Understanding Self-Distillation in the Presence of Label Noise

Rudrajit Das [1] **Sujay Sanghavi** [1]

Abstract

Self-distillation (SD) is the process of first training a "teacher" model and then using its predictions to train a "student" model that has the *same* architecture. Specifically, the student's loss is $\big(\xi * \ell(\text{teacher's predictions}, \text{student's predictions}) + (1-\xi) * \ell(\text{given labels}, \text{student's predictions})\big)$, where ℓ is the loss function and ξ is some parameter $\in [0,1]$. SD has been empirically observed to provide performance gains in several settings. In this paper, we theoretically characterize the effect of SD in two supervised learning problems with *noisy labels*. We first analyze SD for regularized linear regression and show that in the high label noise regime, the optimal value of ξ that minimizes the expected error in estimating the ground truth parameter is surprisingly greater than 1. Empirically, we show that $\xi > 1$ works better than $\xi \leq 1$ even with the cross-entropy loss for several classification datasets when 50% or 30% of the labels are corrupted. Further, we quantify when optimal SD is better than optimal regularization. Next, we analyze SD in the case of logistic regression for binary classification with random label corruption and quantify the range of label corruption in which the student outperforms the teacher (w.r.t. accuracy). To our knowledge, this is the first result of its kind for the cross-entropy loss.

1. Introduction

The core idea of *knowledge distillation* (KD), introduced in Hinton et al. (2015), is to train a student model with a teacher model's *predicted soft labels* (i.e., the output probability distribution over the classes for classification problems) in addition to the original hard labels (one-hot vectors for classification problems) on which the teacher is trained. The original rationale was to use a teacher with large statistical capacity to better model the underlying label distribution compared to the provided hard labels, and have the student with smaller capacity learn some mixture of the teacher's predicted label distribution (a.k.a. "dark knowledge") and the provided label distribution. Specifically, the student's per-sample loss in the KD framework is:

$$\xi * \ell\big(\boldsymbol{y}_T, \boldsymbol{y}_S(\boldsymbol{\theta})\big) + (1-\xi) * \ell\big(\boldsymbol{y}, \boldsymbol{y}_S(\boldsymbol{\theta})\big), \qquad (1)$$

where ℓ is some loss function (usually, regularized cross-entropy loss for classification problems), $\boldsymbol{y}_T$ is the teacher's predicted label, $\boldsymbol{y}$ is the given label on which the teacher is trained, $\boldsymbol{y}_S(\boldsymbol{\theta})$ is the prediction of the student model parameterized by $\boldsymbol{\theta}$, and $\xi \in [0,1]$ is known as the imitation parameter (Lopez-Paz et al., 2015)[1]. KD and its variants have been shown to be beneficial for model compression (i.e., distilling a bigger teacher model's knowledge into a smaller student model), making models robust and improving performance in general (Li et al., 2017; Furlanello et al., 2018; Sun et al., 2019; Ahn et al., 2019; Chen et al., 2020; Xie et al., 2020; Sarfraz et al., 2021; Li et al., 2021; Beyer et al., 2022; Yang et al., 2022); see Gou et al. (2021) for a survey on KD.

The focus of this work is on the special case of the student and teacher having the same architecture, which is known as **self-distillation** (following Mobahi et al. (2020)); we abbreviate it as **SD** henceforth. Since the teacher and student have the same capacity, one would expect the utility of the teacher's dark knowledge to be very limited, if any at all. However, surprisingly, Furlanello et al. (2018) show that SD (with ensembling) yields performance gains in both vision and language tasks with extensive experiments. Further, Li et al. (2017) empirically demonstrate that SD can ameliorate learning in the presence of noisy labels. There are also a few works that theoretically investigate SD, such as Mobahi et al. (2020); Dong et al. (2019); we discuss these in detail in Section 2. The results of these papers are only with the squared loss and *not* the *cross-entropy loss* which is the de facto loss function for classification problems.

In this work, we theoretically analyze SD in the presence of label corruption (in the supervised setting) for the cross-entropy loss as well as the squared loss, characterizing its

[1]UT Austin. Correspondence to: Rudrajit Das <rdas@utexas.edu>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

[1]In this work, we set the temperature parameter suggested in Hinton et al. (2015) equal to 1.

utility and unveiling some new insights including a recommendation for use in practice. We summarize our contributions next and survey the landscape of pertinent theoretical works on KD and SD in Section 2.

Contributions:

(a) First, we consider linear regression with ℓ_2-regularized squared loss in Section 3. Here, the observed label y for a sample $\boldsymbol{x}$ is: $y = \langle \boldsymbol{\theta}^*, \boldsymbol{x} \rangle + \eta$, where $\boldsymbol{\theta}^*$ is the underlying parameter and η is zero-mean random label noise.

- We show that self-distillation (SD) is associated with a **bias-variance tradeoff** in that increasing ξ in Eq. (1) reduces the variance but increases the bias in estimating $\boldsymbol{\theta}^*$ w.r.t. the randomness in label noise; see Theorem 3.2 and Remark 3.3.
- A **surprising algorithmic insight** from our analysis is that the value of ξ that optimally balances this bias-variance tradeoff can be > 1, especially in the *high label noise regime* (i.e., when $\mathbb{E}[\eta^2]$ is large); see Corollary 3.5 and Remark 3.6. This can be interpreted as actively **anti-learning** (or going against) the observed (possibly noisy) labels. But as discussed after Eq. (1), ξ is tuned in $[0, 1]$ in practice. In Section 5 (Table 1), we empirically corroborate our insight for multi-class classification with linear probing[2] using the *cross-entropy* loss by showing that $\xi > 1$ works better than $\xi \leq 1$ for several datasets with 50% or 30% of the training set's labels being corrupted in different ways.
- In Remark 3.7, we show that as the degree of label noise increases, the utility of the teacher's predictions in training the student increases. Intuitively, this happens because the noise component in the teacher's predictions is smaller compared to the original labels. We also empirically verify this insight for the *cross-entropy* loss in Section 5 (Table 3).
- In Theorem 3.8, we characterize when *optimal* SD is **better** than *optimal* ℓ_2 regularization (optimal means with the best parameters); this is the **first** such result.

(b) Next, we look at logistic regression with ℓ_2-regularized cross-entropy loss in Section 4. We consider a binary classification problem where some fraction, say $p < 0.5$, of the training set's labels are randomly flipped. Under some assumptions on the data geometry and the kernel function, *we quantify the range of p in which the student outperforms the teacher in terms of accuracy*; see Theorem 4.3. To our knowledge, this is the **first result** that *provably establishes the utility of SD in the presence of label noise for the **cross-entropy** loss*. The main technical challenge in the analysis is dealing with non-linear equations involving the sigmoid function. We tackle this by employing the first-order Maclaurin series expansion of the sigmoid function and by bounding the corresponding approximation errors; see the proof outline of Theorem 4.3.

[2] i.e., learning a softmax layer on top of a pre-trained network

2. Related Work

There is a growing body of works trying to theoretically explain KD/SD and its benefits. Mobahi et al. (2020) look at regression with the squared loss in Hilbert space, showing that SD essentially amplifies regularization. However, unlike us, they do not explicitly consider the case of noisy labels/observations or discuss the bias-variance tradeoff associated with SD in the presence of label noise. Moreover, they restrict their analysis to $\xi = 1$; so unlike us, they do not have any results on when optimal SD is better than optimal ℓ_2 regularization. Dong et al. (2019) claim that KD is effective in transferring dark knowledge by mimicking early stopping. Further, they propose their own SD algorithm that uses dynamically updated soft labels, and show that in the presence of noisy labels, their algorithm is able to learn the correct labels. In this work, we focus on the standard SD algorithm with fixed soft labels, and moreover, we quantify the range of label corruption in which SD improves accuracy. Unlike our work, Dong et al. (2019) do not quantify when their proposed algorithm improves upon the standard approach of using just hard labels. An important difference between our work and Dong et al. (2019) as well as Mobahi et al. (2020) is that the results of these two papers are with the squared loss, whereas we provide results with the cross-entropy loss in addition to squared loss. The cross-entropy loss is the customary choice for classification problems in practice and is also more challenging to analyze. On the note of cross-entropy loss, Phuong & Lampert (2019) analyze the convergence of linear student networks trained with the cross-entropy loss, and also bound the expected difference between the predictions of the student and teacher. Ji & Zhu (2020) also bound the expected difference between the predictions of the student and teacher for wide neural networks that evolve as linear networks under the NTK assumption. However, Phuong & Lampert (2019) and Ji & Zhu (2020) do not consider how the student might have better generalization than the teacher in the presence of noisy labels. Menon et al. (2021) statistically characterize "good" teachers for distilling knowledge to a student. Kaplun et al. (2022) show that an ensemble of teachers trained with noisy labels can be used to label a new unlabeled dataset, which can be then employed to train a student with good performance. We focus on the (common) case of only one teacher and the student being trained on the same dataset as the teacher. There are also some works such as Cheng et al. (2020); Stanton et al. (2021); Zhou et al. (2021); Pham et al. (2022) that empirically provide some insights on KD. Moreover, there are some similarities between KD and label smoothing (Yuan et al., 2020); we discuss this and also delineate some key differences between them in Appendix A.

3. Linear Regression

Setting: The *observed* label $y \in \mathbb{R}$ is linearly related to the data $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ as $y = \langle \boldsymbol{\theta}^*, \boldsymbol{x} \rangle + \eta$, where $\boldsymbol{\theta}^* \in \mathbb{R}^d$ and $\eta \in \mathbb{R}$ is label noise. Here, $\langle \boldsymbol{\theta}^*, \boldsymbol{x} \rangle$ is the *actual* label of $\boldsymbol{x}$.

The training set consists of n pairs of data points (drawn from $\mathcal{X}$) and noisy labels $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$. Let $\boldsymbol{X} := [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d \times n}$ be the data matrix and $\boldsymbol{Y} := [y_1, \ldots, y_n]^T \in \mathbb{R}^n$ be the label vector. Then, as per the above linear model, $\boldsymbol{Y} = \boldsymbol{X}^T \boldsymbol{\theta}^* + \boldsymbol{\eta}$, for some noise vector $\boldsymbol{\eta} \in \mathbb{R}^n$. We make some standard assumptions on $\boldsymbol{\eta}$.

Assumption 3.1 (Label Noise). *$\boldsymbol{\eta}$ is independent of $\boldsymbol{X}$. Further, each coordinate of $\boldsymbol{\eta}$ has mean 0 and variance γ^2, and is independent of the other coordinates.*

Teacher Model: The teacher tries to learn the underlying model, parameterized by $\boldsymbol{\theta} \in \mathbb{R}^d$, from $(\boldsymbol{X}, \boldsymbol{Y})$ by applying the squared loss with ℓ_2 regularization. Specifically, the teacher's objective function is $f_T(\boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{X}^T\boldsymbol{\theta}\|^2 + \frac{\lambda}{2}\|\boldsymbol{\theta}\|^2$, where $\lambda > 0$ is the ℓ_2-regularization parameter. Now, the *model learned by the teacher* is[3]:

$$\hat{\boldsymbol{\theta}}_T := \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^d} f_T(\boldsymbol{\theta}) = (\boldsymbol{X}\boldsymbol{X}^T + \lambda \boldsymbol{I}_d)^{-1}\boldsymbol{X}\boldsymbol{Y}, \quad (2)$$

where $\boldsymbol{I}_d$ is the identity matrix of size $d \times d$. Substituting $\boldsymbol{Y} = \boldsymbol{X}^T\boldsymbol{\theta}^* + \boldsymbol{\eta}$ in Equation (2), we get:

$$\hat{\boldsymbol{\theta}}_T = (\boldsymbol{X}\boldsymbol{X}^T + \lambda \boldsymbol{I}_d)^{-1}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{\theta}^* + \boldsymbol{\eta}). \quad (3)$$

Student Model Trained with Self-Distillation: Following Equation (1), here the student is trained with a weighted sum of (i) the ℓ_2-regularized squared loss between the student's predictions and the *teacher's predictions*, and (ii) the ℓ_2-regularized squared loss between the student's predictions and the *original labels* on which the teacher was trained. For the ith sample, the teacher's prediction is $\hat{y}_i = \langle \hat{\boldsymbol{\theta}}_T, \boldsymbol{x}_i \rangle$. Define $\hat{\boldsymbol{Y}} := [\hat{y}_1, \ldots, \hat{y}_n]^T \in \mathbb{R}^n$; note that $\hat{\boldsymbol{Y}} = \boldsymbol{X}^T\hat{\boldsymbol{\theta}}_T$.

