Robust Distillation for Worst-class Performance: On the Interplay Between Teacher and Student Objectives

Abstract

Knowledge distillation is a popular technique that has been shown to produce remarkable gains in average accuracy. However, recent work has shown that these gains are not uniform across subgroups in the data, and can often come at the cost of accuracy on rare subgroups and classes. Robust optimization is a common remedy to improve worst-class accuracy in standard learning settings, but in distillation it is unknown whether it is best to apply robust objectives when training the teacher, the student, or both. This work studies the interplay between robust objectives for the teacher and student. Empirically, we show that that jointly modifying the teacher and student objectives can lead to better worst-class student performance and even Pareto improvement in the tradeoff between worst-class and overall performance. Theoretically, we show that the *per-class calibration* of teacher scores is key when training a robust student. Both the theory and experiments support the surprising finding that applying a robust teacher training objective does not always yield a more robust student.

1 INTRODUCTION

Knowledge distillation, wherein one trains a *teacher* model and uses its predictions to train a *student* model of similar or smaller capacity, has proven to be a powerful tool that improves efficiency while achieving state-of-the-art classification accuracies [Hinton et al., 2015a, Radosavovic et al., 2018, Anil et al., 2018, Pham et al., 2021]. Remarkably, the student accuracy under distillation is capable of even surpassing that of the teacher (e.g. Xie et al. [2020]).

However, recent work has shown that the gains in average accuracy may not be uniform across subgroups, and can hurt performance on subgroups that are rarer or more difficult to classify. This is particularly true of long-tailed classification settings, where the improved average accuracy often comes at the cost of poorer accuracies on the tail classes [Lukasik et al., 2022, Du et al., 2021], and model compression can further amplify these performance disparities [Hooker et al., 2020, Xu et al., 2021].

To mitigate the disparity between average and subgroup accuracy, a common remedy is to train a model to achieve low *worst-group* test error. Suitably modified robust optimization techniques have successfully achieved state-of-the-art worst-class performance with manageable computational overhead [Sagawa et al., 2020a, Sohoni et al., 2020]. However, the evaluation of these techniques has thus far primarily focused on the standard training setting involving a single model. In the increasingly popular distillation setting, which involves both a teacher and student model, there is limited understanding of how these approaches can be applied to achieve the best trade-offs between average and worst-class performance. In particular, it is unknown if the best results come from using a robust objective for the teacher, the student or *both*.

This work studies the interplay between robust training objectives for the teacher and student. We focus on a multi-class classification setting where we define worst-class accuracy as the lowest per-class recall. Empirically, we show that jointly modifying *both* the teacher and student objectives with robust objectives not only improves the worst-class accuracy of the student, but can provide Pareto improvements in the trade-off between average and worst-class performance. Theoretically, we analyze what makes a good teacher when training a robust student, and give to our knowledge the first concrete characterization of this by showing that the student's robustness depends on how *well-calibrated* the teacher's scores are for the individual classes.

Our contributions proceed as follows:

(i) We begin with the problem setup (§2), and adapt existing robust optimization objectives to a distillation setting, allowing for different combinations of modifica-

Submitted to the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023). **To be used for reviewing only**.

tions to *both* the teacher and student objectives (§3). We provide adapted algorithms to address practical training issues that arise when applying robust objectives to both the teacher and student (such as margin-based surrogate losses and shared validation set usage).

(ii) We demonstrate empirically on benchmark image datasets that the different combinations of student and teacher objectives not only improve the student's worst-class accuracy, but yield better trade-offs between average and worst-class performance than baselines (§4). Perhaps surprisingly, we find that the teacher's worst-class accuracy is not always predictive of the teacher's ability to yield robust students.

(iii) We show theoretically that the worst-class robustness of the student depends on the *per-class calibration* of the teacher, and additionally derive robustness guarantees for the student in terms of the teacher's errors (§5).

1.1 RELATED WORK

Worst-group robustness: The goal of achieving good worst-case performance across subgroups can be framed as a (group) distributionally robust optimization (DRO) problem, and can be solved by iteratively updating costs on the individual groups and minimizing the resulting cost-weighted loss [Chen et al., 2017]. Recent variants of this approach have sought to avoid over-fitting through group-specific regularization [Sagawa et al., 2020a,b] or margin-based losses [Narasimhan and Menon, 2021, Kini et al., 2021], and to handle unknown subgroups [Sohoni et al., 2020]. In the context of distillation, Lukasik et al. [2022] propose simple modifications to robustify the student's objective by controlling the strength of the teacher's labels for different groups. In contrast, we propose a more direct and theoretically-grounded procedure that seeks to explicitly optimize for the student's worst-case error.

Relationship to Narasimhan and Menon [2021]: This paper builds on the margin-based DRO framework of Narasimhan and Menon [2021], who also include preliminary distillation experiments on training the teacher with standard ERM and the student with a robust objective. However, this and other prior work [Lukasik et al., 2022] have only explored modifications to the student loss, while training the teacher using a standard procedure. Our robust distillation proposals build on this method, but carry out a more extensive analysis, exploring different combinations of teacher-student objectives and different trade-offs between average and worst-class performance. Additionally, we provide robustness guarantees for the student, equip the DRO algorithms to achieve different trade-offs between overall and worst-case error, and provide a rigorous analysis of different design choices, such as the use of teacher labels for the multiplier updates.

Long-tail learning. There has been much work on training classifiers from long-tail data, ranging from modifications to loss modifications [Cao et al., 2019a, Menon et al., 2021b, Cui et al., 2021] to architectural changes [Wang et al., 2020, Cui et al., 2022]. All these methods focus on the standard single model training setup, and seek to maximize the balanced (and not the worst-class) accuracy. Recent attempts have sought to modify standard distillation for long-tail learning, by either re-balancing the student loss [Zhang et al., 2021], temperature-scaling the teacher predictions [He et al., 2021], employing multiple teachers [Xiang et al., 2020], and leveraging the teacher's intermediate embeddings [Iscen et al., 2021]. The common goal in most of these papers is to modify the student's objective to incorporate different forms of supervision from the teacher. In contrast, we seek to explore modifications to the teacher's training objective to improve the student's robustness.

Role of the teacher's objective. Few previous works have studied how the objective of the teacher affects the student performance. For example, multiple works have studied the effect label smoothing objectives of the teacher model, some finding it to harm the student performance [Müller et al., 2019], improve the student [Shen et al., 2021] or show varying impact depending on the temperature value [Chandrasegaran et al., 2022]. In another work, Lukasik et al. [2020] showed how applying noise correction objectives to the teacher often yield better result than only applying noise correction objectives in the student. We are not aware of a previous work studying the *interplay* between the student and the teacher objectives on the robustness of the student.

