DISTRIBUTIONALLY ROBUST POST-HOC CLASSIFIERS UNDER PRIOR SHIFTS

Anonymous authors
Paper under double-blind review

ABSTRACT

The generalization ability of machine learning models degrades significantly when the test distribution shifts away from the training distribution. We investigate the problem of training models that are robust to shifts caused by changes in the distribution of class-priors or group-priors. The presence of skewed training priors can often lead to the models overfitting to spurious features. Unlike existing methods, which optimize for either the worst or the average performance over classes or groups, our work is motivated by the need for finer control over the robustness properties of the model. We present an extremely lightweight post-hoc approach that performs scaling adjustments to predictions from a pre-trained model, with the goal of minimizing a distributionally robust loss around a chosen target distribution. These adjustments are computed by solving a constrained optimization problem on a validation set and applied to the model during test time. Our constrained optimization objective is inspired from a natural notion of robustness to controlled distribution shifts. Our method comes with provable guarantees and empirically makes a strong case for distributional robust post-hoc classifiers.

1 INTRODUCTION

Distribution shift, a problem characterized by the shift of test distribution away from the training distribution, deteriorates the generalizability of machine learning models and is a major challenge for successfully deploying these models in the wild. We are specifically interested in distribution shifts resulting from changes in marginal class priors or group priors from training to test. This is often caused by a skewed distribution of classes or groups in the training data, and vanilla empirical risk minimization (ERM) can lead to models overfitting to spurious features. These spurious features seem to be predictive on the training data but do not generalize well to the test set. For example, the background can act as a spurious feature for predicting the object of interest in images, *e.g.*, camels in a desert background, water-birds in water background (Sagawa et al., 2020).

Distributionally robust optimization (DRO) (Ben-Tal et al., 2013; Duchi et al., 2016; Duchi & Namkoong, 2018; Sagawa et al., 2020) is a popular framework to address this problem which formulates a robust optimization problem over class- or group-specific losses. The common metrics of interest in the DRO methods are either the average accuracy or the worst accuracy over classes/groups (Menon et al., 2021; Jitkrittum et al., 2022; Rosenfeld et al., 2022; Piratla et al., 2022; Sagawa et al., 2020; Zhai et al., 2021; Xu et al., 2020; Kirichenko et al., 2022). However, these metrics only cover the two ends of the full spectrum of distribution shifts in the priors. We are instead motivated by the need to measure the robustness of the model at various points on the spectrum of distribution shifts.

To this end, we consider applications where we are provided a target prior distribution (that could either come from a practitioner or default to the uniform distribution), and would like to train a model that is robust to varying distribution shifts around this prior. Instead of taking the conventional approach of optimizing for either the average accuracy or the worst-case accuracy, we seek to maximize the minimum accuracy within a δ-radius ball around the specified target distribution. This strategy allows us to encourage generalization on a spectrum of controlled distribution shifts governed by the parameter δ. When $\delta = 0$, our objective is simply the average accuracy for the specified target priors, and when $\delta \to \infty$, it reduces to the model's worst-case accuracy, thus providing a natural way to interpolate between the two extreme goals of average and worst-case optimization.

To train a classifier that performs well on the prescribed distributionally robust objective, we propose a fast and extremely lightweight post-hoc method that learns scaling adjustments to predictions from a pre-trained model. These adjustments are computed by solving a constrained optimization problem on a validation set, and then applied to the model during evaluation time. A key advantage of our method is that it is able to reuse the same pretrained model for different robustness requirements by simply scaling the model predictions. This is contrast to several existing DRO methods that train all model parameters using the robust optimization loss (Sagawa et al., 2020; Piratla et al., 2022), which requires group annotations for the training data and requires careful regularization to make it work with overparameterized models (Sagawa et al., 2020). On the other hand, our approach only needs group annotations for a smaller held-out set and works by only scaling the model predictions of a pre-trained model at test time. Our method also comes with provable convergence guarantees. We apply our method on standard benchmarks for class imbalance and group DRO, and show that it compares favorably to the existing methods when evaluated on a range of distribution shifts away from the target prior distribution.

2 Background

We are primarily interested in two specific prior shifts for distributional robustness of classifiers. In this section, we briefly introduce the problem setting of the two prior shifts and set the notation.

Class-Level Prior Shifts. We are interested in a multi-class classification problem with instance space $\mathcal{X}$ and output space $[m] = \{1, \ldots, m\}$. Let $\mathcal{D}$ denote the underlying data distribution over $\mathcal{X} \times [m]$, the random variables of instance X and label Y satisfy that $(X, Y) \sim \mathcal{D}$. We define the conditional-class probability as $\eta_y(x) = \mathbb{P}(Y = y|X = x)$ and the class priors $\pi_y = \mathbb{P}(Y = y)$, note that $\pi_y = \mathbb{E}\left[\eta_y(x)\right]$. We use $u = [\frac{1}{m}, \ldots, \frac{1}{m}]$ to denote the uniform prior over m classes.

Our goal is then to learn a multi-class classifier $h : \mathcal{X} \to [m]$ that maps an instance $x \in \mathcal{X}$ to one of m classes. We will do so by first learning a scoring function $f : \mathcal{X} \to \Delta_m$ that estimates the conditional-class probability for a given instance, and construct the classifier by predicting the class with the highest score: $h(x) = \arg\max_{j\in[m]} f_j(x)$. We measure the performance of a scoring function using a loss function $\ell : [m] \times \Delta_m \to \mathbb{R}_+$ and measure the per-class loss using $\ell_i(f) := \mathbb{E}\left[\ell(y, f(x)) \,|\, y = i\right]$.