Let $\ell(\boldsymbol{Z}, \boldsymbol{\theta}) := \frac{1}{2}\|\boldsymbol{Z} - \boldsymbol{X}^T\boldsymbol{\theta}\|^2 + \frac{\lambda}{2}\|\boldsymbol{\theta}\|^2$, where λ is the teacher's regularization parameter. Then, the student's objective function is $f_S(\boldsymbol{\theta}; \xi) = \xi\ell(\hat{\boldsymbol{Y}}, \boldsymbol{\theta}) + (1-\xi)\ell(\boldsymbol{Y}, \boldsymbol{\theta})$, where $\xi \in \mathbb{R}$ is known as the imitation parameter (Lopez-Paz et al., 2015). Even though it is standard practice to restrict $\xi \in [0, 1]$, we do not impose this condition. Now, the *model learned by the student* is:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_S(\xi) &:= \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^d} f_S(\boldsymbol{\theta}; \xi) \\ &= \Big(\xi(\boldsymbol{X}\boldsymbol{X}^T + \lambda \boldsymbol{I}_d)^{-1}\boldsymbol{X}\boldsymbol{X}^T + (1-\xi)\boldsymbol{I}_d\Big)\hat{\boldsymbol{\theta}}_T, \quad (4)\end{aligned}$$

where $\hat{\boldsymbol{\theta}}_T$ (in Equation (3)) is the teacher's model. We include a short proof for Equation (4) in Appendix B. Note that $\xi = 0$ corresponds to the teacher, i.e. $\hat{\boldsymbol{\theta}}_S(0) = \hat{\boldsymbol{\theta}}_T$.

[3]We assume that we can converge to the exact optimum of the objective function. All the objective functions in this work are convex, and hence (S)GD will converge to the optimum.

3.1. Estimation Error Comparison

Let us denote the student's error in estimating the ground truth parameter $\boldsymbol{\theta}^*$ with imitation parameter ξ as $\boldsymbol{\epsilon}_S(\xi) := \hat{\boldsymbol{\theta}}_S(\xi) - \boldsymbol{\theta}^*$. Note that $\boldsymbol{\epsilon}_S(0) := \hat{\boldsymbol{\theta}}_S(0) - \boldsymbol{\theta}^* = \hat{\boldsymbol{\theta}}_T - \boldsymbol{\theta}^*$ is the teacher's estimation error. We shall analyze the expected squared norm of the estimation error w.r.t. the random label noise $\boldsymbol{\eta}$, i.e. $\mathbb{E}_{\boldsymbol{\eta}}[\|\boldsymbol{\epsilon}_S(\xi)\|^2]$, as a function of ξ.[4] It will be illustrative to do so in terms of the SVD of $\boldsymbol{X}$. Let $\text{rank}(\boldsymbol{X}) = r$ (note that $r \leq \min(d, n)$) and the SVD decomposition of $\boldsymbol{X}$ be $\sum_{j=1}^r \sigma_j \boldsymbol{u}_j \boldsymbol{v}_j^T$, where $\sigma_1 \geq \ldots \geq \sigma_r > 0$, and each $\boldsymbol{u}_j \in \mathbb{R}^d$ and $\boldsymbol{v}_j \in \mathbb{R}^n$. Also, let $\{\boldsymbol{u}_j\}_{j=1}^d$ be the full set of left singular vectors of $\boldsymbol{X}$ (i.e., even those corresponding to the zero singular values); note that this forms an orthonormal basis for $\mathbb{R}^d$.

Following standard bias-variance decomposition, we have:

$$\mathbb{E}_{\boldsymbol{\eta}}\Big[\big\|\boldsymbol{\epsilon}_S(\xi)\big\|^2\Big] = \underbrace{\big\|\mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\epsilon}_S(\xi)]\big\|^2}_{\text{squared bias}} + \underbrace{\mathbb{E}_{\boldsymbol{\eta}}\Big[\big\|\boldsymbol{\epsilon}_S(\xi) - \mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\epsilon}_S(\xi)]\big\|^2\Big]}_{\text{variance}}. \quad (5)$$

In Thm. 3.2, we shall quantify the squared bias and variance in Eq. (5) as a function of ξ; this is proved in Appendix C.

Theorem 3.2 (Bias^2 and Variance). *Suppose Assumption 3.1 holds. Let $\theta_j^* := (\langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle)^2$. Then,*

(i) *the squared bias $\big\|\mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\epsilon}_S(\xi)]\big\|^2$ is:*

$$\sum_{j=1}^r \theta_j^* \left(\frac{\lambda/\sigma_j^2}{1 + \lambda/\sigma_j^2}\right)^2 \left(1 + \frac{\xi}{1 + \lambda/\sigma_j^2}\right)^2 + \sum_{j=r+1}^d \theta_j^*. \quad (6)$$

(ii) *the variance $\mathbb{E}_{\boldsymbol{\eta}}\Big[\big\|\boldsymbol{\epsilon}_S(\xi) - \mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\epsilon}_S(\xi)]\big\|^2\Big]$ is:*

$$\frac{\gamma^2}{\lambda}\left\{\sum_{j=1}^r \frac{\lambda/\sigma_j^2}{\big(1 + \lambda/\sigma_j^2\big)^2}\left(1 - \xi\left(\frac{\lambda/\sigma_j^2}{1 + \lambda/\sigma_j^2}\right)\right)^2\right\}, \quad (7)$$

where γ^2 is the label noise variance (as per Assumption 3.1).

Remark 3.3 (Bias-Variance Tradeoff as a Function of ξ). *Let us restrict our attention to $\xi \in [0, 1]$ which is the range of ξ used in practice (Lopez-Paz et al., 2015; Li et al., 2017; Sun et al., 2019). From Equation (6), note that the bias increases as the student tries to imitate the teacher more, i.e., as ξ increases. However, from Equation (7), the variance (due to label noise) reduces as the student tries to imitate the teacher more. Thus, SD is associated with a* ***bias-variance tradeoff*** *– a higher value of ξ mitigates the impact of label noise variance at the cost of increasing the estimation bias (and vice versa).*

[4]We do not analyze the expected squared prediction error, i.e. $\mathbb{E}_{\boldsymbol{\eta}, \boldsymbol{x}}\big[\big(\langle \hat{\boldsymbol{\theta}}_S(\xi), \boldsymbol{x} \rangle - \langle \boldsymbol{\theta}^*, \boldsymbol{x} \rangle\big)^2\big]$, because that would force us to make assumptions on the distribution of $\boldsymbol{x}$ (the data) as well. However, it is worth noting that with the standard assumption of $\boldsymbol{x} \sim \mathcal{N}(\vec{0}_d, \mathrm{I}_d)$, the expected squared prediction error is the same as the expected squared norm of the error in estimating $\boldsymbol{\theta}^*$.

Remark 3.4 (**Extension to General Settings**). *Zhou et al. (2021) propose an algorithm **assuming** KD reduces variance compared to standard training in general settings **without any formal proof**. Their algorithm is validated by extensive experiments; this in turn corroborates the validity of the assumption and suggests that the variance-reduction effect of KD holds in more general settings than linear regression. Ours is the **first** work to explicitly **prove** this insight in any setting.*

Putting Eqs. (6) and (7) in Eq. (5), we obtain $\mathbb{E}_{\boldsymbol{\eta}}[\|\boldsymbol{\epsilon}_S(\xi)\|^2]$; note that it is a quadratic function of ξ. Corollary 3.5 provides the optimal value of ξ, say ξ^*, that minimizes $\mathbb{E}_{\boldsymbol{\eta}}[\|\boldsymbol{\epsilon}_S(\xi)\|^2]$ (obtained by simple differentiation).

Corollary 3.5. *Define $\xi^* := \arg\min_{\xi\in\mathbb{R}} \mathbb{E}_{\boldsymbol{\eta}}[\|\boldsymbol{\epsilon}_S(\xi)\|^2]$. Let $c_j := \lambda/\sigma_j^2$ and recall $\theta_j^* := (\langle \boldsymbol{\theta}^*, \boldsymbol{u}_j\rangle)^2$. Then:*

$$\xi^* = \frac{\sum_{j=1}^r \left(\frac{\gamma^2}{\lambda} - \theta_j^*\right)\frac{c_j^2}{(1+c_j)^3}}{\sum_{j=1}^r \left(\frac{\gamma^2}{\lambda}c_j + \theta_j^*\right)\frac{c_j^2}{(1+c_j)^4}}. \tag{8}$$

Thus, setting $\xi = \xi^*$ yields the optimal balance between the squared bias and variance.

Remark 3.6 (**Anti-Learning Observed Labels in Noisy Settings**). *There are scenarios when ξ^* obtained in Corollary 3.5 is more than 1[5], especially when γ is large, i.e., there is a lot of label noise. For e.g., note that $\lim_{\gamma\to\infty}\xi^* = \frac{\sum_{j=1}^r c_j^2/(1+c_j)^3}{\sum_{j=1}^r c_j^3/(1+c_j)^4} > 1$[6]. However, the imitation parameter ξ is restricted to and tuned in $[0,1]$ (Lopez-Paz et al., 2015; Li et al., 2017; Sun et al., 2019). Based on our analysis, we advocate not restricting $\xi \in [0,1]$ and also trying $\xi > 1$ in the high noise regime. Setting $\xi > 1$ can be interpreted as "anti-learning" (or going against) the observed labels.*

In Sec. 5, we provide empirical evidence showing that $\xi > 1$ works better than $\xi \leq 1$ even in classification problems with the *cross-entropy loss* for several noisy datasets; see Table 1.

Remark 3.7 (**Utility of Teacher's Predictions**). *In Appendix D, we show that ξ^* is an increasing function of the label noise variance γ^2, i.e., we should assign more weight to the teacher's predicted labels as γ^2 increases. So in linear regression, the benefit of using teacher's predictions (the core idea of SD) increases with the degree of label noise.*

We make a similar observation in our classification experiments in Section 5 (Table 3) where SD with $\xi = 1$ – i.e., *only using the teacher's predictions* (and completely ignoring the given labels) – consistently yields higher gains (over the teacher) as the amount of label corruption increases.

[5] ξ^* can be negative too, but we shall not focus on this case.

[6] $\sum_j \frac{c_j^3}{(1+c_j)^4} = \sum_j \underbrace{\left(\frac{c_j}{1+c_j}\right)}_{<1}\left(\frac{c_j^2}{(1+c_j)^3}\right) < \sum_j \frac{c_j^2}{(1+c_j)^3}$.

Is Optimal Self-Distillation Better than Optimal ℓ_2 Regularization? Let $e(\lambda,\xi) := \mathbb{E}_{\boldsymbol{\eta}}\left[\|\boldsymbol{\epsilon}_S(\xi)\|^2\right]$ (recall $\boldsymbol{\epsilon}_S(\xi)$ is a function of the ℓ_2-regularization parameter λ too). Since $\xi = 0$ corresponds to using just ℓ_2 regularization, we define $e_{\text{reg}}(\lambda) := e(\lambda, 0)$ as the estimation error obtained using only ℓ_2 regularization (and no SD) with parameter λ. Next, let us define $e_{\text{sd}}(\lambda)$ as the error obtained using SD with ℓ_2-regularizer = λ and the optimal value of $\xi = \xi^*$ from Corollary 3.5 (which is itself a function of λ), i.e., $e_{\text{sd}}(\lambda) := e(\lambda, \xi^*)$. By definition, $e_{\text{sd}}(\lambda) \leq e_{\text{reg}}(\lambda)\ \forall\ \lambda$; we wish to know when and if $\min_\lambda e_{\text{sd}}(\lambda) < \min_\lambda e_{\text{reg}}(\lambda)$ (note the **strict** inequality), i.e., when and if **optimal** SD is better than **optimal** ℓ_2 regularization *by tuning over λ*.

Theorem 3.8. *Let $\lambda^*_{reg} := \arg\min_\lambda e_{\text{reg}}(\lambda)$. It holds that $e_{\text{sd}}(\lambda^*_{reg}) = e_{\text{reg}}(\lambda^*_{reg})$ and $\frac{de_{\text{sd}}(\lambda)}{d\lambda}\big|_{\lambda=\lambda^*_{reg}} = 0$, i.e., λ^*_{reg} is a stationary point of $e_{\text{sd}}(\lambda)$ also. It is a **local maximum** point of $e_{\text{sd}}(\lambda)$ when:*

$$\sum_{k=1}^r \sum_{j=1}^{k-1} \frac{\sigma_j^2\sigma_k^2(\sigma_j^2 - \sigma_k^2)(\theta_k^* - \theta_j^*)}{(\lambda^*_{reg} + \sigma_j^2)^4(\lambda^*_{reg} + \sigma_k^2)^4} < 0, \tag{9}$$

with $\theta_j^* := (\langle \boldsymbol{\theta}^*, \boldsymbol{u}_j\rangle)^2$. When the above holds, optimal self-distillation is better than optimal ℓ_2-regularization.