2 PROBLEM SETUP

We consider a multi-class classification problem with instance space $\mathcal{X}$ and output space $[m] = \{1, \ldots, m\}$. Let D denote the underlying data distribution over $\mathcal{X} \times [m]$, and $D_{\mathcal{X}}$ denote the marginal distribution over $\mathcal{X}$. Let Δ_m denote the $(m-1)$-dimensional probability simplex over m classes. 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 $\pi_y = \mathbb{E}_{X \sim D_{\mathcal{X}}}[\eta_y(X)]$.

Learning objectives. Our goal is to learn a multiclass 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 $f : \mathcal{X} \to \mathbb{R}^m$ that assigns scores $[f_1(x), \ldots, f_m(x)] \in \mathbb{R}^m$ to a given instance x, and construct the classifier by predicting the class with the highest score: $h(x) = \operatorname{argmax}_{j \in [m]} f_j(x)$. We will denote a softmax transformation of f by $\operatorname{softmax}_y(f(x)) = \frac{\exp(f_y(x))}{\sum_j \exp(f_j(x))}$, and use the notation $\operatorname{softmax}_y(f(x)) \propto z_y$ to indicate that $\operatorname{softmax}_y(f(x)) = \frac{z_y}{\sum_{j=1}^m z_j}$.

We measure the efficacy of the scoring function f using a loss function $\ell : [m] \times \mathbb{R}^m \to \mathbb{R}_+$ that assigns a penalty $\ell(y, z)$ for predicting score vector $z \in \mathbb{R}^m$ for true label y. Examples of loss functions include the 0-1 loss: $\ell^{0\text{-}1}(y, z) =$

$\mathbf{1}\left(z \neq \operatorname{argmax}_j f_j(x)\right)$, and the softmax cross-entropy loss: $\ell^{\text{xent}}(y, z) = -f_y(x) + \log\left(\sum_{j\in[m]} \exp\left(f_j(x)\right)\right)$.

Standard objective: A standard machine learning goal entails minimizing the overall expected risk:

$$L^{\text{std}}(f) = \mathbb{E}\left[\ell(Y, f(X))\right]. \tag{1}$$

Balanced objective: In applications where the classes are severely imbalanced, i.e., the class priors π_y are non-uniform and significantly skewed, one may wish to instead optimize a *balanced* version of the above objective, where we average over the conditional loss for each class. Notice that the conditional loss for class y is weighted by the inverse of its prior:

$$\begin{aligned} L^{\text{bal}}(f) &= \frac{1}{m} \sum_{y\in[m]} \mathbb{E}\left[\ell(y, f(X)) \,|\, Y = y\right] \\ &= \frac{1}{m} \sum_{y\in[m]} \frac{1}{\pi_y} \mathbb{E}_X\left[\eta_y(X)\, \ell(y, f(X))\right]. \end{aligned} \tag{2}$$

Robust objective: A more stringent objective would be to focus on the worst-performing class, and minimize a *robust* version of (1) that computes the worst among the m conditional losses:

$$L^{\text{rob}}(f) = \max_{y\in[m]} \frac{1}{\pi_y} \mathbb{E}\left[\eta_y(X)\, \ell(y, f(X))\right]. \tag{3}$$

In practice, focusing solely on either the average or the worst-case performance may not be an acceptable solution, and therefore, in this paper, we will additionally seek to characterize the trade-off between the balanced and robust objectives. One way to achieve this trade-off is to minimize the robust objective, while constraining the balanced objective to be within an acceptable range. This constrained optimization can be equivalently formulated as optimizing a convex combination of the balanced and robust objectives, for trade-off $\alpha \in [0, 1]$:

$$L^{\text{tdf}}(f) = (1-\alpha) L^{\text{bal}}(f) + \alpha L^{\text{rob}}(f). \tag{4}$$

A similar trade-off can also be specified between the standard and robust objectives. To better understand the differences between the standard, balanced and robust objectives in (1)–(4), we look at the optimal scoring function for each given a cross-entropy loss:

Theorem 1 (**Bayes-optimal scorers**). *When ℓ is the cross-entropy loss ℓ^{xent}, the minimizers of* (1)–(3) *over all measurable functions $f : \mathcal{X} \to \mathbb{R}^m$ are given by:*

(i) $L^{\text{std}}(f)$: $\operatorname{softmax}_y(f^*(x)) = \eta_y(x)$
(ii) $L^{\text{bal}}(f)$: $\operatorname{softmax}_y(f^*(x)) \propto \frac{1}{\pi_y}\eta_y(x)$
(iii) $L^{\text{rob}}(f)$: $\operatorname{softmax}_y(f^*(x)) \propto \frac{\lambda_y}{\pi_y}\eta_y(x)$
(iv) $L^{\text{tdf}}(f)$: $\operatorname{softmax}_y(f^*(x)) \propto \frac{(1-\alpha)\frac{1}{m}+\alpha\lambda'_y}{\pi_y}\eta_y(x)$,

for class-specific constants $\lambda, \lambda' \in \mathbb{R}^m_+$ that depend on distribution D.

All proofs are provided in Appendix A. Interestingly, the optimal scorers for all four objectives involve a simple scaling of the conditional-class probabilities $\eta_y(x)$.

3 DISTILLATION FOR WORST-CLASS PERFORMANCE

We adopt the common practice of training both the teacher and student on the same dataset. Specifically, given a training sample $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ drawn from D, we first train a teacher model $p^t : \mathcal{X} \to \Delta_m$, and use it to generate a student dataset $S' = \{(x_1, p^t(x_1)), \ldots, (x_n, p^t(x_n))\}$ by replacing the original labels with the teacher's predictions. We then train a student scorer $f^s : \mathcal{X} \to [m]$ using the re-labeled dataset, and use it to construct the final classifier.

Teacher and student objectives. In a typical setting, both the teacher and student are trained to optimize a version of the standard objective in (1), i.e., the teacher is trained to minimize the average loss against the original training labels, and the student is trained to minimize an average loss against the teacher's predictions:

$$\begin{aligned} \text{Teacher: } \hat{L}^{\text{std}}(f^t) &= \frac{1}{n}\sum_{i=1}^{n} \ell\left(y_i, f^t(x_i)\right); \\ \text{Student: } \hat{L}^{\text{std-d}}(f^s) &= \frac{1}{n}\sum_{i=1}^{n}\sum_{y=1}^{m} p^t_y(x_i)\, \ell\left(y, f(x_i)\right), \end{aligned} \tag{5}$$

where $p^t(x) = \operatorname{softmax}(f^t(x))$. It is also common to have the student use a mixture of the teacher and one-hot labels. For concreteness, we consider a simpler distillation setup without this mixture, though extensions with this mixture would be straightforward to add. This work takes a wider view and explores *what combinations of student and teacher objectives* facilitate better worst-group performance for the student. Our experiments evaluate all *nine* combinations of standard, balanced, and robust teacher objectives, paired with standard, balanced, and robust student objectives.