Let $\{(x_i, y_i)\}_{i=1}^n$ be a set of training data samples. The empirical estimate of training set prior is $\hat{\pi}_i := \frac{1}{n}\sum_{j\in[n]} \mathbf{1}(y_j = i)$ where $\mathbf{1}(\cdot)$ is the indicator function. In class prior shift, the class prior probabilities at test time shift away from $\hat{\pi}$. A special case of such class-level prior shifts includes class-imbalanced learning (Lin et al., 2017; Cui et al., 2019; Cao et al., 2019; Ren et al., 2020; Menon et al., 2021) where $\hat{\pi}$ is a long-tailed distribution while the class prior at test time is usually chosen to be the uniform distribution. Regular ERM tends to focus more on the majority classes at the expense of ignoring the loss of the minority classes. Recent work (Menon et al., 2021) uses temperature-scaled logit adjustment with training class priors to adapt the model for average class accuracy. Our method also applies post-hoc adjustments to model probabilities, but our goal differs from (Menon et al., 2021) as we care for varying distribution shifts around the uniform prior and the scaling adjustments are learned using a held-out set to optimize for a constrained robust loss.

Group-Level Prior Shifts. The notion of *groups* arises when each data point (x, y) is associated with some attribute $a \in A$ that is spuriously correlated with the label. This is used to form $m = |A| \times |Y|$ groups as the cross-product of $|A|$ attributes and $|Y|$ classes. The data distribution $\mathcal{D}$ is taken to be a mixture of m groups with mixture prior probabilities π, and each group-conditional distribution given by $\mathcal{D}_j$, $j \in [m]$. In this scenario, we have n training samples $\{(x_i, y_i)\}_{i=1}^n$ drawn i.i.d. from $\mathcal{D}$, with empirical group prior probabilities $\hat{\pi}$. For skewed group-prior probabilities $\hat{\pi}$, regular ERM is vulnerable to spurious correlations between the attributes and labels, and the accuracy degrades when the test data comes from a shifted group prior (*e.g.*, balanced groups). Domain-aware methods typically assume that the attributes are available for the training examples and optimize for the worst or average group loss (Sagawa et al., 2020; Piratla et al., 2022). However, recent work (Rosenfeld et al., 2022; Kirichenko et al., 2022) has observed that ERM on skewed group priors is able to learn core features (in addition to spurious features), and training a linear classifier on top of ERM learned features with a small balanced held-out set works quite well. Our proposed method is aligned with this recent line of work in assuming access to only a small held-out set with group annotations.

However, we differ in two aspects: **(i)** Our method is more lightweight and works by only scaling the model predictions post-hoc during test time, **(ii)** The scaling adjustments are learned to allow more control over the desired robustness properties than implicitly targeting the worst or average accuracies as in (Kirichenko et al., 2022).

Evaluation Metrics under Prior Shifts. Typical evaluation metrics used under prior shifts are:

- Mean: This evaluation metric assigns uniform weights for each class or group, measuring the average class- or group-level test accuracy.
- Worst: This evaluation metric measures the performance on the worst class or group, and fails to examine the effectiveness of proposed methods on the remaining classes.
- Stratified: To overcome the issue of worst-case metric, stratified metrics are sometimes used that divide the classes into three strata and report average accuracy in each stratum, specifically (i) Head: average accuracy on subsets of classes that contain more than a specified number (*e.g.*, 100) of training samples; (ii) Torso: average accuracy on classes that contain, *e.g.*, 20 to 100 samples; (iii) Tail: average accuracy on tail classes.

The *mean* and *worst* metrics are limited in that they probe the model robustness only on the two ends of the full spectrum. The *stratified* metric looks into three strata of classes, but the design of strata is more heuristic and doesn't allow for a principled interpolation between *mean* and *worst* metrics.

3 Distributional Robustness Objective under Prior Shifts

We assume that we are given a target prior distribution (that could either come from a practitioner or default to the uniform distribution), and our goal is to train a model that is robust to varying distribution shifts around this prior. To this end, we consider an objective that naturally captures this goal and is based on the weighted sum of class-level (or group-level) performances on the test data, *i.e.*, $\sum_{i\in[m]} g_i \text{Acc}_i$, where Acc_i indicates the accuracy of the class/group i:

$$\textbf{δ-worst accuracy:} \qquad \min_{g\in\Delta_m} \sum_{i\in[m]} g_i \text{Acc}_i, \quad \text{s.t. } D(g,r) \le \delta\,. \tag{1}$$

Here Δ_m denotes the $(m-1)$-dimensional probability simplex and $r \in \Delta_m$ is a specified reference (or target) distribution. The δ-worst accuracy seeks the worst-case g-weighted performance with the weights constrained to lie within the δ-radius ball (defined by the divergence $D : \Delta_m \times \Delta_m \to \mathbb{R}$) around the target distribution r. For uniform distribution $r = u$ and any choice of divergence D, it reduces to the mean accuracy for $\delta = 0$ and the worst accuracy for $\delta \to \infty$. The objective interpolates between these two extremes for other values of δ and captures our goal of optimizing for variations around target priors instead of more conventional objectives of optimizing for either the average accuracy at the target prior or the worst-case accuracy. The divergence constraint in the δ-worst objective is convex in g for several divergence functions (*e.g.*, f-divergence, Bregman divergence), and allows for efficient measurement as well (if the per-class accuracies are known) using standard packages such as CVXPY (Diamond & Boyd, 2016).

4 Robust Post-hoc Classifiers under Class Prior Shifts

In this section, we propose a **D**istributionally **RO**bust **P**o**S**t-hoc method, **DROPS**, which enables the reuse of a pre-trained model for different robustness requirements by simply scaling the model predictions. In alignment with the earlier δ-worst objective, we aim to learn a classifier h with scoring function f as follows:

$$\textbf{Goal:} \qquad \min_{f:\mathcal{X}\to\Delta_m} \max_{g\in\Delta_m} \sum_{i\in[m]} g_i\, \mathbb{P}_{X|Y=i}(f(X) \neq Y), \quad \text{s.t. } D(g,r) \le \delta\,. \tag{2}$$