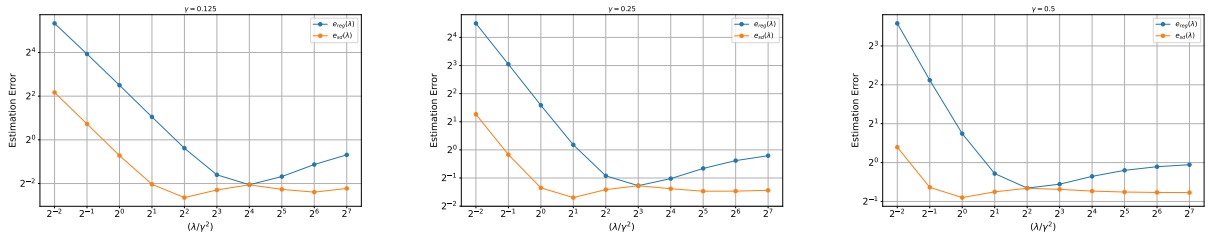
The detailed version and proof of Theorem 3.8 appear in Appendix E. One case when Eq. (9) holds is $\theta_1^* > \ldots > \theta_r^*$ (since $\sigma_1 \geq \ldots \geq \sigma_r$). In general, when the squared projections of $\boldsymbol{\theta}^*$ along the most significant left singular vectors of $\boldsymbol{X}$ (i.e., the ones with "large" singular values) follow the same ordering as the corresponding singular values and the noise variance is large enough, λ^*_{reg} will be a local maximum point of $e_{\text{sd}}(\lambda)$. We formalize this next.

Theorem 3.9. *W.l.o.g., let $\|\boldsymbol{\theta}^*\| = 1$ and $\sigma_1 = 1$. Further, suppose $\sigma_j \leq \delta$ for $j \in \{q+1, \ldots, r\}$ and $\theta_1^* > \ldots > \theta_q^*$. Then, $\lambda = \lambda^*_{reg}$ is a **local maximum** point of $e_{\text{sd}}(\lambda)$ when $\delta \leq \mathcal{O}(\frac{1}{r})$ and $\gamma^2 \geq \frac{\max_{j\in\{1,\ldots,r\}}\theta_j^*}{r-1}$.*

The detailed statement and proof of Theorem 3.9 are in Appendix F. In practice, $\boldsymbol{X}$ is usually low rank and only a few of its singular values are large. So, the assumption of Theorem 3.9 is realistic and that too with $q \ll r$.

To the best of our knowledge, **there are no results** comparable to Theorems 3.8 and 3.9 quantifying when **optimal** SD is better than **optimal** ℓ_2 regularization. Now we consider a *synthetic example* to verify the previous discussion. Suppose $\boldsymbol{\theta}^* = \frac{1}{\sqrt{2}}(\boldsymbol{u}_1 + \boldsymbol{u}_2)$, $n > d = 100$ and $\sigma_j = \frac{1}{j}$ for $j \in \{1, \ldots, d\}$ (so only few singular values are large). Note that Eq. (9) is satisfied. We consider 3 values of $\gamma = \{0.125, 0.25, 0.5\}$ & 10 values of $\lambda = \{2^{i-3}\gamma^2\}$ with $i \in \{1, \ldots, 10\}$. In Figure 1, we plot $e_{\text{reg}}(\lambda)$ and $e_{\text{sd}}(\lambda)$ for these values of γ and λ; see the figure caption for discussion.

If $e_{\text{sd}}(\lambda)$ doesn't have a local maximum at λ^*_{reg}, it is hard to say if λ^*_{reg} is a sub-optimal *local* minimum or the *global*



(a) $\gamma = 0.125$ (b) $\gamma = 0.25$ (c) $\gamma = 0.5$

Figure 1. $e_{\text{reg}}(\lambda)$ and $e_{\text{sd}}(\lambda)$ vs. λ for the synthetic example at the end of Sec. 3. As per Thm. 3.8, note that the global minimum of $e_{\text{reg}}(\lambda)$ is a local maximum of $e_{\text{sd}}(\lambda)$. Observe that $\min_\lambda e_{\text{sd}}(\lambda) < \min_\lambda e_{\text{reg}}(\lambda)$. So, optimal SD does better than optimal ℓ_2-regularization here.

minimum point of $e_{\text{sd}}(\lambda)$; also see Appendix E. If λ^*_{reg} is the global minimum point of $e_{\text{sd}}(\lambda)$, then optimal SD is **not** better than optimal regularization as $e_{\text{sd}}(\lambda^*_{\text{reg}}) = e_{\text{reg}}(\lambda^*_{\text{reg}})$. To complement this, we present Thm. 3.10 (proved in Appendix G).

Theorem 3.10. *There exists $\boldsymbol{\theta}^*$ and $\boldsymbol{X}$ s.t. for any noise variance γ^2, λ^*_{reg} is the **global minimum** point of $e_{\text{sd}}(\lambda)$.*

Significance of Theorems 3.8-3.10: The conventional belief based on Furlanello et al. (2018) is that a student trained with SD generally outperforms the teacher; Theorems 3.8 and 3.9 characterize when this is true in linear regression and Theorem 3.10 is a negative result showing that SD does not always lead to improvement. So unlike most other results that focus on the improvements offered by KD/SD, our results point out some limitations of SD.

It is worth mentioning that all our results and insights here carry over to wide deep neural networks that evolve as linear models (under gradient descent) (Lee et al., 2019).

4. Logistic Regression

We now move onto logistic regression with the cross-entropy loss. Note that *linear probing* (Alain & Bengio, 2016; Kumar et al., 2022) is the same as logistic regression with features obtained from a pre-trained model. It is worth mentioning that our analysis for logistic regression is significantly different from and harder than linear regression.

Setting: We consider a binary classification problem where each sample $\boldsymbol{x} \in \mathcal{X}$ has a discrete label $y(\boldsymbol{x}) \in \{0, 1\}$. Let the marginal distribution of the sample space (with support $\mathcal{X}$) be denoted by $\mathcal{P}$. We assume that there is a feature map $\phi : \mathcal{X} \to \widetilde{\mathcal{X}}$, where $\widetilde{\mathcal{X}}$ is some Hilbert space, and we have access to a sample in terms of its features (in this Hilbert space). We are given $2n$ pairs of data points in terms of features and **corrupted** labels $\{(\phi(\boldsymbol{x}_i), \hat{y}_i)\}_{i=1}^{2n}$, where each $\hat{y}_i \in \{0, 1\}$ and $\boldsymbol{x}_i \underset{\text{iid}}{\sim} \mathcal{P}$. Let the corresponding **actual** labels be $\{y_i\}_{i=1}^{2n}$; we assume that the dataset is balanced, i.e., $|i : y_i = 1| = |i : y_i = 0| = n$. Specifically, without loss of generality (w.l.o.g.), let $y_i = 1$ for $i \in \mathcal{S}_1 := \{1, \ldots, n\}$ and $y_i = 0$ for $i \in \mathcal{S}_0 := \{n+1, \ldots, 2n\}$; our training algorithms are not privy to this. We consider the following corruption model: $\hat{n} < n/2$ samples of *each* class, chosen *randomly*, are provided to us with flipped labels (again, our training algorithms are not privy to this). Specifically, w.l.o.g., let:

$$\hat{y}_i = \begin{cases} 1 - y_i \text{ for } i \in \underbrace{\{1, \ldots, \hat{n}\}}_{:=\mathcal{S}_{1,\text{bad}}} \cup \underbrace{\{n+1, \ldots, n+\hat{n}\}}_{:=\mathcal{S}_{0,\text{bad}}}, \\ y_i \text{ for } i \in \underbrace{\{\hat{n}+1, \ldots, n\}}_{:=\mathcal{S}_{1,\text{good}}} \cup \underbrace{\{n+\hat{n}+1, \ldots, 2n\}}_{:=\mathcal{S}_{0,\text{good}}}. \end{cases}$$

Define $p := \frac{\hat{n}}{n} < \frac{1}{2}$ as the **label corruption fraction**.

Our goal is to learn a separator for the data *w.r.t. the actual labels* by training a logistic regression model on $\{(\phi(\boldsymbol{x}_i), \hat{y}_i)\}_{i=1}^{2n}$. Specifically, for a sample $\boldsymbol{x}$ with feature $\phi(\boldsymbol{x}) \in \widetilde{\mathcal{X}}$, the prediction for the label $y(\boldsymbol{x}) \in \{0, 1\}$ is modeled as $\mathbb{P}(y(\boldsymbol{x}) = 1) = \sigma(\langle \boldsymbol{\theta}, \phi(\boldsymbol{x}) \rangle)$[7], where $\boldsymbol{\theta} \in \widetilde{\mathcal{X}}$ is the parameter that we wish to learn, and $\sigma(z) = \frac{1}{1+e^{-z}}$ for $z \in \mathbb{R}$ is the sigmoid function. We use the binary cross-entropy loss for training; we denote this by BCE : $[0, 1] \times (0, 1) \to \mathbb{R}_{\geq 0}$ and it is defined as:

$$\text{BCE}(q, \hat{q}) = -\big(q \log(\hat{q}) + (1 - q) \log(1 - \hat{q})\big). \quad (10)$$

Next, we state our assumptions on the feature map $\phi(.)$.

Assumption 4.1 (Orthonormality). *The features have unit norm, i.e., $\|\phi(\boldsymbol{x})\|_2 = 1 \; \forall \; \boldsymbol{x} \in \mathcal{X}$. Further, the space of samples in feature space with labels 0 and 1 are orthogonal, i.e., $\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle = 0 \; \forall \; \boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}^2$ with **different** labels.*

Assumption 4.1[8] ensures that the data is separable and indeed there exists a separator.

[7] The bias term can be absorbed within the feature vector $\phi(.)$.

[8] Even with an assumption like $\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle < -u$ for some $u \in (0, 1)$ where $\boldsymbol{x}$ and $\boldsymbol{x}'$ have *different* labels, we can get a result similar to our main result, viz., Theorem 4.3. However, the final bound and proof will be more complicated with such an assumption. Note that even with the perfect orthogonality assumption, the proof of Theorem 4.3 is highly non-trivial.

Assumption 4.2 (Feature Correlation in the Training Set). *$\langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_{i'}) \rangle = c \in (0,1)\ \forall\ i \neq i'$ such that $y_i = y_{i'}$.*

It is true that at face value, Assumption 4.2 seems strong. Instead, an assumption in expectation like $\mathbb{E}_{\boldsymbol{x},\boldsymbol{x}'}\left[\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle \middle| \boldsymbol{x} \text{ and } \boldsymbol{x}' \text{ have the same label}\right] = c$ is more realistic; let us call this Assumption 4.2′ for the sake of discussion. For $n \to \infty$ and when the labels are *corrupted randomly*, we hypothesize that the average[9] prediction (i.e., soft score $\in (0,1)$ assigned to a class) of a model under Assumption 4.2′ is the same as that under Assumption 4.2. We provide empirical evidence to support this hypothesis in Appendix H. Thus, for large n, we argue that Assumption 4.2 is reasonable and an important case to analyze.

Teacher Model: The teacher minimizes the ℓ_2-regularized binary cross-entropy loss with the provided labels as its targets, i.e., the teacher's objective is:

$$f_{\text{T}}(\boldsymbol{\theta}) = \frac{1}{2n}\sum_{i=1}^{2n} \text{BCE}\Big(\hat{y}_i, \sigma\big(\langle \boldsymbol{\theta}, \phi(\boldsymbol{x}_i) \rangle\big)\Big) + \frac{\lambda \|\boldsymbol{\theta}\|^2}{2}.$$

$\lambda > 0$ is the ℓ_2-regularization parameter above. The teacher's estimated parameter is $\boldsymbol{\theta}^*_{\text{T}} := \arg\min_{\boldsymbol{\theta}} f_{\text{T}}(\boldsymbol{\theta})$. The teacher's predicted *soft* label for the ith sample is $y_i^{(\text{T})} := \sigma(\langle \boldsymbol{\theta}^*_{\text{T}}, \phi(\boldsymbol{x}_i) \rangle)$; these are used to train the student.