Given the choice of teacher objective, the student will either optimize a distilled version of the balanced objective in (2):

$$\hat{L}^{\text{bal-d}}(f^s) = \frac{1}{m}\sum_{y\in[m]} \frac{1}{\hat{\pi}^t_y} \frac{1}{n}\sum_{i=1}^{n} p^t_y(x_i)\, \ell\left(y, f^s(x_i)\right), \tag{6}$$

or a distilled version of the robust objective in (3):

$$\hat{L}^{\text{rob-d}}(f^s) = \max_{y\in[m]} \frac{1}{\hat{\pi}^t_y} \frac{1}{n}\sum_{i=1}^{n} p^t_y(x_i)\, \ell\left(y, f^s(x_i)\right). \tag{7}$$

In practice, the teacher's predictions may have a different marginal distribution from the underlying class priors, particularly when temperature scaling is applied to the teacher's logits to soften the predicted probabilities [Narasimhan and Menon, 2021]. To address this, in both (6) and (7) we have

replaced the class priors π_y with the marginal distribution $\hat{\pi}_y^t = \frac{1}{n}\sum_{i=1}^n p_y^t(x_i)$ from the teacher's predictions.

In addition to exploring the combination of objectives that facilitates better worst-group performance for the student, we evaluate a more flexible approach – have both the teachers and the students trade-off between the balanced and robust objectives:

$$\text{Teacher: } \hat{L}^{\text{tdf}}(f^t) = (1-\alpha^t)\hat{L}^{\text{bal}}(f^t) + \alpha^t \hat{L}^{\text{rob}}(f^t) \quad (8)$$
$$\text{Student: } \hat{L}^{\text{tdf-d}}(f^s) = (1-\alpha^s)\hat{L}^{\text{bal-d}}(f^s) + \alpha^s \hat{L}^{\text{rob-d}}(f^s),$$

where $\hat{L}^{\text{bal}}(f^t)$ and $\hat{L}^{\text{rob}}(f^t)$ are the respective empirical estimates of (2) and (3) from the training sample, and $\alpha^t, \alpha^s \in [0,1]$ are the respective tradeoff parameters for the teacher and student. We are thus able to evaluate the Pareto-frontier of balanced and worst-case accuracies, obtained from different combinations of the teachers and students, and trained with different trade-off parameters.

3.1 ROBUST DISTILLATION ALGORITHMS

The different objectives we consider – standard, balanced and robust – entail different loss objectives to ensure efficient optimization during training. For example, while training the standard teacher and student in (5), we take ℓ to be the softmax cross-entropy loss, and optimize it using SGD. For the balanced and robust models, we employ the margin-based surrogates that we detail below, which have shown to be more effective in training over-parameterized networks [Cao et al., 2019b, Menon et al., 2021b, Kini et al., 2021]. Across all objectives, at evaluation we take the loss ℓ in the student and teacher objectives to be the 0-1 loss.

Margin-based surrogate for balanced objective. When the teacher or student model being trained is over-parameterized, i.e., has sufficient capacity to correctly classify all examples in the training set, the use of an outer weighting term in the objective (such as the inverse class marginals in (6)) can be ineffective. In other words, a model that yields zero training objective would do so irrespective of what outer weights we choose. To remedy this problem, we make use of the margin-based surrogate of Menon et al. [2021b], and incorporate the outer weights as margin terms within the loss. For the balanced student objective in (6), this would look like:

$$\widetilde{L}^{\text{bal-d}}(f^s) = \frac{1}{n}\sum_{i=1}^n \mathcal{L}^{\text{mar}}\left(p^t(x_i), f^s(x_i); \mathbf{1}/\hat{\pi}^t\right), \quad (9)$$

where $\mathcal{L}^{\text{mar}}(\mathbf{p}, \mathbf{f}; \mathbf{c}) =$

$$\frac{1}{m}\sum_{y\in[m]} p_y \log\left(1 + \sum_{j\neq y}\exp\left(\log(c_y/c_j) - (f_y - f_j)\right)\right),$$

for teacher probabilities $\mathbf{p} \in \Delta_m$, student scores $\mathbf{f} \in \mathbb{R}^m$, and per-class costs $\mathbf{c} \in \mathbb{R}_+^m$. For the balanced teacher, the margin-based objective would take a similar form, but with one-hot labels.

Algorithm 1 Distilled Margin-based DRO

Inputs: Teacher p^t, Student hypothesis class $\mathcal{F}$, Training set S, Validation set S^{val}, Step-size $\gamma \in \mathbb{R}_+$, Number of iterations K, Loss ℓ, Initial student $f^0 \in \mathcal{F}$, Initial multipliers $\lambda^0 \in \Delta_m$
Compute $\hat{\pi}_j^t = \frac{1}{n}\sum_{(x,y)\in S} p_j^t(x),\ \forall j \in [m]$
Compute $\hat{\pi}_j^{t,\text{val}} = \frac{1}{n^{\text{val}}}\sum_{(x,y)\in S^{\text{val}}} p_j^t(x),\ \forall j \in [m]$
For $k = 0$ to $K-1$
$\tilde{\lambda}_j^{k+1} = \lambda_j^k \exp\left(\gamma \hat{R}_j\right), \forall j \in [m]$ where $\hat{R}_j = \frac{1}{n^{\text{val}}}\frac{1}{\hat{\pi}_j^{t,\text{val}}}\sum_{(x,y)\in S^{\text{val}}} p_j^t(x)\,\ell(j, f^k(x))$
$\lambda_y^{k+1} = \frac{\tilde{\lambda}_y^{k+1}}{\sum_{j=1}^m \tilde{\lambda}_j^{k+1}}, \forall y$
$f^{k+1} \in \operatorname{argmin}_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^n \mathcal{L}^{\text{mar}}\left(p^t(x_i), f(x_i); \frac{\lambda^{k+1}}{\hat{\pi}^t}\right)$
// Replaced with a few steps of SGD
End For
Output: $\bar{f}^s : x \mapsto \frac{1}{K}\sum_{k=1}^K f^k(x)$

We include a proof in Appendix A.3 showing that a scoring function that minimizes the surrogate objective in (9) also minimizes the the balanced objective in (6) (when ℓ is the cross-entropy loss, and the student is chosen from a sufficiently flexible function class). In practice, the margin term $\log(c_y/c_j)$ encourages a larger margin of separation for classes y for which the cost c_y is relatively higher.

