P. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning (ICML)*, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Hovy, D. and Søgaard, A. Tagging performance correlates with age. In *Association for Computational Linguistics (ACL)*, pp. 483–488, 2015.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, pp. 448–456, 2015.

Khani, F., Raghunathan, A., and Liang, P. Maximum weighted loss discrepancy. *arXiv preprint arXiv:1906.03518*, 2019.

Kim, M. P., Ghorbani, A., and Zou, J. Multiaccuracy: Black-box post-processing for fairness in classification. In *Association for the Advancement of Artificial Intelligence (AAAI)*, pp. 247–254, 2019.

Kingma, D. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., Lee, T., David, E., Stavness, I., Guo, W., Earnshaw, B. A., Haque, I. S., Beery, S., Leskovec, J., Kundaje, A., Pierson, E., Levine, S., Finn, C., and Liang, P. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021.

Lam, H. and Zhou, E. Quantifying input uncertainty in stochastic optimization. In *2015 Winter Simulation Conference*, 2015.

Levy, D., Carmon, Y., Duchi, J. C., and Sidford, A. Large-scale methods for distributionally robust optimization. *arXiv preprint arXiv:2010.05893*, 2020.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3730–3738, 2015.

Loshchilov, I. and Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017.

McCoy, R. T., Pavlick, E., and Linzen, T. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Association for Computational Linguistics (ACL)*, 2019.

Mohri, M., Sivek, G., and Suresh, A. T. Agnostic federated learning. In *International Conference on Machine Learning (ICML)*, pp. 4615–4625, 2019.

Muandet, K., Balduzzi, D., and Schölkopf, B. Domain generalization via invariant feature representation. In *International Conference on Machine Learning (ICML)*, pp. 10–18, 2013.

Nam, J., Cha, H., Ahn, S., Lee, J., and Shin, J. Learning from failure: Training debiased classifier from biased classifier. *arXiv preprint arXiv:2007.02561*, 2020.

Namkoong, H. and Duchi, J. Variance regularization with convex objectives. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Oakden-Rayner, L., Dunnmon, J., Carneiro, G., and Ré, C. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, pp. 151–159, 2020.

Oren, Y., Sagawa, S., Hashimoto, T., and Liang, P. Distributionally robust language modeling. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

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

Pezeshki, M., Kaba, S.-O., Bengio, Y., Courville, A., Precup, D., and Lajoie, G. Gradient starvation: A learning proclivity in neural networks. *arXiv preprint arXiv:2011.09468*, 2020.

Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. On fairness and calibration. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5684–5693, 2017.

Ren, M., Zeng, W., Yang, B., and Urtasun, R. Learning to reweight examples for robust deep learning. In *International Conference on Machine Learning (ICML)*, 2018.

Rockafellar, R. T. and Uryasev, S. Optimization of conditional value-at-risk. *Journal of Risk*, 2:21–41, 2000.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020a.

Sagawa, S., Raghunathan, A., Koh, P. W., and Liang, P. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning (ICML)*, 2020b.

Shimodaira, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90:227–244, 2000.

Shu, J., Xie, Q., Yi, L., Zhao, Q., Zhou, S., Xu, Z., and Meng, D. Meta-Weight-Net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1919–1930, 2019.

Sohoni, N. S., Dunnmon, J. A., Angus, G., Gu, A., and Ré, C. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *arXiv preprint arXiv:2011.12945*, 2020.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

Tatman, R. Gender and dialect bias in YouTube's automatic captions. In *Workshop on Ethics in Natural Langauge Processing*, volume 1, pp. 53–59, 2017.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The Caltech-UCSD Birds-200-2011 dataset. Technical report, California Institute of Technology, 2011.

Williams, A., Nangia, N., and Bowman, S. A broad-coverage challenge corpus for sentence understanding through inference. In *Association for Computational Linguistics (ACL)*, pp. 1112–1122, 2018.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., and Brew, J. HuggingFace's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

Woodworth, B., Gunasekar, S., Ohannessian, M. I., and Srebro, N. Learning non-discriminatory predictors. In *Conference on Learning Theory (COLT)*, pp. 1920–1953, 2017.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations (ICLR)*, 2017.

Zhang, J., Menon, A., Veit, A., Bhojanapalli, S., Kumar, S., and Sra, S. Coping with label shift via distributionally robust optimisation. *arXiv preprint arXiv:2010.12230*, 2020.

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2017.