Student Model Trained Only with Teacher's Soft Labels: Here we set $\xi = 1$. Thus, the student minimizes the ℓ_2-regularized binary cross-entropy loss with the teacher's predicted *soft* labels as its targets, i.e., student's objective is:

$$f_{\text{S}}(\boldsymbol{\theta}) = \frac{1}{2n}\sum_{i=1}^{2n} \text{BCE}\Big(y_i^{(\text{T})}, \sigma\big(\langle \boldsymbol{\theta}, \phi(\boldsymbol{x}_i) \rangle\big)\Big) + \frac{\lambda \|\boldsymbol{\theta}\|^2}{2}.$$

λ is the same ℓ_2-regularization parameter used by the teacher. The student's estimated parameter is $\boldsymbol{\theta}^*_{\text{S}} := \arg\min_{\boldsymbol{\theta}} f_{\text{S}}(\boldsymbol{\theta})$.

4.1. Comparison of Student and Teacher

We now characterize the conditions under which the student outperforms the teacher w.r.t. classification accuracy; to our knowledge, this is the **first result** of its kind. For completeness, the teacher's population accuracy is defined as $100 * \mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}}\left[\mathbb{1}\left(y(\boldsymbol{x}) = \mathbb{1}\left(\sigma\big(\langle \boldsymbol{\theta}^*_{\text{T}}, \phi(\boldsymbol{x}) \rangle\big) > \frac{1}{2}\right)\right)\right]\%$[10]. The student's accuracy is defined similarly with $\boldsymbol{\theta}^*_{\text{S}}$ replacing $\boldsymbol{\theta}^*_{\text{T}}$.

Theorem 4.3 (When is Student's Accuracy > Teacher's Accuracy?). *Suppose we have access to the population, i.e., $n \to \infty$. Further, let Assumptions 4.1 and 4.2 hold with $c = \Theta(1)$ in Assumption 4.2 (recall that $c < 1$). Define $\hat{\lambda} := 2n\lambda$ and $r := \frac{(1-c)}{4\hat{\lambda}}$. Suppose λ is chosen so that $\hat{\lambda} \in \left[\frac{1-c}{2.16}, \frac{1-c}{0.40}\right]$, which corresponds to $r \in [0.10, 0.54]$. If the label corruption fraction*

$$p \in \left(\max\left(\frac{1.08 - r}{2.08}, \frac{1+r}{3.7}\right), 1 - \frac{0.51(1+r)^2}{1+2r}\right),$$

then the student achieves 100% population accuracy (w.r.t. the true labels), while the teacher only achieves a population accuracy of 100(1-p)% (again, w.r.t. the true labels).

Discussion: In our setup, there exists $0 < p_{\text{low}} < p_{\text{high}} < 0.5$ such that (i) when $p \leq p_{\text{low}}$, the teacher attains 100% accuracy and so there is no need for SD, (ii) when $p \in (p_{\text{low}}, p_{\text{high}})$, the student attains 100% accuracy while the teacher attains $100(1-p)\%$ accuracy, and (iii) when $p \geq p_{\text{high}}$, both the teacher and student attain $100(1-p)\%$ accuracy. The range of p in Theorem 4.3 $\subseteq (p_{\text{low}}, p_{\text{high}})$; our range is more conservative than the actual range because we had to impose some more restrictions on p in order to control certain error terms in our analysis. In Figure 2, we plot the teacher's and student's accuracies as a function of p for $r = \{0.2, 0.3, 0.4\}$ obtained by exactly solving for $\boldsymbol{\theta}^*_{\text{T}}$ and $\boldsymbol{\theta}^*_{\text{S}}$ (through a computer). In all the cases, it can be seen that the range of p where the student outperforms the teacher as per Theorem 4.3 falls within the actual range of p where the student outperforms the teacher.

The detailed proof of Theorem 4.3 can be found in Appendix I; we now outline the **key steps in the proof**.

Step 1 (Details in Appendix I.1). In Lemma I.1, we obtain expressions for the teacher's predicted soft labels $\{y_i^{(\text{T})}\}_{i=1}^{2n}$. Specifically, we get:

$$y_i^{(\text{T})} = \begin{cases} \hat{\lambda}\hat{\alpha}\ \forall\ i \in \mathcal{S}_{1,\text{bad}}, (1-\hat{\lambda}\alpha)\ \forall\ i \in \mathcal{S}_{1,\text{good}}, \\ (1-\hat{\lambda}\hat{\alpha})\ \forall\ i \in \mathcal{S}_{0,\text{bad}}, \hat{\lambda}\alpha\ \forall\ i \in \mathcal{S}_{0,\text{good}}, \end{cases} \tag{11}$$

where $\alpha \geq 0$ and $\hat{\alpha} \geq 0$ are obtained by jointly solving:

$$\sigma\Big(cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\hat{\alpha}\Big) = \hat{\lambda}\hat{\alpha}, \text{ and} \tag{12}$$

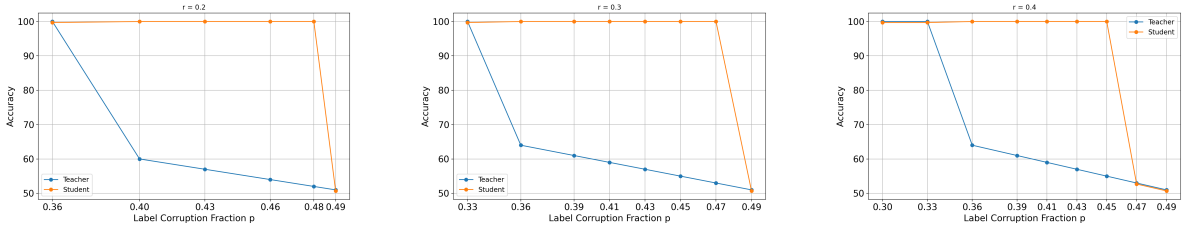
$$\sigma\Big(cn\big(\alpha - (\alpha+\hat{\alpha})p\big) + (1-c)\alpha\Big) = 1 - \hat{\lambda}\alpha. \tag{13}$$

We focus on the interesting case of:
(a) p being large enough so that the teacher misclassifies the incorrectly labeled points ($\mathcal{S}_{1,\text{bad}} \cup \mathcal{S}_{0,\text{bad}}$) because otherwise, there is no need for SD, and
(b) $\hat{\lambda}$ being chosen sensibly so that the teacher at least correctly classifies the correctly labeled points ($\mathcal{S}_{1,\text{good}} \cup \mathcal{S}_{0,\text{good}}$) because otherwise, SD is hopeless.

Later in Step 3, we impose conditions on p (a lower bound) and $\hat{\lambda}$ s.t. **(a)** and **(b)** hold by requiring $\hat{\lambda}\hat{\alpha} < \frac{1}{2}$ and $\hat{\lambda}\alpha < \frac{1}{2}$.

[9] This is taken over the training set.

[10] $\mathbb{1}(.)$ is the indicator function. Specifically, $\mathbb{1}(z) = 1$ if z is true and 0 if z is false.



(a) $r = 0.2$. Derived bound in Thm. 4.3: $p \in (0.423, 0.475)$.

(b) $r = 0.3$. Derived bound in Thm. 4.3: $p \in (0.375, 0.461)$.

(c) $r = 0.4$. Derived bound in Thm. 4.3: $p \in (0.378, 0.444)$.

Figure 2. Comparison of student's and teacher's accuracies for different values of label corruption fraction p obtained by exactly solving Equations (12) and (13) for the teacher and Equations (15) and (16) for the student (using a computer). We set $c = 0.1$ and $n = 5000$ here. In all the cases, note that the range of p where the student outperforms the teacher as per Theorem 4.3 falls within the actual range of p where the student outperforms the teacher.

Step 2 (Details in Appendix I.3). Similar to the teacher in Step 1, in Lemma I.2, we show that the student's predicted soft label for the ith sample, $y_i^{(\text{S})}$, turns out to be:

$$y_i^{(\text{S})} = \begin{cases} \hat{\lambda}\hat{\alpha} + \hat{\lambda}\hat{\beta} \; \forall \, i \in \mathcal{S}_{1,\text{bad}}, \\ (1 - \hat{\lambda}\alpha - \hat{\lambda}\beta) \; \forall \, i \in \mathcal{S}_{1,\text{good}}, \\ (1 - \hat{\lambda}\hat{\alpha} - \hat{\lambda}\hat{\beta}) \; \forall \, i \in \mathcal{S}_{0,\text{bad}}, \\ \hat{\lambda}\alpha + \hat{\lambda}\beta \; \forall \, i \in \mathcal{S}_{0,\text{good}}, \end{cases} \tag{14}$$

where $\beta \geq 0$ and $\hat{\beta} \geq 0$ (assuming $\hat{\lambda}\hat{\alpha} < \frac{1}{2}$ and $\hat{\lambda}\alpha < \frac{1}{2}$) are obtained by jointly solving:

$$\sigma\Big(cn\big(\beta - (\beta + \hat{\beta})p\big) - (1-c)\hat{\beta}\Big) = \hat{\lambda}\hat{\alpha} + \hat{\lambda}\hat{\beta}, \text{ and} \tag{15}$$

$$\sigma\Big(cn\big(\beta - (\beta + \hat{\beta})p\big) + (1-c)\beta\Big) = 1 - \hat{\lambda}\alpha - \hat{\lambda}\beta. \tag{16}$$

Now note that if $\hat{\lambda}\hat{\alpha} + \hat{\lambda}\hat{\beta} > \frac{1}{2}$ and $\hat{\lambda}\alpha + \hat{\lambda}\beta < \frac{1}{2}$, then the student has managed to correctly classify all the points in the training set; we ensure this in Step 3 by upper bounding p. The tradeoff here is that the (1-0) accuracy of the student increases at the cost of decreased confidence in classifying the correctly labeled points compared to the teacher.

Step 3 (Details in Appendix I.5). Now we come to the *challenging* part of the proof. To obtain a range for p, we need to analytically solve Equation (12) and Equation (13) for the teacher and then Equation (15) and Equation (16) for the student, which is particularly *challenging* due to the non-linearity of the sigmoid function present in these equations. Our proof technique involves employing the first-order Maclaurin series expansion of the sigmoid function which enables us to bound $\alpha, \hat{\alpha}, \beta$ and $\hat{\beta}$ as a function of p, $\hat{\lambda}$ and c in a small range (while imposing some conditions on p and $\hat{\lambda}$ to keep the range small). Using this, we can bound the teacher's and student's predictions, and then impose conditions on p and $\hat{\lambda}$ such that the teacher only correctly classifies the correctly labeled points and errs on all the incorrectly labeled points (i.e., $\hat{\lambda}\alpha < \frac{1}{2}$ and $\hat{\lambda}\hat{\alpha} < \frac{1}{2}$; see Step 1) but the student correctly classifies all the points (i.e., $\hat{\lambda}\alpha + \hat{\lambda}\beta < \frac{1}{2}$ and $\hat{\lambda}\hat{\alpha} + \hat{\lambda}\hat{\beta} > \frac{1}{2}$; see Step 2). Finally, since $n \to \infty$, population accuracy $\to$ training accuracy.

In Appendix J, we discuss the logistic regression analogue of SD's variance-reducing effect in linear regression.

5. Empirical Results

We consider multi-class classification with the cross-entropy loss on several vision datasets available in PyTorch's torchvision, namely, CIFAR-100 with 100 classes, Caltech-256 with 257 classes, Food-101 with 101 classes, StanfordCars with 196 classes and Flowers-102 with 102 classes. Since Caltech-256 does not have any default train/test split, we pick 25k random images from the full dataset to form the training set, while the remaining images form the test set. For all datasets, we train a softmax layer on top of a pre-trained ResNet-34/VGG-16 model on ImageNet which is kept fixed, i.e., we do *linear probing* on ResNet-34/VGG-16. No data augmentation is involved. Next, we describe the three types of label corruption that we experiment on.