Margin-based DRO for robust objective. Minimizing the robust objective with plain SGD can be difficult due to the presence of the outer "max" over m classes. The key difficulty is in computing reliable stochastic gradients for the max objective, especially given a small batch size. The standard approach is to instead use a (group) distributionally-robust optimization (DRO) procedure, which comes in multiple flavors Chen et al. [2017], Sagawa et al. [2020a], Kini et al. [2021]. We employ the margin-based variant of group DRO [Narasimhan and Menon, 2021] as it naturally extends the margin-based objective used in the balanced setting.

We illustrate below how this applies to the robust student objective in (7). The procedure for the robust teacher is similar, but involves one-hot labels. For a student hypothesis class $\mathcal{F}$, we first re-write the minimization in (7) over $f \in \mathcal{F}$ into an equivalent min-max optimization using per-class multipliers $\lambda \in \Delta_m$:

$$\min_{f\in\mathcal{F}}\max_{\lambda\in\Delta_m}\sum_{y\in[m]}\frac{\lambda_y}{\hat{\pi}_y^t}\frac{1}{n}\sum_{i=1}^n p_y^t(x_i)\,\ell\left(y, f(x_i)\right),$$

and then maximize over λ for fixed f, and minimize over f for fixed λ:

$$\lambda_y^{k+1} \propto \lambda_y^k \exp\left(\gamma \frac{1}{n\hat{\pi}_y^t}\sum_{i=1}^n p_y^t(x_i)\,\ell\left(y, f^k(x_i)\right)\right), \forall y$$

$$f^{k+1} \in \operatorname*{argmin}_{f \in \mathcal{F}} \sum_{y \in [m]} \frac{\lambda_y^{k+1}}{n \hat{\pi}_y^t} \sum_{i=1}^{n} p_y^t(x_i)\, \ell\left(y, f(x_i)\right),$$

where $\gamma > 0$ is a step-size parameter. The updates on λ implement exponentiated gradient (EG) ascent to maximize over the simplex [Shalev-Shwartz et al., 2011].

Following Narasimhan and Menon [2021], we make two modifications to the above updates when used to train over-parameterized networks that can fit the training set perfectly. First, we perform the updates on λ using a small held-out validation set $S^{\text{val}} = \{(x_1, y_1), \ldots, (x_{n^{\text{val}}}, y_{n^{\text{val}}})\}$, instead of the training set, so that the λs reflect how well the model generalizes out-of-sample. Second, in keeping with the balanced objective, we modify the weighted objective in the f-minimization step to include a margin-based surrogate. Algorithm 1 provides a summary of these steps and returns a scorer that averages over the K iterates: $\bar{f}^s(x) = \frac{1}{K}\sum_{k=1}^{K} f^k(x)$. While the averaging is needed for our theoretical analysis, in practice, we find it sufficient to return the last scorer f^K. In Appendix D, we describe how Algorithm 3 can be easily modified to trade-off between the balanced and robust objectives, as shown in (8).

4 EXPERIMENTS

To empirically understand the interplay of teacher and student objectives, we explore the following questions: *what combination of teacher and student objectives yield the highest worst-class accuracy? Can some combinations improve worst-class accuracy without sacrificing average accuracy?*

Datasets. We evaluate the proposed distillation protocols on benchmark image datasets: *(i)* CIFAR-10, *(ii)* CIFAR-100 [Krizhevsky, 2009], *(iii)* TinyImageNet (a subset of ImageNet with 200 classes) [Le and Yang, 2015], and *(iv)* ImageNet [Russakovsky et al., 2015]. We also include long-tailed versions of the first three datasets created by downsampling tail classes [Cui et al., 2019]. For both the original and long-tailed versions of the datasets, there are often biases in worst-class performance, possibly due to some classes being easier to learn [Lukasik et al., 2022, Hooker et al., 2020]. For all datasets, as done in prior work [Menon et al., 2021b, Narasimhan and Menon, 2021], we randomly split the original default test set in half to create a validation set and test set, and use the same validation and test sets for the long-tailed training sets as for the original versions.

Architectures. We evaluate our distillation protocols in both a self-distillation and compression setting. On all CIFAR datasets, all teachers were trained with the ResNet-56 architecture and students were trained with either ResNet-56 or ResNet-32. On TinyImageNet and ImageNet, teachers and students were trained with ResNet-18. More details on these architectures can be found in Lukasik et al. [2022] and He et al. [2016] (see, e.g., Table 7 in Lukasik et al. [2022]). Self-distillation results are reported in the main paper (teacher/student share the same architecture), and we include results with compressed students in Appendix F.

Hyperparameters. We apply temperature scaling to the teacher scores, i.e., compute $p^t(x) = \text{softmax}(f^t(x)/\gamma)$, and vary the temperature parameter γ over a range of $\{1, 3, 5\}$. A higher temperature produces a softer probability distribution over classes [Hinton et al., 2015b]. Unless otherwise specified, the temperature hyperparameters were chosen to achieve the highest worst-class accuracy on the validation set. We closely mimic the learning rate and regularization settings from prior work [Menon et al., 2021b, Narasimhan and Menon, 2021] (see Appendix E for details).

Which objective combinations are most robust? We begin by exploring the effect of the interaction between student and teacher objectives on worst-class accuracy. In Table 1, we search over combinations of the standard, balanced, and robust objectives for the teacher (L^{std}, L^{bal}, L^{rob}) and the student ($L^{\text{std-d}}$, $L^{\text{bal-d}}$, $L^{\text{rob-d}}$) (note that on the original datasets, L^{std} is equivalent to L^{bal}). For each combination, following prior conventions in long-tailed learning [Menon et al., 2021b, Lukasik et al., 2022], we report the *average accuracy* over all classes, and the *worst-class accuracy*, or minimum per-class recall over all classes (see (3)). For datasets with a long tail or high number of classes, we also report the *worst-k accuracy*, which is the average of the the worst k per-class recalls.

The first surprising finding in Table 1 is that *applying the robust objective twice isn't always best.* For all but one dataset, the $L^{\text{rob}}/L^{\text{rob-d}}$ teacher/student combination was outperformed by some other combination of either $L^{\text{std}}/L^{\text{rob-d}}$, $L^{\text{rob}}/L^{\text{std-d}}$, or $L^{\text{bal}}/L^{\text{rob-d}}$. Still, in the winning combination, at least one of the objectives was robust. This suggests that while the robust objective is effective for controlling worst-class accuracy, there may be some information loss in applying it twice to both the teacher and student.