4.1 Bayes-optimal Scorer

Recall that we measure the performance of a scoring function using a loss function $\ell : [m] \times \Delta_m \to \mathbb{R}_+$, and the per-class loss using $\ell_i(f) := \mathbb{E}\left[\ell(y, f(x)) \,|\, y = i\right]$. To facilitate theoretical analysis, we consider the scenario where the target distribution r is the uniform prior $u = [\frac{1}{m}, \ldots, \frac{1}{m}]$. We are then interested in minimizing the following robust objective optimization:

$$\min_{f:\mathcal{X}\to\Delta_m} \mathrm{DRL}(f;\delta) = \min_{f:\mathcal{X}\to\Delta_m} \max_{g\in\mathbb{G}(\delta)} \sum_{i=1}^{m} g_i\, \ell_i(f)\,, \quad (3)$$

where $\mathbb{G}(\delta) = \{g \in \Delta_m \,|\, D(g,u) \le \delta\}$ for some $\delta > 0$ and divergence function $D : \Delta_m \times \Delta_m \to \mathbb{R}_+$. We first derive the Bayes-optimal solution to the learning problem in equation 3.

Theorem 1 (Bayes-Optimal scorer)**.** *Suppose $\ell(y,z)$ is a proper loss that is convex in its second argument and $D(g,\cdot)$ is convex in g. Let $\delta > 0$ be such that $\mathbb{G}(\delta)$ is non-empty. The optimal solution to equation 3 takes the following form for some $g^* \in \mathbb{G}(\delta)$:*

$$f^*_y(x) \propto \frac{g^*_y}{\pi_y} \cdot \eta_y(x)\,.$$

The proof is provided in Appendix A.1 and would use in its intermediate step, the following standard result for cost-sensitive learning:

Lemma 2. *Suppose $\ell(y,z)$ is a proper loss and $\ell_i(f) = \mathbb{E}\left[\ell(y, f(x)) \,|\, y = i\right]$. For any fixed $g \in \Delta_m$, the following is the minimizer to the objective $\sum_{i=1}^m g_i \cdot \ell_i(f)$ over all measurable functions $f : \mathcal{X} \to \Delta_m$: $f^*_y(x) \propto \frac{g_y}{\pi_y} \cdot \eta_y(x)$.*

4.2 Post-Hoc Approach

Given a pre-trained model $\hat{\eta} : \mathcal{X} \to \Delta_m$ that estimates the conditional-class probability function η, we seek to approximate the Bayes-optimal classifier. To do so, we write the Lagrangian form of equation 3 as: $\mathcal{L}(f,g,\lambda) = \sum_{i=1}^m g_i \cdot \ell_i(f) - \lambda(D(g,u) - \delta)$. We next show the optimization w.r.t. the Lagrangian form has the following equivalent unconstrained problem, as specified in Proposition 1.

$$\min_{f:\mathcal{X}\to\Delta_m} \max_{g\in\mathbb{G}(\delta)} \min_{\lambda\ge 0} \mathcal{L}(f,g,\lambda)\,. \quad (4)$$

Proposition 1. *The equivalence of two optimization tasks:*

$$\min_{f:\mathcal{X}\to\Delta_m} \mathrm{DRL}(f;\delta) \iff \min_{f:\mathcal{X}\to\Delta_m} \max_{g\in\mathbb{G}(\delta)} \min_{\lambda\ge 0} \mathcal{L}(f,g,\lambda)\,.$$

To understand why the above two optimization tasks are equivalent, we point out that if the constraint is violated, then the minimizer over λ would yield an unbounded objective. The maximizer over g, as a result, will always choose a g that satisfies the constraint.

Build upon Proposition 1, one can then solve the equivalent min-max problem by alternating between a gradient-descent update on λ, an Exponentiated Gradient (EG)-ascent update (Kivinen & Warmuth, 1997) on g, and a full minimization over f: $\min_{f:\mathcal{X}\to\Delta_m} \sum_{i=1}^m g_i \cdot \ell_i(f)$, the optimal solution for which is given from Lemma 2 by $f^*_y(x) \propto \frac{g_y}{\pi_y} \cdot \eta_y(x)$. Thus, for the pre-trained model $\hat{\eta}$, we propose to make post-hoc adjustments over $\hat{\eta}$ to make the classifiers more robust to prior shifts. We construct the "post-shifted" classifier by predicting the class with the "post-shifted" highest score for each x:

$$\textbf{DROPS:} \quad h(x) \in \operatorname*{arg\,max}_{i\in[m]} \frac{g_i}{\hat{\pi}_i} \cdot \hat{\eta}_i(x)\,.$$

Intuitively, model prediction of class i is up-scaled if the class i is assigned with a large weight g_i, or is of a small prior $\hat{\pi}_i$.

An Empirical Alternative Given a sample x, the "post-shifted" classifier takes the form $h(x) \in \mathrm{argmax}_{i\in[m]} \frac{g_i}{\hat{\pi}_i} \cdot \hat{\eta}_i(x)$. The softmax activation function then maps the per-class mode prediction logit, denoted by $\mathrm{Logit}_i(x)$, into the corresponding probability. Specifically, for $i \in [m]$, we have $\hat{\eta}_i(x) = \frac{\exp(\mathrm{Logit}_i(x))}{\sum_{j\in[m]} \exp(\mathrm{Logit}_j(x))}$. Thus, an empirical alternative of the proposed DROPS is a logit-adjustment approach where, given a sample x, for fixed class weights $\frac{g_i}{\hat{\pi}_i}$, we have:

$$\textbf{DROPS:}\ h(x) \in \mathrm{argmax}_{y\in[m]} \frac{g_y}{\hat{\pi}_y} \cdot \hat{\eta}_y(x) \iff h(x) \in \mathrm{argmax}_{y\in[m]} \mathrm{Logit}_y(x) + \log(\frac{g_y}{\hat{\pi}_y})\,.$$