Corruption Type 1 (Random Corruption): Suppose the set of labels is $[C] := \{1, \ldots, C\}$. Consider a sample whose true label is $c \in [C]$. A corruption level of $100p$ % means we observe this sample's label as c with probability $(1-p)$ or some random $i \in [C] \setminus c$ with probability $\frac{p}{C-1}$ for each such $i \neq c$. We call this **random** corruption.[11]

Corruption Type 2 (Hierarchical Corruption (Hendrycks et al., 2018)): Here, the label corruption only occurs between similar classes. This is a more realistic type of corruption compared to random corruption. By default, CIFAR-100 comes with 20 super-classes each containing 5 semantically similar classes; for e.g., the super-class "fish" consists of aquarium fish, flatfish, ray,

[11] This has been also called symmetric noise in prior work; see for e.g., Chen et al. (2019).

shark and trout. Unfortunately, the other datasets don't have any semantically similar classes provided by default. Now, we describe the exact corruption scheme. Consider a sample whose true class is c and super-class is $S = \{c_1, \ldots, c_{|S|}\}$. A corruption level of $100p$ % means we observe this sample's label as c with probability $(1-p)$ or some random $c' \in S \setminus c$ with probability $\frac{p}{|S|-1}$ (for each such $c' \neq c$).

Corruption Type 3 (Adversarial Corruption): Instead of semantically similar classes, we determine "hard" classes for each class by looking at the output of the teacher in the noiseless (i.e., no-corruption) case and induce label corruption only among these hard classes. Specifically, in the noiseless case, for a sample $\boldsymbol{x}$, let $p_{\mathrm{T}}(\boldsymbol{x}, c)$ be the teacher's predicted probability of $\boldsymbol{x}$ belonging to class $c \in \{1, \ldots, C\}$. Also, let $\mathcal{X}_c$ be the set of samples in the training set belonging to class c. Now, for each class c, we compute $\boldsymbol{\nu}_c = \left[\frac{1}{|\mathcal{X}_c|}\sum_{\boldsymbol{x}\in\mathcal{X}_c} p_{\mathrm{T}}(\boldsymbol{x}, 1), \ldots, \frac{1}{|\mathcal{X}_c|}\sum_{\boldsymbol{x}\in\mathcal{X}_c} p_{\mathrm{T}}(\boldsymbol{x}, C)\right] \in \mathbb{R}^C$, and define the k hardest classes for class c to be the indices in $\{1, \ldots, C\} \setminus c$ corresponding to the k largest values in $\boldsymbol{\nu}_c$. For our experiments, we take $k = 5$. Now, we describe the corruption scheme. Consider a sample whose true class is c and the set of hardest 5 classes for c is S. A corruption level of $100p$ % means we observe this sample's label as c with probability $(1-p)$ or some random $c' \in S$ with probability $p/5$. We call this **adversarial** corruption.

(i) Verifying Remark 3.6: In Remark 3.6, we advocated trying $\xi > 1$ in the high noise regime. We shall now test our recommendation on several noisy datasets. The teacher is trained with the ℓ_2-regularized cross-entropy loss and the student's per-sample loss is given by Equation (1) where ℓ is the ℓ_2-regularized cross-entropy loss. Following our theory setting, the teacher and student are both trained with the same ℓ_2-regularization parameter; the common weight decay value (PyTorch's ℓ_2-reg. parameter) is set to 5×10^{-4}. Note that this weight decay value was the first one that we tried (i.e., it was not cherry-picked); in fact, we show results with other weight decay values in Appendix K.3. We defer the remaining experimental details to Appendix L. In Table 1, we list the student's improvement over the teacher (i.e., student's test accuracy - teacher's test accuracy)[12] averaged across 3 different runs for different values of ξ in the case of 50% random, hierarchical and adversarial corruption while doing linear probing with ResNet-34 and VGG-16. Note that the value of ξ yielding the biggest improvement is > 1. Table 2 shows similar results for 30% corruption with ResNet-34 (using the same weight decay value, viz., 5×10^{-4}); even here, the best performing ξ is > 1. We remind the reader that our recommendation of using $\xi > 1$ is *only for the high noise regime*. In Appendix K.1, we show that $\xi > 1$ is detrimental in the *noiseless case*.

[12]The individual accuracies of the teacher and student can be found in Appendix L; we omit them in the main text for brevity.

Table 1. **50% Corruption:** Average ($\pm$ 1 std.) improvement of student over teacher (i.e., student's test set accuracy - teacher's test set accuracy) with different values of the imitation parameter ξ; recall that $\xi = 0$ corresponds to the teacher. Observe that in all cases, the value of ξ yielding the biggest improvement is more than 1 (although in Food-101 with ResNet-34, $\xi = 1$ does just as well as $\xi > 1$). This is consistent with our message in Remark 3.6, where we advocate trying $\xi > 1$ in the high noise regime.

ξ	Improvement of student over teacher
0.2	2.22 ± 0.12 %
0.5	5.18 ± 0.03 %
0.7	6.84 ± 0.06 %
1.0	8.54 ± 0.29 %
1.2	9.66 ± 0.23 %
1.5	$\mathbf{10.04 \pm 0.51}$ **%**
1.7	$\mathbf{9.81 \pm 0.55}$ **%**
2.0	8.56 ± 0.73 %

(a) 50% Random Corruption in Caltech-256 w/ ResNet-34

ξ	Improvement of student over teacher
0.5	0.89 ± 0.10 %
1.0	2.01 ± 0.14 %
1.5	3.13 ± 0.11 %
2.0	4.22 ± 0.20 %
2.5	5.28 ± 0.13 %
3.0	$\mathbf{5.78 \pm 0.12}$ **%**
3.5	$\mathbf{5.86 \pm 0.18}$ **%**
4.0	5.32 ± 0.33 %

(b) 50% Random Corruption in Caltech-256 w/ VGG-16

ξ	Improvement of student over teacher
0.2	0.98 ± 0.12 %
0.5	2.46 ± 0.11 %
0.7	3.38 ± 0.02 %
1.0	4.19 ± 0.09 %
1.2	$\mathbf{4.46 \pm 0.19}$ **%**
1.5	$\mathbf{4.46 \pm 0.17}$ **%**
1.7	$\mathbf{4.32 \pm 0.18}$ **%**
2.0	3.52 ± 0.23 %

(c) 50% Hierarchical Corruption in CIFAR-100 w/ ResNet-34

ξ	Improvement of student over teacher
0.2	1.10 ± 0.09 %
0.5	2.69 ± 0.02 %
0.7	3.72 ± 0.05 %
1.0	5.29 ± 0.11 %
1.2	6.26 ± 0.09 %
1.5	$\mathbf{7.20 \pm 0.14}$ **%**
1.7	$\mathbf{7.23 \pm 0.17}$ **%**
2.0	6.42 ± 0.26 %

(d) 50% Hierarchical Corruption in CIFAR-100 w/ VGG-16

ξ	Improvement of student over teacher
0.2	0.13 ± 0.08 %
0.5	0.97 ± 0.04 %
0.7	1.45 ± 0.01 %
1.0	$\mathbf{1.85 \pm 0.09}$ **%**
1.2	$\mathbf{1.87 \pm 0.06}$ **%**
1.5	$\mathbf{1.86 \pm 0.08}$ **%**
1.7	$\mathbf{1.80 \pm 0.05}$ **%**
2.0	1.53 ± 0.02 %

(e) 50% Adversarial Corruption in Food-101 w/ ResNet-34

ξ	Improvement of student over teacher
0.2	0.79 ± 0.23 %
0.5	2.14 ± 0.09 %
0.7	2.96 ± 0.04 %
1.0	3.85 ± 0.05 %
1.2	$\mathbf{4.22 \pm 0.15}$ **%**
1.5	$\mathbf{4.39 \pm 0.29}$ **%**
1.7	$\mathbf{4.20 \pm 0.34}$ **%**
2.0	3.53 ± 0.49 %

(f) 50% Adversarial Corruption in Food-101 w/ VGG-16

(ii) Verifying Remark 3.7: In Remark 3.7, we claimed that the utility of the teacher's predictions increases with the amount of label noise. To demonstrate this, we train the student with $\xi = 1$ which corresponds to setting the teacher's

Table 2. **30% Corruption:** Average ($\pm$ 1 std.) improvement of student over teacher (i.e., student's test set accuracy - teacher's test set accuracy) with different values of the imitation parameter ξ (recall that $\xi = 0$ corresponds to the teacher). Just like in Table 1, note that the value of ξ yielding the biggest improvement is more than 1. This further validates Remark 3.6.

ξ	Improvement of student over teacher
0.2	0.89 ± 0.15 %
0.5	2.15 ± 0.06 %
0.7	2.75 ± 0.10 %
1.0	3.32 ± 0.11 %
1.2	$\mathbf{3.53 \pm 0.16}$ %
1.5	$\mathbf{3.46 \pm 0.12}$ %
1.7	2.96 ± 0.24 %
2.0	1.79 ± 0.29 %

(a) 30% Random Corruption in Stanford Cars w/ ResNet-34

ξ	Improvement of student over teacher
0.5	-0.12 ± 0.20 %
1.0	0.54 ± 0.02 %
1.5	0.86 ± 0.01 %
2.0	1.57 ± 0.34 %
2.5	2.05 ± 0.27 %
3.0	2.49 ± 0.25 %
3.5	2.62 ± 0.12 %
4.0	2.87 ± 0.09 %
4.5	$\mathbf{3.01 \pm 0.22}$ %
5.0	$\mathbf{3.21 \pm 0.07}$ %
5.5	$\mathbf{2.94 \pm 0.33}$ %
6.0	$\mathbf{3.06 \pm 0.09}$ %

(b) 30% Adversarial Corruption in Flowers-102 w/ ResNet-34

predicted *soft* labels as the student's targets (just as we did in Section 4) and completely ignoring the provided labels. All other experimental details (including weight decay) are the same as (i) before. In Table 3, we show the student's improvement over the teacher averaged across 3 different runs for varying degrees and types of label corruption with ResNet-34; see the table caption for discussion.

In Appendix M, we show some empirical results for full network training from scratch.

6. Limitations

There are some limitations of our work which pave the way for interesting directions of future work. Our results in Section 4 are under Assumption 4.2; it would be nice to derive similar results under a weaker assumption such as in expectation (see the discussion after Assumption 4.2) or by assuming that the feature inner products are bounded in some range. Also, our results in Section 4 are with $\xi = 1$; one could try to obtain results with a general ξ to shed some light on how to better tune ξ, like we did for linear regression. Moreover, all our results are for convex problems; the analysis would be much more challenging for non-convex problems such as full network training where the results may depend on which optimum the teacher model converges to. We also hope that future work can do more experimentation with full network training.

Table 3. **ResNet-34 with $\xi = 1$:** Average ($\pm$ 1 std.) improvement of student over teacher (i.e., student's test set accuracy - teacher's test set accuracy) with different kinds and varying levels of label corruption. Observe that *as the corruption level increases, so does the improvement of the student over the teacher* for all types of corruption. This shows that the utility of the teacher's predictions (which is the core idea of SD) increases with the amount of label noise corroborating our claim in Remark 3.7.

Corruption (corr.) level	**Random** corr.: Improvement of student	**Adversarial** corr.: Improvement of student
0%	-0.04 ± 0.02 %	-0.04 ± 0.02 %
10%	2.51 ± 0.11 %	2.32 ± 0.10 %
30%	6.14 ± 0.16 %	5.08 ± 0.25 %
50%	8.54 ± 0.29 %	5.77 ± 0.19 %

(a) Caltech-256 (Random and Adversarial Corruption)

Corruption (corr.) level	**Random** corr.: Improvement of student	**Hierarchical** corr.: Improvement of student
0%	-0.23 ± 0.06 %	-0.23 ± 0.06 %
10%	0.63 ± 0.11 %	1.19 ± 0.08 %
30%	1.34 ± 0.13 %	2.80 ± 0.06 %
50%	2.11 ± 0.15 %	4.19 ± 0.09 %

(b) CIFAR-100 (Random and Hierarchical Corruption)

Corruption (corr.) level	**Random** corr.: Improvement of student	**Adversarial** corr.: Improvement of student
0%	-0.37 ± 0.10 %	-0.37 ± 0.10 %
10%	0.10 ± 0.04 %	0.25 ± 0.05 %
30%	0.47 ± 0.04 %	0.77 ± 0.06 %
50%	1.12 ± 0.08 %	1.85 ± 0.09 %

(c) Food-101 (Random and Adversarial Corruption)

7. Conclusion

In this work, we analyzed the utility of self-distillation (SD) in supervised learning with noisy labels. Our main algorithmic insight was introducing the idea of assigning negative weight to the provided labels in the SD objective, i.e., setting $\xi > 1$ in the high label noise regime. On the theoretical side, we characterized when optimal SD is strictly better than optimal regularization in linear regression. Further, for a binary classification problem with random label corruption, we quantified the range of label corruption in which the student outperforms the teacher under certain assumptions on the data.