To understand this information loss on the teacher's side, we highlight a second surprising finding that *the teacher with the best worst-class accuracy alone did not always produce the student with the best worst-class accuracy*. The robust teacher had the highest worst-class accuracy across all datasets, but for CIFAR-10 and all three long-tailed datasets, it was actually the L^{std} or L^{bal} teacher that produced the best robust student. This shows that there is more to a good teacher than just having good worst-class performance – in fact, we show theoretically in Section 5 that the property of the teacher that is most important for robust student performance is a form of *calibration* of per-class scores.

Trading off accuracy and robustness. Table 1 focuses on worst-class accuracy, but practitioners often must consider the trade-off between average accuracy and worst-class accuracy when deploying any model. To address this, we introduced the $L^{\text{tdf}}/L^{\text{tdf-d}}$ objectives for the teacher/student with trade-off parameters α^t, α^s. Figure 1 plots average and

Table 1: Worst-class accuracy comparisons for different combinations of teacher/student objectives. Worst-1 test accuracy is reported (worst-10 for TinyImageNet-LT) (best in **bold**), and average test accuracy is shown in parentheses. Mean accuracies are reported over repeat trainings (see extended table in Appendix for standard errors). Note that on the original datasets, L^{std} and $L^{\text{std-d}}$ are equivalent to L^{bal} and $L^{\text{bal-d}}$.

Student Obj.	**CIFAR-10** Teacher Obj. L^{std}	**CIFAR-10** Teacher Obj. L^{rob}	**CIFAR-100** Teacher Obj. L^{std}	**CIFAR-100** Teacher Obj. L^{rob}	**TinyImageNet** Teacher Obj. L^{std}	**TinyImageNet** Teacher Obj. L^{rob}
None	86.48 (93.74)	90.09 (92.67)	42.22 (72.42)	43.42 (68.81)	8.42 (56.79)	11.87 (48.40)
$L^{\text{std-d}}$	87.66 (94.34)	90.12 (94.07)	43.81 (74.61)	**45.33** (73.67)	6.32 (57.83)	10.53 (55.36)
$L^{\text{rob-d}}$	**90.94** (92.54)	85.14 (89.58)	42.96 (68.71)	27.59 (54.79)	9.98 (49.84)	**16.58** (46.11)

Student Obj.	**CIFAR-10-LT** Teacher Obj. L^{std}	**CIFAR-10-LT** Teacher Obj. L^{bal}	**CIFAR-10-LT** Teacher Obj. L^{rob}	**CIFAR-100-LT** Teacher Obj. L^{std}	**CIFAR-100-LT** Teacher Obj. L^{bal}	**CIFAR-100-LT** Teacher Obj. L^{rob}	**TinyImageNet-LT** Teacher Obj. L^{std}	**TinyImageNet-LT** Teacher Obj. L^{bal}	**TinyImageNet-LT** Teacher Obj. L^{rob}
None	57.26 (76.27)	68.52 (79.85)	74.80 (80.29)	0.00 (43.33)	3.75 (47.55)	10.33 (44.27)	0.00 (33.15)	2.11 (35.96)	4.92 (27.23)
$L^{\text{std-d}}$	36.67 (69.50)	66.96 (79.25)	71.15 (80.95)	0.00 (43.86)	2.39 (48.95)	7.32 (47.93)	0.00 (26.05)	0.00 (27.21)	1.87 (25.34)
$L^{\text{bal-d}}$	71.23 (80.50)	70.52 (81.12)	72.96 (80.71)	4.39 (50.40)	7.08 (50.10)	7.19 (47.51)	0.20 (30.43)	2.82 (39.41)	4.77 (38.41)
$L^{\text{rob-d}}$	63.85 (76.81)	**75.56** (80.81)	69.21 (76.72)	9.05 (33.75)	**12.52** (34.05)	10.32 (36.83)	0.00 (22.66)	**4.93** (35.43)	3.32 (25.11)

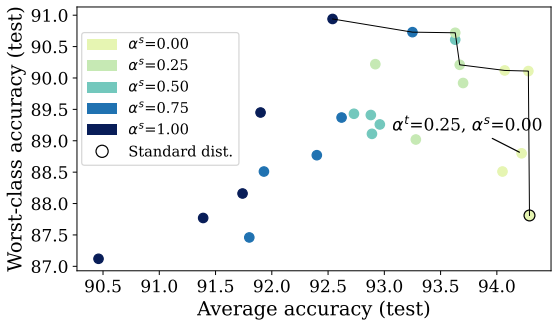

Figure 1: All α^t, α^s combinations for CIFAR-10 on test. The black line traces out the Pareto frontier. Average accuracy is roughly determined by α^s. The labeled point corresponds to the "best" combination selected in Table 2 based on validation criteria, but other domain-specific tradeoff criteria could yield any of these other points.

worst-class accuracies for a full spread of α^t, α^s parameters. First, we note that lower α^s usually leads to higher average accuracy (this is not always the case for α^t, which we show in more detail in Appendix F). Figure 1 also shows that combinations of α^t, α^s yield a roughly concave Pareto frontier of solutions with different average and worst-class accuracies to choose from. Selecting the best combination of trade-off parameters α^t, α^s in practice depends on domain-specific decisions regarding the importance of worst-class vs. average accuracy. Any selection criteria based on some trade-off of worst-class vs. average accuracy can be applied over the validation set to select α^t, α^s as hyperparameters. We demonstrate one such set of selection criteria here: in Table 2, we select α^t, α^s to maximize worst-class accuracy on validation, subject to having at least as high average accuracy as standard distillation (within error margin) on the validation set. Other candidate criteria include weighted sums of worst-class accuracy and average accuracy, or constrained optimization criteria from Cotter et al. [2019].

Comparison to baselines. Finally, we contextualize the performance of the proposed $L^{\text{tdf}}/L^{\text{tdf-d}}$ objectives and the training protocol in Algorithm 1 by comparing to several state-of-the-art methods. In addition to *standard distillation* (training the teacher with L^{std} and the student with $L^{\text{std-d}}$), we compare the proposed objective combinations with two recent works focusing on robust distillation [Lukasik et al., 2022, Narasimhan and Menon, 2021], both of which use a standard objective for the teacher and modify only the student objective for worst-class performance. From Narasimhan and Menon [2021], we consider the following two methods: *(i) Post-shifting:* this non-distillation approach directly constructs a new scoring model by making post-hoc adjustments to the teacher, so as to maximize the robust accuracy on the validation sample. *(ii) Robust student:* this approach trains a student using $L^{\text{rob-d}}$ from a standard teacher. From Lukasik et al. [2022], we compare to their two proposed *AdaMargin* and *AdaAlpha* methods. Both methods are motivated by the observation that the margin defined for each class y by $\gamma_{\text{avg}}(y, p^{\text{t}}(x)) = p^{\text{t}}_y(x) - \frac{1}{m-1}\sum_{y' \neq y} p^{\text{t}}_{y'}(x)$ correlates with whether distillation improves over one-hot training [Lukasik et al., 2022]. AdaMargin uses that quantity as a margin in the distillation loss, whereas AdaAlpha uses it to adaptively mix between the one-hot and distillation losses. Additionally, for long-tailed datasets, we include a comparison to Menon et al. [2021b] which we refer to as *balanced student*, where the student is distilled with a balanced objective $L^{\text{bal-d}}$ from a standard teacher. Finally, we also include a comparison to the *Group DRO* method for subgroup robustness without distillation (Algorithm 1 in Sagawa et al. [2020a]). This method differs from our DRO procedure in that they do not apply a margin-based loss.