4.3 CONVERGENCE OF POST-HOC CLASSIFIER

In practice, we only have access to an empirical estimate of the Lagrangian form computed using a validation set $S = \{(\tilde{x}_1, \tilde{y}_1), \ldots, (\tilde{x}_n, \tilde{y}_n)\}$:

$$\hat{\mathcal{L}}(f, g, \lambda) = \sum_{i=1}^{m} g_i \cdot \hat{\ell}_i(f) - \lambda(D(g, u) - \delta)\,, \tag{5}$$

where $\hat{\ell}_i = \frac{1}{\hat{\pi}_i} \sum_{(x,y)\in S: y=i} \ell(y, f(x))$ and $\hat{\pi}_i = \frac{1}{n}|\{(x, y) \in S : y = i\}|$. We can then approximately solve the saddle-point problem in equation 4 by repeating the following three steps for T number of iterations:

Step 1: updating $\lambda^{(t)}$ Given the step size $\eta_\lambda > 0$, we perform gradient updates on λ through:

$$\lambda^{(t+1)} \;=\; \lambda^{(t)} - \eta_\lambda \nabla \mathcal{L}(f^{(t)}, g^{(t)}, \lambda^{(t)}) = \lambda^{(t)} - \eta_\lambda(\delta - D(g^{(t)}, u))\,.$$

For KL divergence, the updated $\lambda^{(t+1)}$ amounts to $\lambda^{(t+1)} \leftarrow \lambda^{(t)} - \eta_\lambda(\delta - \sum_i g_i^{(t)} \log(\frac{g_i^{(t)}}{u_i}))$. We can clip λ if the update falls out of the desired range: $\lambda^{(t+1)} \leftarrow \text{clip}_{[0,R]}\left(\lambda^{(t)} - \eta_\lambda(\delta - D(g, u))\right)$ for some $R > 0$.

Step 2: updating $g^{(t)}$ Given the step size $\eta_g > 0$, we perform gradient updates on g through:

$$g_i^{(t+1)} \leftarrow g_i^{(t)} \cdot \exp\left(\eta_g \cdot \nabla_{g_i} \hat{\mathcal{L}}(f^{(t)}, g^{(t)}, \lambda^t)\right).$$

To obtain the updates for g, we consider the fixed $\lambda^{(t)} \in \mathbb{R}_+$ and adopt a EG-style computation.

Step 3: scaling the predictions In this step, we apply the post-hoc shifts on the classifier's prediction through $f_y^{(t+1)}(x) \propto \frac{g_y^{(t+1)}}{\hat{\pi}_y} \cdot \hat{\eta}_y(x)$, in which we use the pre-trained estimate $\hat{\eta}$ of the conditional-class probabilities.

The final scorer returned is an average of scorers from each iteration: $\bar{f}(x) = \frac{1}{T}\sum_{t=1}^{T} f^{(t)}(x)$.

We now provide a convergence guarantee for this procedure. For a fixed pre-trained model $\hat{\eta} : \mathcal{X} \to \Delta_m$, we denote the class of post-hoc adjusted functions derived from it by

$$\mathcal{F} = \left\{ f : x \mapsto \left(\frac{\beta_1 \hat{\eta}_1(x)}{\sum_j \beta_j \hat{\eta}_j(x)}, \ldots, \frac{\beta_m \hat{\eta}_m(x)}{\sum_j \beta_j \hat{\eta}_j(x)} \right) \mid \beta \in \Delta_m \right\},$$

and we let $|\mathcal{F}|$ denote a measure of complexity of this function class.

Theorem 3. *Fix $\delta > 0$. Suppose $\ell(y, z)$ be a proper loss that is convex and L_ℓ-Lipschitz in its second argument, and $\ell(y, z) \le B_\ell$, for some $B_\ell > 0$. Suppose $D(g, u) \le B_D$, for some $B_D > 0$, and D is convex and L_D-Lipschitz in its first argument. Furthermore, suppose $\max_{y\in[m]} \frac{1}{\pi_y} \le Z$, for some $Z > 0$. Then when $n \ge 8Z\log(2m/\alpha)$, and we set $T = \mathcal{O}(n)$, $R = 2B_\ell Z/\delta$, $\eta_\lambda = \frac{R}{B_D\sqrt{n}}$ and $\eta_g = \frac{1}{2B_\ell Z + RL_D}\sqrt{\frac{\log(m)}{n}}$, we have with probability at least $1 - \alpha$ over draw of validation sample $S \sim \mathcal{D}^n$, the classifier returned $\bar{f}(x) = \frac{1}{T}\sum_{t=1}^{T} f^{(t)}(x)$ satisfies:*

$$\text{DRL}(\bar{f}; \delta) \;\le\; \min_{f:\mathcal{X}\to\Delta_m} \text{DRL}(f; \delta) \;+\; \mathcal{O}\left(\sqrt{\frac{\log(m|\mathcal{F}|/\alpha)}{n}} \;+\; \mathbb{E}_x\left[\|\hat{\eta}(x) - \eta(x)\|_1\right]\right).$$

The proof is provided in Appendix A.3. The complexity measure $|\mathcal{F}|$ can be further bounded using, for example, standard covering number arguments (Shalev-Shwartz & Ben-David, 2014).

Notice that the convergence to the optimal classifier is bounded by two terms: the first is a sample complexity term that goes to 0 as the number of validation samples $n \to \infty$; the second term measures how well the pre-trained model $\hat{\eta}$ is calibrated, i.e., how well its scores match the underlying conditional-class probabilities.

4.4 Robust Post-hoc Classifiers under Group Prior Shifts

We now introduce the variant of our approach to address group prior shifts. We are interested in the performance of the trained classifiers with additional attribute information available in a held-out validation set. Note that the settings of class and group prior shifts differ in the knowledge of group information at validation and test time. Thus, the first variant of DROPS under group prior shifts is the same as the class prior setting: DROPS completely ignores the per-example group information and post-shifts the model predictions using only class labels. We consider another variant where we have access to the attribute information during both validation and test. This scenario can actually arise in practice when the attribute information is readily available for test examples, such as device or sensor type in the case of data coming from different devices/sensors. In this case, the attribute-specific class priors take the form of $\pi_{a,i} = \mathbb{P}(y = i|a)$. DROPS can be naturally adapted to this setting by learning multiple sets of scaling adjustments, one for each attribute type, and using the scaling adjustment corresponding to the attribute of the test example at prediction time. We provide the discussion, and the form of the Bayes-optimal classifier for this setting in Appendix B.