Acknowledgements

This work was supported by NSF TRIPODS grant 1934932. The authors are also grateful to anonymous reviewers for their feedback which helped in improving this manuscript.

References

Ahn, S., Hu, S. X., Damianou, A., Lawrence, N. D., and Dai, Z. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9163–9171, 2019.

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Beyer, L., Zhai, X., Royer, A., Markeeva, L., Anil, R., and Kolesnikov, A. Knowledge distillation: A good teacher is patient and consistent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10925–10934, 2022.

Chen, P., Liao, B. B., Chen, G., and Zhang, S. Understanding and utilizing deep neural networks trained with noisy labels. In *International Conference on Machine Learning*, pp. 1062–1070. PMLR, 2019.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. E. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.

Cheng, X., Rao, Z., Chen, Y., and Zhang, Q. Explaining knowledge distillation by quantifying the knowledge. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12925–12935, 2020.

Dong, B., Hou, J., Lu, Y., and Zhang, Z. Distillation $\approx$ early stopping? harvesting dark knowledge utilizing anisotropic information retrieval for overparameterized neural network. *arXiv preprint arXiv:1910.01255*, 2019.

Furlanello, T., Lipton, Z., Tschannen, M., Itti, L., and Anandkumar, A. Born again neural networks. In *International Conference on Machine Learning*, pp. 1607–1616. PMLR, 2018.

Gou, J., Yu, B., Maybank, S. J., and Tao, D. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

Hendrycks, D., Mazeika, M., Wilson, D., and Gimpel, K. Using trusted data to train deep networks on labels corrupted by severe noise. *Advances in neural information processing systems*, 31, 2018.

Hinton, G., Vinyals, O., Dean, J., et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

Ji, G. and Zhu, Z. Knowledge distillation in wide neural networks: Risk bound, data efficiency and imperfect teacher. *Advances in Neural Information Processing Systems*, 33: 20823–20833, 2020.

Kakade, S. M., Sridharan, K., and Tewari, A. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. *Advances in neural information processing systems*, 21, 2008.

Kaplun, G., Malach, E., Nakkiran, P., and Shalev-Shwartz, S. Knowledge distillation: Bad models can be good role models. *arXiv preprint arXiv:2203.14649*, 2022.

Kumar, A., Raghunathan, A., Jones, R., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., and Pennington, J. Wide neural networks of any depth evolve as linear models under gradient descent. *Advances in neural information processing systems*, 32, 2019.

Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., and Hoi, S. C. H. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34: 9694–9705, 2021.

Li, Y., Yang, J., Song, Y., Cao, L., Luo, J., and Li, L.-J. Learning from noisy labels with distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1910–1918, 2017.

Lopez-Paz, D., Bottou, L., Schölkopf, B., and Vapnik, V. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2015.

Lukasik, M., Bhojanapalli, S., Menon, A., and Kumar, S. Does label smoothing mitigate label noise? In *International Conference on Machine Learning*, pp. 6448–6458. PMLR, 2020.

Menon, A. K., Rawat, A. S., Reddi, S., Kim, S., and Kumar, S. A statistical perspective on distillation. In *International Conference on Machine Learning*, pp. 7632–7642. PMLR, 2021.

Mobahi, H., Farajtabar, M., and Bartlett, P. Self-distillation amplifies regularization in hilbert space. *Advances in Neural Information Processing Systems*, 33:3351–3361, 2020.

Pham, M., Cho, M., Joshi, A., and Hegde, C. Revisiting self-distillation. *arXiv preprint arXiv:2206.08491*, 2022.

Phuong, M. and Lampert, C. Towards understanding knowledge distillation. In *International Conference on Machine Learning*, pp. 5142–5151. PMLR, 2019.

Sarfraz, F., Arani, E., and Zonooz, B. Knowledge distillation beyond model compression. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 6136–6143. IEEE, 2021.

Stanton, S., Izmailov, P., Kirichenko, P., Alemi, A. A., and Wilson, A. G. Does knowledge distillation really work? *Advances in Neural Information Processing Systems*, 34: 6906–6919, 2021.

Sun, S., Cheng, Y., Gan, Z., and Liu, J. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*, 2019.

Wei, J., Liu, H., Liu, T., Niu, G., Sugiyama, M., and Liu, Y. To smooth or not? when label smoothing meets noisy labels. In *International Conference on Machine Learning*, pp. 23589–23614. PMLR, 2022.

Xie, Q., Luong, M.-T., Hovy, E., and Le, Q. V. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10687–10698, 2020.

Yang, S., Sanghavi, S., Rahmanian, H., Bakus, J., and SVN, V. Toward understanding privileged features distillation in learning-to-rank. *Advances in Neural Information Processing Systems*, 35:26658–26670, 2022.

Yuan, L., Tay, F. E., Li, G., Wang, T., and Feng, J. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3903–3911, 2020.

Zhou, H., Song, L., Chen, J., Zhou, Y., Wang, G., Yuan, J., and Zhang, Q. Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective. *arXiv preprint arXiv:2102.00650*, 2021.

Appendix

Contents

- Appendix A: Relevant Prior Work on Label Smoothing
- Appendix B: Proof of Equation (4)
- Appendix C: Proof of Theorem 3.2
- Appendix D: Behavior of ξ^* w.r.t. γ^2
- Appendix E: Detailed Version and Proof of Theorem 3.8
- Appendix F: Detailed Version and Proof of Theorem 3.9
- Appendix G: Proof of Theorem 3.10
- Appendix H: Empirical Motivation for Assumption 4.2
- Appendix I: Proof of Theorem 4.3
- Appendix J: Variability of Predictions of Teacher and Student
- Appendix K: More Empirical Results
- Appendix L: Detailed Empirical Results
- Appendix M: Preliminary Empirical Results for Full Network Training

A. Relevant Prior Work on Label Smoothing

As Yuan et al. (2020) explain, SD can be viewed as an advanced version of label smoothing (LS) where the smoothing distribution is the teacher's predicted label distribution instead of the uniform distribution. One thing to note is that under no circumstance can we *only* use the uniform distribution to train a model; otherwise it will learn nothing. However, as we *theoretically* show in Section 4, *only* using the teacher's predictions to train the student (i.e., setting $\xi = 1$) can also lead to improvement compared to the teacher; we also back this up via empirical results with $\xi = 1$ in Section 5.

Just as we show that SD has a variance-reduction effect by enhancing regularization, Lukasik et al. (2020) show that LS has a regularization/shrinkage effect. However, unlike us, Lukasik et al. (2020) do not compute the optimal combining coefficient α (analogue of ξ in SD) and have no *theoretical* insights like Remark 3.6 on the choice of α. Further, Lukasik et al. (2020) do not have any *theoretical* results in the classification setting.[13]

In the high label noise regime, Wei et al. (2022) advocate assigning *negative weight to the uniform distribution* and *weight > 1 to the hard label* in LS; they call this *negative* label smoothing (*NLS*). This is actually contrary to our result for SD; we show that in the high noise regime, one should assign *weight > 1 to the teacher's predicted label* and *negative weight to the hard label* in SD. Wei et al. (2022) claim that NLS is helpful as it improves model confidence; in contrast, our analysis for the classification problem reveals that the student's accuracy increases at the cost of decreased confidence in classifying the correctly labeled points compared to the teacher (see the last sentence in Step 2 of the proof sketch of Theorem 4.3). Our explanation for why SD is helpful in learning with noisy labels is that SD has a noise variance-reducing effect. So even though SD and LS are related, the ways in which they operate in the high label noise regime are different.

B. Proof of Equation (4)

Recall that the student's objective is:

$$f_S(\boldsymbol{\theta};\xi) = \xi\ell(\hat{\boldsymbol{Y}},\boldsymbol{\theta}) + (1-\xi)\ell(\boldsymbol{Y},\boldsymbol{\theta}), \tag{17}$$

where $\ell(\boldsymbol{Z},\boldsymbol{\theta}) := \frac{1}{2}\|\boldsymbol{Z} - \boldsymbol{X}^T\boldsymbol{\theta}\|^2 + \frac{\lambda}{2}\|\boldsymbol{\theta}\|^2$. Expanding Equation (17), we get:

$$f_S(\boldsymbol{\theta};\xi) = \xi\Big(\frac{1}{2}\|\hat{\boldsymbol{Y}} - \boldsymbol{X}^T\boldsymbol{\theta}\|^2\Big) + (1-\xi)\Big(\frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{X}^T\boldsymbol{\theta}\|^2\Big) + \frac{\lambda}{2}\|\boldsymbol{\theta}\|^2. \tag{18}$$

Now:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_S(\xi) := \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^d} f_S(\boldsymbol{\theta};\xi) &= (\boldsymbol{X}\boldsymbol{X}^T + \lambda\boldsymbol{I}_d)^{-1}\boldsymbol{X}(\xi\hat{\boldsymbol{Y}} + (1-\xi)\boldsymbol{Y}) \\ &= \xi(\boldsymbol{X}\boldsymbol{X}^T + \lambda\boldsymbol{I}_d)^{-1}\boldsymbol{X}\boldsymbol{X}^T\hat{\boldsymbol{\theta}}_T + (1-\xi)\hat{\boldsymbol{\theta}}_T,\end{aligned} \tag{19}$$

where Equation (19) is obtained by using $\hat{\boldsymbol{Y}} = \boldsymbol{X}^T\hat{\boldsymbol{\theta}}_T$ and Equation (2). Rewriting Equation (19) slightly gives us the desired result.

C. Proof of Theorem 3.2

Plugging in $\hat{\boldsymbol{\theta}}_T$ from Equation (3) in Equation (4), we get:

$$\hat{\boldsymbol{\theta}}_S(\xi) = \Big(\xi(\boldsymbol{X}\boldsymbol{X}^T + \lambda\boldsymbol{I}_d)^{-1}\boldsymbol{X}\boldsymbol{X}^T + (1-\xi)\boldsymbol{I}_d\Big)(\boldsymbol{X}\boldsymbol{X}^T + \lambda\boldsymbol{I}_d)^{-1}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{\theta}^* + \boldsymbol{\eta}). \tag{20}$$

With the SVD notation of $\boldsymbol{X}$, we can rewrite $\hat{\boldsymbol{\theta}}_S(\xi)$ from Equation (20) as:

$$\hat{\boldsymbol{\theta}}_S(\xi) = \sum_{j=1}^{r}\frac{\langle\boldsymbol{\theta}^*,\boldsymbol{u}_j\rangle}{\big(1+\lambda/\sigma_j^2\big)}\left(1-\xi\Big(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2}\Big)\right)\boldsymbol{u}_j + \sum_{j=1}^{r}\frac{\langle\boldsymbol{\eta},\boldsymbol{v}_j\rangle/\sigma_j}{\big(1+\lambda/\sigma_j^2\big)}\left(1-\xi\Big(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2}\Big)\right)\boldsymbol{u}_j. \tag{21}$$

Also, since $\{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_d\}$ forms an orthonormal basis for $\mathbb{R}^d$, we have:

$$\boldsymbol{\theta}^* = \sum_{j=1}^{d}\langle\boldsymbol{\theta}^*,\boldsymbol{u}_j\rangle\boldsymbol{u}_j.$$

[13] In fact, our analysis can be extended to *theoretically* investigate their *empirically justified* claim of applying LS on the teacher prior to SD for improving the student's performance; this would be interesting future work.