Table 2 shows the average and worst-class accuracies on test for these baselines compared to the combination of α^t, α^s selected using the selection criteria previously described. The selection criteria for α^t, α^s are applied over the validation set, and thus do not directly translate to test per-

Table 2: Comparison to baselines for the selected α^t, α^s combination on test data.

Method	CIFAR-10 Average acc.	CIFAR-10 Worst-1 acc.	CIFAR-100 Average acc.	CIFAR-100 Worst-1 acc.	TinyImageNet Average acc.	TinyImageNet Worst-1 acc.
Selected α^t, α^s combo	$\mathbf{94.28} \pm 0.06$	$\mathbf{90.11} \pm 0.23$	73.22 ± 0.26	$\mathbf{48.40} \pm 1.47$	$\mathbf{58.09} \pm 0.13$	9.47 ± 1.76
Standard distillation	$\mathbf{94.34} \pm 0.07$	87.66 ± 0.40	$\mathbf{74.61} \pm 0.15$	43.81 ± 0.58	57.83 ± 0.13	6.32 ± 2.31
Post shift [NM'21]	92.16 ± 0.18	88.60 ± 0.35	61.22 ± 0.36	38.19 ± 0.40	43.02 ± 0.79	14.39 ± 1.13
Robust student [NM'21]	92.72 ± 0.05	89.90 ± 0.21	68.45 ± 0.13	43.62 ± 1.27	48.06 ± 0.24	$\mathbf{16.27} \pm 0.43$
AdaMargin [LBMK'22]	93.69 ± 0.06	88.42 ± 0.36	73.58 ± 0.11	43.91 ± 1.11	52.45 ± 0.08	$\mathbf{15.41} \pm 0.71$
AdaAlpha [LBMK'22]	$\mathbf{94.31} \pm 0.01$	88.33 ± 0.14	74.15 ± 0.08	45.46 ± 0.67	57.22 ± 0.08	7.62 ± 2.17
Group DRO [SKHL'20]	92.34 ± 0.07	89.32 ± 0.21	65.18 ± 0.08	43.89 ± 1.12	48.78 ± 0.21	11.38 ± 1.79

Method	CIFAR-10-LT Average acc.	CIFAR-10-LT Worst-1 acc.	CIFAR-100-LT Average acc.	CIFAR-100-LT Worst-1 acc.	TinyImageNet-LT Average acc.	TinyImageNet-LT Worst-10 acc.
Selected α^t, α^s combo	79.02 ± 0.08	$\mathbf{75.43} \pm 0.39$	43.94 ± 0.16	$\mathbf{14.52} \pm 0.68$	26.91 ± 0.16	$\mathbf{6.04} \pm 0.25$
Standard distillation	77.39 ± 0.10	60.12 ± 0.56	$\mathbf{46.01} \pm 0.16$	0.00 ± 0.00	26.05 ± 0.18	0.00 ± 0.00
Post shift [NM'21]	78.28 ± 0.05	74.33 ± 0.09	29.88 ± 0.61	10.01 ± 0.72	21.32 ± 0.49	2.58 ± 0.42
Robust student [NM'21]	80.05 ± 0.13	74.91 ± 0.24	30.79 ± 0.18	12.28 ± 0.46	21.59 ± 0.19	1.55 ± 0.37
Bal. student [MJRJVK'21]	$\mathbf{81.36} \pm 0.14$	71.60 ± 0.38	50.40 ± 0.12	4.39 ± 0.66	$\mathbf{30.43} \pm 0.06$	0.20 ± 0.18
AdaMargin [LBMK'22]	72.69 ± 0.24	47.52 ± 0.95	31.26 ± 0.21	0.00 ± 0.00	4.41 ± 0.09	0.00 ± 0.00
AdaAlpha [LBMK'22]	70.83 ± 0.28	43.64 ± 1.09	42.52 ± 0.08	0.00 ± 0.00	27.95 ± 0.14	0.00 ± 0.00
Group DRO [SKHL'20]	74.39 ± 0.17	59.93 ± 0.59	40.47 ± 0.17	0.19 ± 0.17	27.78 ± 0.13	0.00 ± 0.00

formance: the selected α^t, α^s combination sometimes has lower average test accuracy than standard distillation. Still, overall, the selected α^t, α^s combination is Pareto efficient compared to all other baselines (dominant in at least one of average accuracy or worst-k accuracy). Among the rest of the different α^t, α^s candidates (as in Figure 1), there actually exist combinations that Pareto dominate all baselines in test performance (additional plots in Appendix F). While we only show results from our simple example selection criteria in Table 2, this suggests that there is room for alternative selection criteria to yield even better results. The challenge, as with all hyperparameter selection, is that selection on the validation set comes with a generalization gap between validation and test.

5 THEORETICAL ANALYSIS

Complementing our empirical findings, our theoretical analysis explores what constitutes a good teacher and how it aids a student in achieving robustness. To simplify our exposition, we present our theoretical analysis for a student trained using Algorithm 1 to yield good worst-class performance. Our results easily extend to the case where the student seeks to trade-off between average and worst-case performance.