5 Related Work

Class-Imbalanced Learning. Most work in class-imbalanced learning is typically interested in generalizing on a uniform class prior when the training data has a skewed or imbalanced class prior. Existing solutions to the class imbalanced learning problem could be categorized into three major lines: (1) Information augmentation methods, which make use of additional information such as open set data (Wei et al., 2021), adopt a transfer learning approach to enrich the representation on the tail classes (Liu et al., 2020; Yin et al., 2019), or use advanced data-augmentation techniques (Perez & Wang, 2017; Shorten & Khoshgoftaar, 2019); (2) Module improvement methods, *i.e.*, the decoupled training on the head/tail classes (Kang et al., 2019; Chu et al., 2020; Zhong et al., 2021), or through an ensemble way to make use of multiple networks with different expertise/concentration (Zhou et al., 2020; Guo & Wang, 2021; Wang et al., 2020); (3) The most related to our work are the class re-balancing based methods, which mitigate the impact of class-imbalanced data by adjusting the logits using the class prior (Menon et al., 2021), or modify the loss values by referring to the label frequency (Ren et al., 2020) and sample influence (Park et al., 2021), or align the distributions of the model prediction and a set of balanced validation set (Zhang et al., 2021). Label shift is a related problem setting where the training class prior is not so imbalanced but the class prior shifts during test time with $p(x|y)$ staying the same, and the goal is to mitigate the effect of this shift (Lipton et al., 2018; Azizzadenesheli et al., 2019; Alexandari et al., 2020). However, our goal differs from these methods as we are primarily interested in generalization at worst case variations around the target distribution.

Group Distributional Robustness. It has been observed that classifiers learned with regular ERM are vulnerable to spurious correlations between the attributes and labels, and tend to perform worse when the test data comes from a shifted group prior. Most methods for group distributional robustness are interested in the average or worst group performance. Several prior works utilize the group information at the training time (Sagawa et al., 2020; Piratla et al., 2022). Recent works consider a more practical setting where the classifier does not have access to the group information at the training time, *i.e.*., data re-balancing (Idrissi et al., 2022) or re-weighting high loss examples (Liu et al., 2021). It was also observed recently that ERM is able to learn features that can be reused for group robustness by training a linear classifier using a balanced held-out set (Kirichenko et al., 2022; Rosenfeld et al., 2022). Out work is in similar vein in assuming access to only a held-out set with group annotations, however our method is more lightweight and allows to optimize for varying worst-case perturbations around the target distribution of interest using post-hoc scaling of model predictions. Other related work includes CGD (Piratla et al., 2022) that proposes a learning paradigm for optimizing the average group accuracy, and Invariant risk minimization (IRM) (Arjovsky et al., 2019; Ahuja et al., 2020) that aims to learn core features using data from multiple environments for mitigating the spurious correlations.

6 EXPERIMENTS

In this section, we empirically demonstrate the effectiveness of our proposed method DROPS, for the tasks of class-imbalanced learning and group distributional robustness.

6.1 EXPERIMENTS ON CLASS-IMBALANCED LEARNING

We consider the class-imbalanced task to illustrate the robustness of DROPS under class prior shifts. For CIFAR-10, and CIFAR-100 datasets, we down-sample the number of samples for each class to simulate the class-imbalance as done in earlier works (Cui et al., 2019; Cao et al., 2019). We define the imbalance ratio as $\rho := \frac{\max_{y\in[m]} \pi_y}{\min_{y\in[m]} \pi_y}$ and experiment with $\rho \in [10, 50, 100]$.

To demonstrate the effectiveness of DROPS, we compare the performance of our proposed method with several popular class-imbalanced learning approaches, including: Cross-Entropy (CE) loss, Focal Loss (Lin et al., 2017), Class-Balanced (CB) loss (Cui et al., 2019), LDAM (Cao et al., 2019), Balanced-Softmax (Ren et al., 2020), and Logit-adjustment (Menon et al., 2021) on the original test data. All methods are trained with the same architecture (ResNet 18 (He et al., 2016)) with 5 random seeds, same data augmentation techniques, the same SGD optimizer with a momentum of 0.9 with Nesterov acceleration. All methods share the same initial learning rate of 0.1 and a piece-wise constant learning rate schedule of $[10^{-2}, 10^{-3}, 10^{-4}]$ at $[30, 80, 110]$ epochs, respectively. We use a batch size of 128 for all methods and train the model for 140 epochs. A balanced held-out validation set is utilized for hyper-parameter tuning. All baseline models are picked by referring to the mean performances on the validation set. DROPS obtains the optimal post-hoc shifts under a variety of δ_{train} parameters, which is the perturbation hyper-parameter δ in the Lagrangian of equation 4. We take the divergence D to be KL-divergence.

Table 1: Performance comparisons on class-imbalanced CIFAR datasets: mean $\pm$ std of averaged class accuracy, $\delta = 1.0$-worst case accuracy, and worst class accuracy of 5 runs are reported. Best performed methods in each setting are highlighted.