So, using Equation (21):

$$\boldsymbol{\epsilon}_S(\xi) = -\sum_{j=1}^{r} \langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right) \left(1 + \frac{\xi}{1+\lambda/\sigma_j^2} \right) \boldsymbol{u}_j - \sum_{j=r+1}^{d} \langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle \boldsymbol{u}_j + \sum_{j=1}^{r} \frac{\langle \boldsymbol{\eta}, \boldsymbol{v}_j \rangle / \sigma_j}{\left(1+\lambda/\sigma_j^2\right)^2} \left(1 - \xi \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right) \right) \boldsymbol{u}_j. \quad (22)$$

Using Assumption 3.1, we have:

$$\mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\epsilon}_S(\xi)] = -\sum_{j=1}^{r} \langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right) \left(1 + \frac{\xi}{1+\lambda/\sigma_j^2} \right) \boldsymbol{u}_j - \sum_{j=r+1}^{d} \langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle \boldsymbol{u}_j. \quad (23)$$

Thus, using the orthonormality of $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d\}$, we get:

$$\left\| \mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\epsilon}_S(\xi)] \right\|^2 = \sum_{j=1}^{r} \left(\langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle \right)^2 \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right)^2 \left(1 + \frac{\xi}{1+\lambda/\sigma_j^2} \right)^2 + \sum_{j=r+1}^{d} \left(\langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle \right)^2. \quad (24)$$

Next:

$$\mathbb{E}_{\boldsymbol{\eta}} \left[\left\| \boldsymbol{\epsilon}_S(\xi) - \mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\epsilon}_S(\xi)] \right\|^2 \right] = \mathbb{E}_{\boldsymbol{\eta}} \left[\left\| \sum_{j=1}^{r} \frac{\langle \boldsymbol{\eta}, \boldsymbol{v}_j \rangle / \sigma_j}{\left(1+\lambda/\sigma_j^2\right)^2} \left(1 - \xi \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right) \right) \boldsymbol{u}_j \right\|^2 \right] \quad (25)$$

$$= \sum_{j=1}^{r} \frac{\mathbb{E}_{\boldsymbol{\eta}}\left[\left(\langle \boldsymbol{\eta}, \boldsymbol{v}_j \rangle \right)^2 \right]}{\sigma_j^2 \left(1+\lambda/\sigma_j^2\right)^2} \left(1 - \xi \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right) \right)^2 \quad (26)$$

$$= \sum_{j=1}^{r} \frac{\boldsymbol{v}_j^T \mathbb{E}_{\boldsymbol{\eta}}\left[\boldsymbol{\eta}\boldsymbol{\eta}^T \right] \boldsymbol{v}_j}{\sigma_j^2 \left(1+\lambda/\sigma_j^2\right)^2} \left(1 - \xi \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right) \right)^2 \quad (27)$$

$$= \gamma^2 \left\{ \sum_{j=1}^{r} \frac{1}{\sigma_j^2 \left(1+\lambda/\sigma_j^2\right)^2} \left(1 - \xi \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right) \right)^2 \right\}. \quad (28)$$

Equation (26) follows from the orthonormality of the $\boldsymbol{u}_j$'s, Equation (27) follows because the $\boldsymbol{v}_j$'s are independent of $\boldsymbol{\eta}$ from Assumption 3.1, and Equation (28) follows because $\mathbb{E}_{\boldsymbol{\eta}}\left[\boldsymbol{\eta}\boldsymbol{\eta}^T\right] = \gamma^2 \mathrm{I}_n$ from Assumption 3.1 and because $\boldsymbol{v}_j^T \boldsymbol{v}_j = 1$ for all $j \in \{1, \ldots, r\}$. Rewriting Equation (28) slightly differently, we get:

$$\mathbb{E}_{\boldsymbol{\eta}} \left[\left\| \boldsymbol{\epsilon}_S(\xi) - \mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\epsilon}_S(\xi)] \right\|^2 \right] = \frac{\gamma^2}{\lambda} \left\{ \sum_{j=1}^{r} \frac{\lambda/\sigma_j^2}{\left(1+\lambda/\sigma_j^2\right)^2} \left(1 - \xi \left(\frac{\lambda/\sigma_j^2}{1+\lambda/\sigma_j^2} \right) \right)^2 \right\}. \quad (29)$$

D. Behavior of ξ^* w.r.t. γ^2

Proposition D.1. *ξ^* (in Corollary 3.5) is an increasing function of γ^2.*

Proof. Let $\rho = \gamma^2$. Then from Corollary 3.5:

$$\xi^* = \frac{\sum_{j=1}^{r} \left(\frac{\rho}{\lambda} - \theta_j^* \right) \frac{c_j^2}{(1+c_j)^3}}{\sum_{j=1}^{r} \left(\frac{\rho}{\lambda} c_j + \theta_j^* \right) \frac{c_j^2}{(1+c_j)^4}}. \quad (30)$$

Now,

$$\frac{\partial \xi^*}{\partial \rho} = \frac{\left(\sum_{j=1}^{r} \frac{c_j^2}{(1+c_j)^3} \right) \left(\sum_{j=1}^{r} \frac{\theta_j^* c_j^2}{(1+c_j)^4} \right) + \left(\sum_{j=1}^{r} \frac{c_j^3}{(1+c_j)^4} \right) \left(\sum_{j=1}^{r} \frac{\theta_j^* c_j^2}{(1+c_j)^3} \right)}{\lambda \left(\sum_{j=1}^{r} \left(\frac{\rho}{\lambda} c_j + \theta_j^* \right) \frac{c_j^2}{(1+c_j)^4} \right)^2} > 0. \quad (31)$$

Thus, ξ^* is an increasing function of ρ, i.e., γ^2. ■

E. Detailed Version and Proof of Theorem 3.8

Theorem E.1 (**Detailed Version of Theorem 3.8**). *The following hold with $\theta_j^* := \left(\langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle\right)^2$ (and with $'$ denoting the derivative w.r.t. λ):*

$$e_{\text{sd}}(\lambda) = e_{\text{reg}}(\lambda) - \frac{\left(e'_{reg}(\lambda)\right)^2}{h(\lambda)} \text{ and } e'_{\text{sd}}(\lambda) = e'_{reg}(\lambda)\Big(1 - \frac{2e''_{reg}(\lambda)}{h(\lambda)} + \frac{e'_{reg}(\lambda)h'(\lambda)}{(h(\lambda))^2}\Big),$$
$$\text{where } e_{\text{reg}}(\lambda) = \sum_{j=1}^{r} \frac{\lambda^2\theta_j^*}{(\lambda+\sigma_j^2)^2} + \sum_{j=r+1}^{d} \theta_j^* + \sum_{j=1}^{r} \frac{\gamma^2\sigma_j^2}{(\lambda+\sigma_j^2)^2} \text{ and } h(\lambda) = 4\sum_{j=1}^{r}\Big(\frac{\gamma^2}{\sigma_j^2} + \theta_j^*\Big)\frac{\sigma_j^4}{(\lambda+\sigma_j^2)^4}. \quad (32)$$

*Let $\lambda^*_{reg} := \arg\min_\lambda e_{\text{reg}}(\lambda)$. Then, $e_{\text{sd}}(\lambda^*_{reg}) = e_{\text{reg}}(\lambda^*_{reg})$ and $e'_{\text{sd}}(\lambda^*_{reg}) = 0$, i.e., $\lambda = \lambda^*_{reg}$ is a stationary point of $e_{\text{sd}}(\lambda)$ also. It is a **local maximum** point of $e_{\text{sd}}(\lambda)$ when:*

$$\sum_{k=1}^{r}\sum_{j=1}^{k-1} \frac{\sigma_j^2\sigma_k^2(\sigma_j^2-\sigma_k^2)(\theta_k^*-\theta_j^*)}{(\lambda^*_{reg}+\sigma_j^2)^4(\lambda^*_{reg}+\sigma_k^2)^4} < 0. \quad (33)$$

When the above holds[14], optimal self-distillation is better than optimal ℓ_2-regularization.

Note that if $\lambda = \lambda^*_{\text{reg}}$ is not a local maximum point of $e_{\text{sd}}(\lambda)$, it could be a sub-optimal *local* minimum point or the *global* minimum point of $e_{\text{sd}}(\lambda)$. The other stationary points of $e_{\text{sd}}(\lambda)$ are obtained by solving (this follows from Equation (32)):

$$1 - \frac{2e''_{\text{reg}}(\lambda)}{h(\lambda)} + \frac{e'_{\text{reg}}(\lambda)h'(\lambda)}{(h(\lambda))^2} = 0. \quad (34)$$

Unfortunately, it seems difficult to determine whether a root of Equation (34) or λ^*_{reg} will be the global minimum point of $e_{\text{sd}}(\lambda)$. If λ^*_{reg} is the global minimum point of $e_{\text{sd}}(\lambda)$, then optimal SD is **not** better than (i.e., does not yield any improvement over) optimal ℓ_2-regularization as $e_{\text{sd}}(\lambda^*_{\text{reg}}) = e_{\text{reg}}(\lambda^*_{\text{reg}})$.

Proof. Using Equation (6) and Equation (7) in Equation (5) while using our notation of $\theta_j^* = \left(\langle \boldsymbol{\theta}^*, \boldsymbol{u}_j \rangle\right)^2$ and $c_j = \lambda/\sigma_j^2$ (from Corollary 3.5), we get:

$$e(\lambda,\xi) = \sum_{j=1}^{r}\theta_j^*\left(\frac{c_j}{1+c_j}\right)^2\left(1+\frac{\xi}{1+c_j}\right)^2 + \sum_{j=r+1}^{d}\theta_j^* + \frac{\gamma^2}{\lambda}\left\{\sum_{j=1}^{r}\frac{c_j}{(1+c_j)^2}\left(1-\xi\left(\frac{c_j}{1+c_j}\right)\right)^2\right\}. \quad (35)$$

Thus,

$$e_{\text{reg}}(\lambda) := e(\lambda,0) = \sum_{j=1}^{r}\theta_j^*\left(\frac{c_j}{1+c_j}\right)^2 + \sum_{j=r+1}^{d}\theta_j^* + \frac{\gamma^2}{\lambda}\sum_{j=1}^{r}\frac{c_j}{(1+c_j)^2}. \quad (36)$$

Next, we compute $e_{\text{sd}}(\lambda) := e(\lambda,\xi^*)$.

Lemma E.2.

$$e_{\text{sd}}(\lambda) = e_{\text{reg}}(\lambda) - \frac{\left(\sum_{j=1}^{r}\left(\theta_j^* - \frac{\gamma^2}{\lambda}\right)\frac{c_j^2}{(1+c_j)^3}\right)^2}{\sum_{j=1}^{r}\left(\frac{\gamma^2}{\lambda}c_j + \theta_j^*\right)\frac{c_j^2}{(1+c_j)^4}}. \quad (37)$$

Lemma E.2 involves a little bit of algebra; we prove it in Appendix E.1.

Since the c_j's depend on λ, let us substitute c_j in Equation (36) and Equation (37) and rewrite them.

$$e_{\text{reg}}(\lambda) = \sum_{j=1}^{r}\frac{\lambda^2\theta_j^*}{(\lambda+\sigma_j^2)^2} + \sum_{j=r+1}^{d}\theta_j^* + \sum_{j=1}^{r}\frac{\gamma^2\sigma_j^2}{(\lambda+\sigma_j^2)^2}. \quad (38)$$

[14] Also, assume that $\lambda^*_{\text{reg}} \geq 0$ as the ℓ_2-regularization parameter is supposed to be non-negative.

$$e_{\text{sd}}(\lambda) = e_{\text{reg}}(\lambda) - \left(\left(\underbrace{\sum_{j=1}^{r} \left(\lambda\theta_j^* - \gamma^2\right) \frac{\sigma_j^2}{(\lambda+\sigma_j^2)^3}}_{:=g(\lambda)} \right)^2 \right) \Bigg/ \left(\sum_{j=1}^{r} \left(\frac{\gamma^2}{\sigma_j^2} + \theta_j^*\right) \frac{\sigma_j^4}{(\lambda+\sigma_j^2)^4} \right). \tag{39}$$

Interestingly, it can be checked that $g(\lambda) = \frac{1}{2} e'_{\text{reg}}(\lambda)$; here $'$ indicates the derivative w.r.t. λ. Plugging this in Equation (39), we get:

$$e_{\text{sd}}(\lambda) = e_{\text{reg}}(\lambda) - \frac{\left(e'_{\text{reg}}(\lambda)\right)^2}{h(\lambda)}, \text{ where } h(\lambda) = 4\sum_{j=1}^{r} \left(\frac{\gamma^2}{\sigma_j^2} + \theta_j^*\right) \frac{\sigma_j^4}{(\lambda+\sigma_j^2)^4}. \tag{40}$$

Now note that:

$$e'_{\text{sd}}(\lambda) = e'_{\text{reg}}(\lambda)\Big(1 - \frac{2e''_{\text{reg}}(\lambda)}{h(\lambda)} + \frac{e'_{\text{reg}}(\lambda)h'(\lambda)}{(h(\lambda))^2}\Big). \tag{41}$$

Thus, $e'_{\text{reg}}(\lambda) = 0 \implies e'_{\text{sd}}(\lambda) = 0$, i.e., any stationary point of $e_{\text{reg}}(\lambda)$ is also a stationary point of $e_{\text{sd}}(\lambda)$.