What constitutes a good teacher? We first characterize the properties of a good teacher when the student's goal is to minimize the robust population objective $L^{\text{rob}}(f^s)$ in (3). In particular, does the student's ability to perform well on this worst-case objective depend on the teacher also performing well on the same objective? Given scores from a teacher p^t, the student minimizes the robust distillation objective $\hat{L}^{\text{rob-d}}(f^s)$ in (7), and uses this as a proxy for the actual objective $L^{\text{rob}}(f^s)$ we care about. Intuitively, an *ideal* teacher would then be one that provides a good proxy for the student, and ensures that the difference $|\hat{L}^{\text{rob-d}}(f^s) - L^{\text{rob}}(f^s)|$ is as small as possible. Below, we provide a simple bound on this difference:

Theorem 2. *Suppose $\ell(y, z) \leq B, \forall x \in \mathcal{X}$ for some $B > 0$. Let $\pi_y^t = \mathbb{E}_x\left[p_y^t(x)\right]$, and let the following denote the per-class expected and empirical student losses respectively:*
$\phi_y(f^s) = \frac{1}{\pi_y^t}\mathbb{E}_x\left[p_y^t(x)\,\ell\left(y, f^s(x)\right)\right]$;
$\hat{\phi}_y(f^s) = \frac{1}{\hat{\pi}_y^t}\frac{1}{n}\sum_{i=1}^n p_y^t(x_i)\,\ell\left(y, f^s(x_i)\right)$.
Then for teacher p^t and student f^s:

$$|\hat{L}^{\text{rob-d}}(f^s) - L^{\text{rob}}(f^s)| \leq \underbrace{B \max_{y\in[m]} \mathbb{E}_x\left[\left|\frac{p_y^t(x)}{\pi_y^t} - \frac{\eta_y(x)}{\pi_y}\right|\right]}_{\textit{Calibration error}} + \underbrace{\max_{y\in[m]}\left|\phi_y(f^s) - \hat{\phi}_y(f^s)\right|}_{\textit{Estimation error}}.$$

The *calibration error* captures how well the teacher's predictions mimic the conditional-class distribution $\eta(x) \in \Delta_m$, up to per-class normalizations π. This suggests that even if p^t does not achieve good worst-class performance, as long as it is *well-calibrated* within each class (as measured by the calibration error), it will serve as a good teacher.

The *estimation error* captures how well the teacher aids in the student's out-of-sample generalization. The prior work by Menon et al. [2021a] study this question in detail for the standard student objective, and provide a bound that depends on the variance induced by the teacher's predictions on the

student's objective: the lower the variance, the better the student's generalization. In Appendix B, we carry out a similar analysis with the estimation error in the theorem.

Calibration and worst-case error. We illustrate how, perhaps counterintuitively, a teacher with low worst-class accuracy might still have scores p^t that are well calibrated to match the true conditional-class distributions η. For this, we use a hypothetical "image classification" task with labels $y \in \{\text{cat}, \text{panda}, \text{other}\}$, and a single one-dimensional feature $x \in [0, 1]$ representing the fraction of black pixels in the image, uniformly distributed over the interval. Suppose the solid lines in Figure 2 below give the conditional-class distributions $\eta_y(x)$ for the cat and panda classes (pandas are rarer than cats in the dataset, with $\pi_{\text{cat}} = \frac{1}{2}$ and $\pi_{\text{panda}} = \frac{1}{4}$). Suppose the dashed lines in Figure 2 also give hypothetical teacher model scores $p^t_y(x)$, where $p^t_{\text{cat}}(x) = 2\eta_{\text{cat}}(x)$, and $p^t_{\text{panda}}(x) = \frac{1}{2}\eta_{\text{panda}}(x)$ (these arbitrary teacher scores do not necessarily correspond to softmax outputs from a neural network). This teacher model always outputs a higher score for the cat label than the panda label. However, the model still satisfies the necessary calibration property: $\frac{p^t_y(x)}{E_x[p^t_y(x)]} = \frac{\eta_y(x)}{\pi_y}$ for $y \in \{\text{cat}, \text{panda}\}$, despite the fact that the argmax predictions from this model has zero recall for the panda class. This illustrates that the important property of the teacher's scores is how well they mimic the *shape* of the conditional-class distributions, and not necessarily their worst-class predictive accuracy.

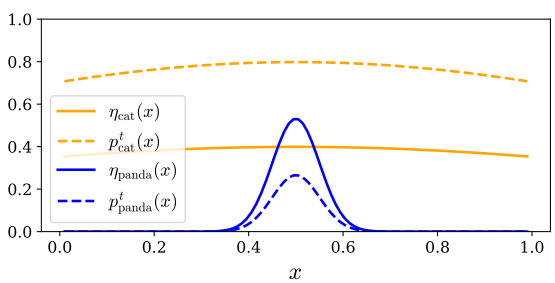

Figure 2: Hypothetical conditional-class distributions $\eta_y(x)$ and trained model scores $p^t_y(x)$ for $y \in \{\text{cat}, \text{panda}\}$.

Relation to Bayes-optimal scorers. When the teacher outputs the conditional-class probabilities, i.e. $p^t(x) = \eta(x)$, the calibration error is trivially zero (recall that the normalization term $\pi^t_y = \pi_y$ in this case). Theorem 1 shows that the Bayes-optimal scorer for the standard cross-entropy loss achieves this; however, in practice with finite data and model class limitations, a teacher trained with the cross-entropy loss is often far from approximating $\eta(x)$ exactly. In practice, it remains an open question what methodology might produce a teacher that most closely mimics these conditional-class distribution shapes for all classes in finite samples. For example, while the standard cross-entropy objective might lead to well calibrated model scores for a majority class, the scores may not match for rare classes. Our experiments explored training with different losses from Section 2 that encourage the teacher to approximate scaled versions of $\eta(x)$; however, future exploration of other practical training possibilities would be interesting to compare.

Robustness guarantee for student. We next provide robustness guarantees for the student output by Algorithm 1 in terms of the calibration and estimation errors described above. We do so for a fixed teacher p^t, and a *self-distillation* setup where the student is chosen from the same function class $\mathcal{F}$ as the teacher, and can thus exactly mimic the teacher's predictions.

Proposition 3. *Suppose $p^t \in \mathcal{F}$ and $\mathcal{F}$ is closed under linear transformations. Let $\bar{\lambda}_y = (\prod_{k=1}^K \lambda^k_y/\pi^t_y)^{1/K}, \forall y$. Then the scoring function $\bar{f}^s(x) = \frac{1}{K}\sum_{k=1}^K f^k(x)$ output by Alg. 1 is of the form:* $\text{softmax}_j(\bar{f}^s(x)) \propto \bar{\lambda}_j p^t_j(x)$, $\forall j \in [m]$, $\forall (x, y) \in S$.