Method	***CIFAR-10* (Averaged)**			***CIFAR-100* (Averaged)**		
	$\rho = 10$	$\rho = 50$	$\rho = 100$	$\rho = 10$	$\rho = 50$	$\rho = 100$
Cross Entropy	86.51±0.17	75.74±0.97	69.28±0.94	55.59±0.50	43.39±0.76	37.96±0.27
Focal	75.34±4.65	58.55±3.34	46.23±2.11	51.52±0.93	32.68±0.75	26.53±0.74
LDAM	86.63±0.33	76.00±0.78	68.81±1.01	55.42±0.67	43.03±0.68	38.35±0.63
Balanced-Softmax	87.18±0.24	78.76±0.44	35.55±0.35	56.37±0.54	45.10±0.51	41.21±0.23
Logit-Adjust	89.01±0.30	82.60±0.32	77.49±0.41	59.18±0.30	47.19±1.33	43.03±0.94
Logit-Adj (post-hoc)	88.98±0.29	82.48±0.75	77.94±0.50	59.33±0.51	47.77±1.49	43.68±1.08
DROPS	89.00±0.24	84.01±0.44	81.04±0.45	54.03±0.73	40.27±0.96	35.72±0.78
Method	***CIFAR-10* ($\delta = 1.0$-worst case acc)**			***CIFAR-100* ($\delta = 1.0$-worst case acc)**		
	$\rho = 10$	$\rho = 50$	$\rho = 100$	$\rho = 10$	$\rho = 50$	$\rho = 100$
Cross Entropy	79.98±0.13	59.48±1.75	43.30±3.63	27.06±0.53	11.24±0.97	6.52±0.18
Focal	64.96±5.67	43.04±5.05	30.46±3.13	23.62±1.19	6.32±0.57	3.42±0.24
LDAM	79.68±0.61	59.58±3.06	42.94±2.42	27.44±1.56	11.26±0.60	6.62±0.71
Balanced-Softmax	80.72±0.51	68.34±0.55	24.10±0.33	30.36±0.81	17.06±0.47	12.00±0.52
Logit-Adjust	82.18±0.46	73.50±1.09	63.62±1.70	28.02±1.83	15.86±1.26	11.66±1.40
Logit-Adj (post-hoc)	82.52±0.47	73.84±1.69	64.54±1.33	33.46±1.26	17.96±1.08	14.34±0.53
DROPS	86.28±0.13	80.80±0.68	77.12±1.26	43.56±0.76	28.42±1.60	23.38±0.88
Method	***CIFAR-10* (Worst)**			***CIFAR-100* (Worst)**		
	$\rho = 10$	$\rho = 50$	$\rho = 100$	$\rho = 10$	$\rho = 50$	$\rho = 100$
Cross Entropy	78.29±0.86	55.32±3.58	36.00±6.11	9.79±2.57	1.38±0.91	0.00±0.00
Focal	63.69±5.99	40.73±4.98	28.16±3.94	8.65±3.58	0.00±0.00	0.00±0.00
LDAM	76.87±0.99	56.57±3.18	35.99±2.72	10.57±2.65	1.82±1.16	0.27±0.50
Balanced-Softmax	78.86±0.84	67.11±0.98	22.63±0.29	16.17±3.85	6.05±2.24	2.58±1.42
Logit-Adjust	80.54±1.11	71.24±1.72	61.18±2.07	15.99±3.52	5.44±1.31	1.87±1.21
Logit-Adj (post-hoc)	81.41±1.24	72.00±1.72	63.16±2.43	16.21±3.93	4.47±1.71	3.39±1.46
DROPS	82.87±1.30	76.22±2.17	73.26±1.98	31.05±3.67	15.68±2.61	10.67±2.96

Performances Comparisons on the Generic δ-Worst Case Accuracy Experiment results in Table 1 demonstrate that DROPS can not only give promising performance under the reported δ-worst case accuracy (for $\delta = 1.0$), it also outperforms other methods in the worst case accuracy, and remains competitive on the mean accuracy as well (performs best on the CIFAR-10 setting).

To further investigate the robustness of each method under different level of prior shifts, we visualize the performance of each method under a list of δ within the δ-worst case accuracy. For the uniform distribution $r = u$, the δ values recover both the mean accuracy (with $\delta = 0$) and the worst accuracy (for large enough δ), and interpolates between these two metrics for other values of δ. For DROPS, we train using $\delta_{\text{train}} \in [0.5, 1.0, 1.5, 2.0]$ (CIFAR-10) and $\delta_{\text{train}} \in [0.5, 1.0, 2.0, 3.0, 4.0]$ (CIFAR-100) with KL-divergence while evaluating for the aforementioned range of δ_{eval} values. We also evaluate using Reverse-KL divergence to examine the behavior under a different divergence function from that used in learning the scaling adjustments with DROPS. Figure 1 illustrates the robustness and effectiveness of our proposed DROPS method: specifically, in each sub-figure of Figure 1, the x-axis indicates the value of δ, and the y-axis denotes the corresponding δ-worse case accuracy. For experiment results on CIFAR-10 datasets (1st row), the curves of DROPS under different δ_{train} are consistently higher than the other baselines, indicating that with the increasing of perturbation level of the distribution shifts (from left to right in each sub-figure), DROPS is more robust to the distribution shift. As for CIFAR-100 dataset, we do observe that optimizing for the controlled worst case performance may lead to a trade-off in the worst case accuracy and averaged test accuracy, i.e., Logit-Adj is more competitive than DROPS in the measure of the averaged accuracy, while DROPS still suffers less from the increase of perturbation level.

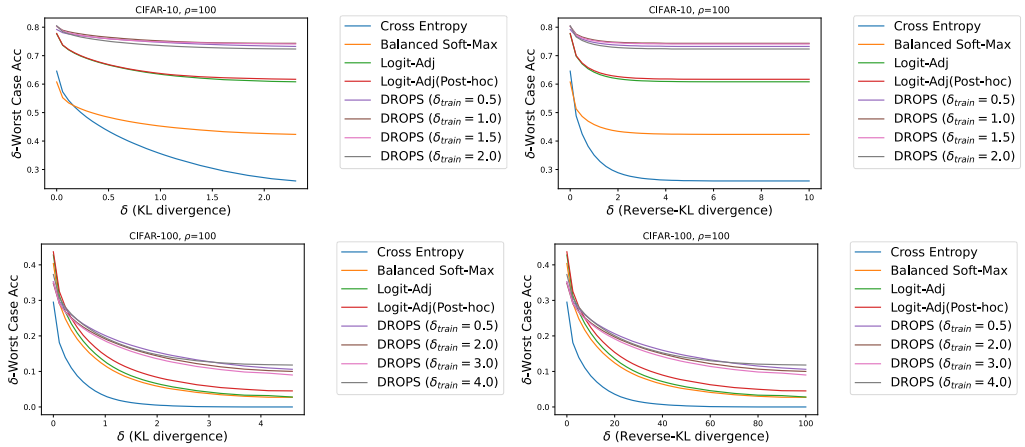

Figure 1: Performance/Robustness of methods under different perturbation δ-worst case accuracy. (1st row: CIFAR-10 with imbalance ratio $\rho = 100$, adopt the KL-divergence measure (left) and Reverse-KL divergence (right) for the δ-worst case accuracy calculation; 2nd row: CIFAR-100 with imbalance ratio $\rho = 100$, adopt the KL-divergence measure (left) and Reverse-KL divergence (right) for the δ-worst case accuracy calculation.)

6.2 Experiments on Group Robustness Tasks

We consider the group robustness tasks to show the robustness of DROPS under group prior shifts. For waterbirds and CelebA datasets, we obtain the datasets as described in (Sagawa et al., 2020).

We compare the performance of our proposed method with several popular group-Dro approaches, i.e., Just-train-twice (JTT) (Liu et al., 2021), Group Distributional Robustness (G-DRO) (Sagawa et al., 2020), data balancing strategies (SUBG) (Idrissi et al., 2021), last-layer re-training (DFR) (Kirichenko et al., 2022). Among the baseline methods, ERM, JTT, G-DRO, and SUBG are trained with the same architecture (ResNet 50 (He et al., 2016)) for 5 runs, the base model for DFR and our approach DROPS also used the same ERM pre-trained model. Note that the knowledge requirement over group information and validation set differs among the reported methods, we clarify the differences in the column "Group Info" of Table 2.

As for the performance of DROPS in the group DRO datasets, since there are only two classes within the two datasets, we define a single post-hoc scalar as w, which is considered to be a hyper-parameter by scaling the classifier's prediction on class 1. Note that the spurious features (minority groups among each class) tend to have a significant impact on the ERM performances by referring to the imbalanced group distribution. We use DROPS to learn post-doc corrections for both ERM

trained model and for the DFR (Kirichenko et al., 2022) model that additionally retrains the last layer of the ERM-trained model with a balanced held-out set. We refer to the post-hoc corrected DFR as DROPS* in the result tables.

Performance Comparisons on various δ-Worst Accuracies. Post-hoc scaling on the ERM model (referred as DROPS in Table 2) improves the robustness of the model as measured at various δ values (including $\delta = 0$ and $\delta \to \infty$). DROPS also outperforms JTT on the CelebA dataset. When applying post-hoc scaling to a better pre-trained model, i.e., with DFR (Kirichenko et al., 2022) that re-trains the last layer on the validation set, DROPS* outperforms all baseline methods in most settings, with the performance improvement especially clear on the CelebA dataset.

Table 2: Performance comparisons on Waterbirds and CelebA: averaged accuracy with training prior weight and uniform weight, δ-worst accuracy for several δ perturbations, and worst group-level test accuracy are reported. The *Group Info* column indicates whether the group labels are available to the methods on train/validation sets. The symbol ✓✓means the method also re-trains the last layer on the validation set (w/o needing the group information for training).

Method	**Group Info**		(Train prior)	***Waterbirds*** ($r \to$ uniform prior)					
	Train	**Val**	**Averaged**	**Averaged**	$\delta = 0.05$**-Worst**	$\delta = 0.1$**-Worst**	$\delta = 0.2$**-Worst**	$\delta = 0.5$**-Worst**	**Worst**
ERM	✗	✓	98.08±0.20	88.09±0.90	84.31±1.24	82.71±1.41	80.51±1.68	76.65±2.25	70.65±3.32
JTT	✗	✓	93.05±0.36	89.56±0.69	88.59±0.74	88.24±0.75	87.81±0.77	87.13±0.85	86.18±1.08
DROPS	✗	✓	97.95±0.16	88.55±1.15	85.50±1.51	84.33±1.64	82.81±1.79	80.41±2.00	77.14±2.15
G-DRO	✓	✓	93.03±0.34	91.67±0.22	91.23±0.33	91.06±0.34	90.83±0.40	90.44±0.52	89.85±0.73
SUBG	✓	✓	91.97±0.50	90.05±0.44	89.46±0.40	89.24±0.39	88.98±0.40	88.59±0.49	88.12±0.76
$\text{DFR}^{\text{Tr}}_{\text{Tr}}$	✓	✓	95.83±0.94	93.45±0.49	92.77±0.48	92.51±0.51	92.16±0.55	91.58±0.67	90.72±0.91
$\text{DFR}^{\text{Val}}_{\text{Tr}}$	✗	✓✓	93.17±1.30	93.29±0.80	92.98±0.84	92.86±0.85	92.70±0.87	92.42±0.88	92.01±0.88
DROPS*	✗	✓✓	93.01±1.32	93.42±0.61	93.08±0.81	92.98±0.90	92.76±1.02	92.45±1.23	91.99±1.56
Method	**Group Info**		(Train prior)	***CelebA*** ($r \to$ uniform prior)					
	Train	**Val**	**Averaged**	**Averaged**	$\delta = 0.05$**-Worst**	$\delta = 0.1$**-Worst**	$\delta = 0.2$**-Worst**	$\delta = 0.5$**-Worst**	**Worst**
ERM	✗	✓	95.33±0.12	81.18±1.60	73.78±2.36	70.53±2.69	65.97±3.15	57.82±3.98	44.89±5.30
JTT	✗	✓	87.78±0.73	85.04±1.05	83.34±1.35	82.89±1.49	82.35±1.71	81.51±2.12	79.71±2.85
DROPS	✗	✓	90.67±0.76	89.05±0.88	86.80±1.34	85.89±1.55	84.67±1.88	82.59±2.50	79.44±3.58
G-DRO	✓	✓	92.59±0.87	90.95±0.52	90.18±0.46	89.89±0.43	89.53±0.37	88.97±0.26	88.21±0.17
SUBG	✓	✓	91.09±0.62	88.75±0.34	87.69±0.31	87.27±0.41	86.72±0.73	85.80±0.29	84.45±0.28
$\text{DFR}^{\text{Tr}}_{\text{Tr}}$	✓	✓	90.36±0.64	89.25±0.73	87.30±0.97	86.53±1.09	85.50±1.26	83.77±1.63	81.22±2.31
$\text{DFR}^{\text{Val}}_{\text{Tr}}$	✗	✓✓	90.90±0.79	91.13±0.40	90.32±0.54	90.02±0.58	89.64±0.65	89.05±0.77	88.25±1.03
DROPS*	✗	✓✓	95.69±0.41	93.59±0.36	92.64±0.46	92.28±0.51	91.82±0.58	91.10±0.68	90.19±0.81