Theorem 4. *Suppose $p^t \in \mathcal{F}$ and $\mathcal{F}$ is closed under linear transformations. Suppose ℓ is the cross-entropy loss ℓ^{xent}, $\ell(y, z) \leq B$ and $\max_{y\in[m]} \frac{1}{\pi^t_y} \leq Z$, for some $B, Z > 0$. Furthermore, suppose for any $\delta \in (0, 1)$, the following bound holds on the estimation error in Theorem 2: with probability at least $1 - \delta$ (over draw of $S \sim D^n$), $\forall f \in \mathcal{F}$, $\max_{y\in[m]} \left|\phi_y(f) - \hat{\phi}_y(f)\right| \leq \Delta(n, \delta)$, for some $\Delta(n, \delta) \in \mathbb{R}_+$ that is increasing in $1/\delta$, and goes to 0 as $n \to \infty$. Then when the step size $\gamma = \frac{1}{2BZ}\sqrt{\frac{\log(m)}{K}}$ and $n^{\text{val}} \geq 8Z\log(2m/\delta)$, we have that with probability at least $1 - \delta$ (over draw of $S \sim D^n$ and $S^{\text{val}} \sim D^{n^{\text{val}}}$),*

$$L^{\text{rob}}(\bar{f}^s) \leq \min_{f\in\mathcal{F}} L^{\text{rob}}(f) + \underbrace{2\Delta(n^{\text{val}}, \delta/2) + 2\Delta(n, \delta/2)}_{\textit{Estimation error}}$$

$$+ \underbrace{2B \max_{y\in[m]} \mathbb{E}_x \left[\left|\frac{p^t_y(x)}{\pi^t_y} - \frac{\eta_y(x)}{\pi_y}\right|\right]}_{\textit{Calibration error}} + \underbrace{4BZ\sqrt{\frac{\log(m)}{K}}}_{\textit{EG convergence}}.$$

Proposition 3 shows the student not only learns to mimic the teacher on the training set, but improves upon it by making per-class adjustments to its predictions. Theorem 4 shows that these adjustments are chosen to close-in on the gap to the optimal robust scorer in $\mathcal{F}$. However, the student's convergence to the optimal scorer in $\mathcal{F}$ would still be limited by the teacher's calibration error: even when the sample sizes and number of iterations $n, n^{\text{val}}, K \to \infty$, the student's optimality gap may still be non-zero when the teacher is poorly calibrated.

Conclusion and future work. Overall, we show empirically and theoretically the value of applying different combinations of teacher/student objectives, not only for improving worst-class accuracy, but also to achieve efficient tradeoffs between average and worst-class accuracy. An interesting future avenue for exploration would be to extend our distillation setup to incorporate other forms of teacher supervision such as intermediate embeddings or ensembled scores (e.g., Iscen et al. [2021]).

References

Rohan Anil, Gabriel Pereyra, Alexandre Tachard Passos, Robert Ormandi, George Dahl, and Geoffrey Hinton. Large scale distributed neural network training through online distillation. 2018. URL https://openreview.net/pdf?id=rkr1UDeC-.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in Neural Information Processing Systems*, 2019a.

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*, 32:1567–1578, 2019b.

Keshigeyan Chandrasegaran, Ngoc-Trung Tran, Yunqing Zhao, and Ngai-Man Cheung. Revisiting label smoothing and knowledge distillation compatibility: What was missing?, 2022. URL https://arxiv.org/abs/2206.14532.

Robert S Chen, Brendan Lucier, Yaron Singer, and Vasilis Syrgkanis. Robust optimization for non-convex objectives. In *NeurIPS*, 2017.

Andrew Cotter, Heinrich Jiang, Serena Wang, Taman Narayan, Seungil You, Karthik Sridharan, and Maya R. Gupta. Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. *Journal of Machine Learning Research (JMLR)*, 20(172): 1–59, 2019.

Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 715–724, 2021.

Jiequan Cui, Shu Liu, Zhuotao Tian, Zhisheng Zhong, and Jiaya Jia. Reslt: Residual learning for long-tailed recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019.

Mengnan Du, Subhabrata Mukherjee, Yu Cheng, Milad Shokouhi, Xia Hu, and Ahmed Hassan Awadallah. What do compressed large language models forget? robustness challenges in model compression, 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*, pages 770–778, 2016.

Yin-Yin He, Jianxin Wu, and Xiu-Shen Wei. Distilling virtual examples for long-tailed recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 235–244, 2021.

G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv:1503.02531*, 2015a.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531, 2015b.

Sara Hooker, Nyalleng Moorosi, Gregory Clark, Samy Bengio, and Emily Denton. Characterising bias in compressed models. *arXiv preprint arXiv:2010.03058*, 2020.

Ahmet Iscen, André Araujo, Boqing Gong, and Cordelia Schmid. Class-balanced distillation for long-tailed visual recognition. *arXiv preprint arXiv:2104.05279*, 2021.

Ganesh Ramachandra Kini, Orestis Paraskevas, Samet Oymak, and Christos Thrampoulidis. Label-imbalanced and group-sensitive classification under overparameterization. In *Advances in Neural Information Processing Systems*, 2021.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. CS 231N, 2015.

Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. Teacher's pet: understanding and mitigating biases in distillation. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=ph3AYXpwEb.

Aditya K Menon, Ankit Singh Rawat, Sashank Reddi, Seungyeon Kim, and Sanjiv Kumar. A statistical perspective on distillation. In *International Conference on Machine Learning*, pages 7632–7642. PMLR, 2021a.

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 (ICLR)*, 2021b.

Rafael Müller, Simon Kornblith, and Geoffrey Hinton. *When Does Label Smoothing Help?* Curran Associates Inc., Red Hook, NY, USA, 2019.

Harikrishna Narasimhan and Aditya K Menon. Training over-parameterized models with non-decomposable objectives. *Advances in Neural Information Processing Systems*, 34, 2021.

Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V Le. Meta pseudo labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11557–11568, 2021.

I. Radosavovic, P. Dollár, R. Girshick, G. Gkioxari, and K. He. Data distillation: Towards omni-supervised learning. In *CVPR*, 2018.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020a.

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

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

Zhiqiang Shen, Zechun Liu, Dejia Xu, Zitian Chen, Kwang-Ting Cheng, and Marios Savvides. Is label smoothing truly incompatible with knowledge distillation: An empirical study. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=PObuuGVrGaZ`.

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

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

Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella Yu. Long-tailed recognition by routing diverse distribution-aware experts. In *International Conference on Learning Representations*, 2020.

Robert C Williamson, Elodie Vernet, and Mark D Reid. Composite multiclass losses. *Journal of Machine Learning Research*, 17:1–52, 2016.

Liuyu Xiang, Guiguang Ding, and Jungong Han. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *European Conference on Computer Vision*, pages 247–263. Springer, 2020.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020.

Canwen Xu, Wangchunshu Zhou, Tao Ge, Ke Xu, Julian McAuley, and Furu Wei. Beyond preserved accuracy: Evaluating loyalty and robustness of BERT compression. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021.

Shaoyu Zhang, Chen Chen, Xiyuan Hu, and Silong Peng. Balanced knowledge distillation for long-tailed learning. *arXiv preprint arXiv:2104.10510*, 2021.