7 Conclusions

We study the problem of improving the the distributional robustness of a pre-trained model under controlled distribution shifts. We propose DROPS, a fast and lightweight post-hoc method that learns scaling adjustments to predictions from a pre-trained model. DROPS learns the adjustments by solving a constrained optimization problem on a held-out validation set, and then applies these adjustments to the model predictions during evaluation time. DROPS is able to reuse the same pre-trained model for different robustness requirements by simply scaling the model predictions. For group robustness tasks, our approach only needs group annotations for a smaller held-out set. We also showed provable convergence guarantees for our method. Experimental results on standard benchmarks for class imbalance (CIFAR-10, CIFAR-100) and group DRO (Waterbirds, CelebA) demonstrate the effectiveness and robustness of DROPS when evaluated on a range of distribution shifts away from the target prior distribution.

REFERENCES

Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *International Conference on Machine Learning*, pp. 145–155. PMLR, 2020.

Amr Alexandari, Anshul Kundaje, and Avanti Shrikumar. Maximum likelihood with bias-corrected calibration is hard-to-beat at label shift adaptation. In *International Conference on Machine Learning*, pp. 222–232. PMLR, 2020.

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

Kamyar Azizzadenesheli, Anqi Liu, Fanny Yang, and Animashree Anandkumar. Regularized learning for domain adaptation under label shifts. In *International Conference on Learning Representations*, 2019.

Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019.

Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling. Feature space augmentation for long-tailed data. In *European Conference on Computer Vision*, pp. 694–710. Springer, 2020.

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.

Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

John Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *arXiv preprint arXiv:1810.08750*, 2018.

John Duchi, Peter Glynn, and Hongseok Namkoong. Statistics of robust optimization: A generalized empirical likelihood approach. *arXiv preprint arXiv:1610.03425*, 2016.

Hao Guo and Song Wang. Long-tailed multi-label visual recognition by collaborative training on uniform and re-balanced samplings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15089–15098, 2021.

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.

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

Badr Youbi Idrissi, Martin Arjovsky, Mohammad Pezeshki, and David Lopez-Paz. Simple data balancing achieves competitive worst-group-accuracy. In *Conference on Causal Learning and Reasoning*, pp. 336–351. PMLR, 2022.

Wittawat Jitkrittum, Aditya Krishna Menon, Ankit Singh Rawat, and Sanjiv Kumar. Elm: Embedding and logit margins for long-tail learning. *arXiv preprint arXiv:2204.13208*, 2022.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019.

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

Jyrki Kivinen and Manfred K Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *information and computation*, 132(1):1–63, 1997.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on scomputer vision*, pp. 2980–2988, 2017.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pp. 3122–3130. PMLR, 2018.

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.

Jialun Liu, Yifan Sun, Chuchu Han, Zhaopeng Dou, and Wenhui Li. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2970–2979, 2020.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=37nvvqkCo5.

Harikrishna Narasimhan, Andrew Cotter, and Maya Gupta. Optimizing generalized rate metrics with three players. In *Advances in Neural Information Processing Systems*, 2019.

Seulki Park, Jongin Lim, Younghan Jeon, and Jin Young Choi. Influence-balanced loss for imbalanced visual classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 735–744, 2021.

Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.

Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. Focus on the common good: Group distributional robustness follows. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=irARV_2VFs4.

Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in neural information processing systems*, 33:4175–4186, 2020.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. Domain-adjusted regression or: Erm may already learn features sufficient for out-of-distribution generalization. *arXiv preprint arXiv:2202.06856*, 2022.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020.

S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, New York, NY, USA, 2014.

Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and trends in Machine Learning*, 4(2):107–194, 2011.

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

Maurice Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171–176, 1958.

Serena Wang, Harikrishna Narasimhan, Yichen Zhou, Sara Hooker, Michal Lukasik, and Aditya Krishna Menon. Robust distillation for worst-class performance, 2022.

Tao Wang, Yu Li, Bingyi Kang, Junnan Li, Junhao Liew, Sheng Tang, Steven Hoi, and Jiashi Feng. The devil is in classification: A simple framework for long-tail instance segmentation. In *European conference on computer vision*, pp. 728–744. Springer, 2020.

Hongxin Wei, Lue Tao, Renchunzi Xie, and Bo An. Open-set label noise can improve robustness against inherent label noise. *Advances in Neural Information Processing Systems*, 34:7978–7992, 2021.

Ziyu Xu, Chen Dan, Justin Khim, and Pradeep Ravikumar. Class-weighted classification: Trade-offs and robust approaches. In *International Conference on Machine Learning*, pp. 10544–10554. PMLR, 2020.

Xi Yin, Xiang Yu, Kihyuk Sohn, Xiaoming Liu, and Manmohan Chandraker. Feature transfer learning for face recognition with under-represented data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5704–5713, 2019.

Runtian Zhai, Chen Dan, Zico Kolter, and Pradeep Ravikumar. Doro: Distributional and outlier robust optimization. In *International Conference on Machine Learning*, pp. 12345–12355. PMLR, 2021.

Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. Distribution alignment: A unified framework for long-tail visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2361–2370, 2021.

Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16489–16498, 2021.

Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9719–9728, 2020.