Next, $\lambda^*_{\text{reg}} := \arg\min_\lambda e_{\text{reg}}(\lambda)$ satisfies:

$$e'_{\text{reg}}(\lambda^*_{\text{reg}}) = 2\sum_{j=1}^{r} \left(\lambda^*_{\text{reg}}\theta_j^* - \gamma^2\right) \frac{\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^3} = 0. \tag{42}$$

From Equation (41), $e'_{\text{sd}}(\lambda^*_{\text{reg}}) = 0$, i.e., $\lambda = \lambda^*_{\text{reg}}$ is a stationary point of $e_{\text{sd}}(\lambda)$ also. We shall now show that $\lambda = \lambda^*_{\text{reg}}$ can be a *local maximum* point of $e_{\text{sd}}(\lambda)$ in many cases. For that, we need to check the sign of $e''_{\text{sd}}(\lambda^*_{\text{reg}})$. Note that:

$$e''_{\text{sd}}(\lambda^*_{\text{reg}}) = e''_{\text{reg}}(\lambda^*_{\text{reg}}) \left(1 - \frac{2e''_{\text{reg}}(\lambda^*_{\text{reg}})}{h(\lambda^*_{\text{reg}})}\right). \tag{43}$$

The above follows by just differentiating Equation (41) and evaluating it at $\lambda = \lambda^*_{\text{reg}}$ while using the fact that $e'_{\text{reg}}(\lambda^*_{\text{reg}}) = 0$. Also note that $e''_{\text{reg}}(\lambda^*_{\text{reg}}) > 0$ as $\lambda = \lambda^*_{\text{reg}}$ is a minimizer of $e_{\text{reg}}(\lambda)$. Let us now examine the sign of $t = \left(1 - \frac{2e''_{\text{reg}}(\lambda^*_{\text{reg}})}{h(\lambda^*_{\text{reg}})}\right)$. After a bit of algebra:

$$t = 1 - \frac{\sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^4} \left(\theta_j^*\sigma_j^2 + 3\gamma^2 - 2\lambda^*_{\text{reg}}\theta_j^*\right)}{\sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^4} \left(\gamma^2 + \theta_j^*\sigma_j^2\right)} = \frac{2\sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^4} \left(\lambda^*_{\text{reg}}\theta_j^* - \gamma^2\right)}{\sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^4} \left(\gamma^2 + \theta_j^*\sigma_j^2\right)} \tag{44}$$

The denominator of t is positive so we only need to analyze the sign of the numerator, $\sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^4} \left(\lambda^*_{\text{reg}}\theta_j^* - \gamma^2\right)$; let us refer to it as t_2 for brevity. From Equation (42), we have that:

$$\lambda^*_{\text{reg}} = \frac{\gamma^2 \sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^3}}{\sum_{j=1}^{r} \frac{\theta_j^*\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^3}}. \tag{45}$$

Using this, we get:

$$t_2 = \underbrace{\left(\frac{\gamma^2}{\sum_{j=1}^{r} \frac{\theta_j^*\sigma_j^2}{(\lambda^*_{\text{reg}}+\sigma_j^2)^3}}\right)}_{>0} \underbrace{\left(\sum_{j,k} \frac{\sigma_j^2\sigma_k^2(\theta_j^* - \theta_k^*)}{(\lambda^*_{\text{reg}}+\sigma_j^2)^4(\lambda^*_{\text{reg}}+\sigma_k^2)^3}\right)}_{:=t_3} \tag{46}$$

Simplifying t_3 a bit, we get:

$$t_3 = \sum_{k=1}^{r}\sum_{j=1}^{k-1} \frac{\sigma_j^2\sigma_k^2(\sigma_j^2 - \sigma_k^2)(\theta_k^* - \theta_j^*)}{(\lambda^*_{\text{reg}}+\sigma_j^2)^4(\lambda^*_{\text{reg}}+\sigma_k^2)^4}. \tag{47}$$

So, $t_3 < 0 \implies t_2 < 0 \implies t < 0 \implies e''_{\text{sd}}(\lambda^*_{\text{reg}}) < 0$; but this means $\lambda = \lambda^*_{\text{reg}}$ is a local maximum point of $e_{\text{sd}}(\lambda)$. ■

E.1. Proof of Lemma E.2

Proof. Note that $e(\lambda, \xi)$ is a quadratic function of ξ; specifically, it is of the form $a\xi^2 + b\xi + c$, where:

$$a = \sum_{j=1}^{r} \left(\frac{\gamma^2}{\lambda} c_j + \theta_j^*\right) \frac{c_j^2}{(1+c_j)^4}, b = 2\sum_{j=1}^{r} \left(\theta_j^* - \frac{\gamma^2}{\lambda}\right) \frac{c_j^2}{(1+c_j)^3}, \text{ and } c = e_{\text{reg}}(\lambda). \tag{48}$$

By simple differentiation, $\xi^* = \arg\min_{\boldsymbol{\xi} \in \mathbb{R}} e(\lambda, \xi) = -\frac{b}{2a}$ (which is what we obtained in Corollary 3.5). A little bit of algebra gives us:

$$e(\lambda, \xi^*) = c - \frac{b^2}{4a}. \tag{49}$$

Plugging in the values of a, b and c from Equation (48) in yields:

$$e_{\text{sd}}(\lambda) := e(\lambda, \xi^*) = e_{\text{reg}}(\lambda) - \frac{\left(\sum_{j=1}^{r} \left(\theta_j^* - \frac{\gamma^2}{\lambda}\right) \frac{c_j^2}{(1+c_j)^3}\right)^2}{\sum_{j=1}^{r} \left(\frac{\gamma^2}{\lambda} c_j + \theta_j^*\right) \frac{c_j^2}{(1+c_j)^4}}. \tag{50}$$

This finishes the proof. ■

F. Detailed Version and Proof of Theorem 3.9

Theorem F.1 (**Detailed Version of Theorem 3.9**). *Without loss of generality, let $\|\boldsymbol{\theta}^*\| = 1$ and $\sigma_1 = 1$. Further, suppose $\sigma_j \leq \delta$ for $j \in \{q+1, \ldots, r\}$ and $\theta_1^* > \ldots > \theta_q^*$. Also, suppose $\lambda_{reg}^* > 0$. For any $\nu > 1$, if $\delta \leq \frac{1}{\sqrt{2\nu r}} \sqrt{\min_{k \in \{1,\ldots,q\}} \left(\sigma_k^2 (1-\sigma_k^2)(\theta_1^* - \theta_k^*)\right)}$ and $\gamma^2 \geq \frac{\max_{j \in \{1,\ldots,r\}} \theta_j^*}{\nu - 1}$, then $\lambda = \lambda_{reg}^*$ is a **local maximum** point of $e_{\text{sd}}(\lambda)$.*

Theorem 3.9 is obtained by using $\nu = r$ in Theorem F.1.

Proof. Define $v_k := \sum_{j=1}^{k-1} \frac{\sigma_j^2 \sigma_k^2 (\sigma_j^2 - \sigma_k^2)(\theta_k^* - \theta_j^*)}{(\lambda_{\text{reg}}^* + \sigma_j^2)^4 (\lambda_{\text{reg}}^* + \sigma_k^2)^4}$. For $\lambda = \lambda_{\text{reg}}^*$ to be a local maximum point of $e_{\text{sd}}(\lambda)$, we must have $\sum_{k=1}^{r} v_k < 0$ as per Theorem 3.8.

Let us analyze v_k for $k > q$ first. Using $\sigma_k \leq \delta$ for $k > q$, $(\sigma_j^2 - \sigma_k^2) \leq \sigma_j^2 \leq \sigma_1^2 = 1$ for $j < k$ and $|\theta_k^* - \theta_j^*| \leq \|\boldsymbol{\theta}^*\|^2 = 1$, we get for $k > q$:

$$|v_k| \leq \delta^2 \sum_{j=1}^{k-1} \frac{\sigma_j^2}{(\lambda_{\text{reg}}^* + \sigma_j^2)^4 (\lambda_{\text{reg}}^* + \sigma_k^2)^4}. \tag{51}$$

Now since $\lambda_{\text{reg}}^* > 0$, we can further simplify Equation (51):

$$|v_k| \leq \delta^2 \sum_{j=1}^{k-1} \frac{\sigma_j^2}{(\lambda_{\text{reg}}^*)^8} = \frac{\delta^2}{(\lambda_{\text{reg}}^*)^8} \left(\sum_{j=1}^{q} \underbrace{\sigma_j^2}_{\leq 1} + \sum_{j=q+1}^{r} \underbrace{\sigma_j^2}_{\leq \delta^2} \right) \leq \frac{\delta^2 (q + r\delta^2)}{(\lambda_{\text{reg}}^*)^8}. \tag{52}$$

Summing up Equation (52) from $k = q+1$ through to $k = r$, we get:

$$\sum_{k=q+1}^{r} v_k \leq \sum_{k=q+1}^{r} |v_k| \leq \frac{r\delta^2 (q + r\delta^2)}{(\lambda_{\text{reg}}^*)^8}. \tag{53}$$

Let us now look at $k \leq q$. Since $\theta_1^* > \ldots > \theta_q^*$, we have that $v_k < 0$ for all $k \leq q$. Note that for each $k \leq q$:

$$v_k \leq \frac{\sigma_k^2 (1-\sigma_k^2)(\theta_k^* - \theta_1^*)}{(\lambda_{\text{reg}}^* + 1)^4 (\lambda_{\text{reg}}^* + \sigma_k^2)^4} \leq \frac{\sigma_k^2 (1-\sigma_k^2)(\theta_k^* - \theta_1^*)}{(\lambda_{\text{reg}}^* + 1)^8}, \tag{54}$$

where the last step follows using $\lambda_{\text{reg}}^* > 0$. Thus,

$$\sum_{k=1}^{q} v_k \leq \frac{1}{(\lambda_{\text{reg}}^* + 1)^8} \sum_{k=1}^{q} \sigma_k^2 (1-\sigma_k^2)(\theta_k^* - \theta_1^*) \leq \frac{-q \min_{k \in \{1,\ldots,q\}} \left(\sigma_k^2 (1-\sigma_k^2)(\theta_1^* - \theta_k^*)\right)}{(\lambda_{\text{reg}}^* + 1)^8}. \tag{55}$$

Using Equation (53) and Equation (55), we get:

$$\sum_{k=1}^{r} v_k = \sum_{k=1}^{q} v_k + \sum_{k=q+1}^{r} v_k \leq -\frac{q \min_{k\in\{1,\dots,q\}} \left(\sigma_k^2(1-\sigma_k^2)(\theta_1^* - \theta_k^*)\right)}{(\lambda_{\text{reg}}^* + 1)^8} + \frac{r\delta^2(q + r\delta^2)}{(\lambda_{\text{reg}}^*)^8}. \tag{56}$$

So to ensure $\sum_{k=1}^{r} v_k < 0$, ensuring:

$$\frac{r\delta^2(q + r\delta^2)}{(\lambda_{\text{reg}}^*)^8} < \frac{q \min_{k\in\{1,\dots,q\}} \left(\sigma_k^2(1-\sigma_k^2)(\theta_1^* - \theta_k^*)\right)}{(\lambda_{\text{reg}}^* + 1)^8} \tag{57}$$

suffices. This implies:

$$\frac{\lambda_{\text{reg}}^* + 1}{\lambda_{\text{reg}}^*} < \underbrace{\left(\frac{q \min_{k\in\{1,\dots,q\}} \left(\sigma_k^2(1-\sigma_k^2)(\theta_1^* - \theta_k^*)\right)}{r\delta^2(q + r\delta^2)}\right)^{1/8}}_{:=z}. \tag{58}$$

For any $\nu > 1$, note that $z > \nu$ for $\delta^2 < \frac{1}{2\nu r} \min_{k\in\{1,\dots,q\}} \left(\sigma_k^2(1-\sigma_k^2)(\theta_1^* - \theta_k^*)\right)$. In that case, we must have $\lambda_{\text{reg}}^* > \frac{1}{z-1}$, which can be ensured by having:

$$\lambda_{\text{reg}}^* > \frac{1}{\nu - 1}. \tag{59}$$

From Equation (42), recall that $\lambda_{\text{reg}}^* = \frac{\gamma^2 \sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda_{\text{reg}}^* + \sigma_j^2)^3}}{\sum_{j=1}^{r} \frac{\theta_j^* \sigma_j^2}{(\lambda_{\text{reg}}^* + \sigma_j^2)^3}}$. Now since $\lambda_{\text{reg}}^* > 0$, we have that:

$$\lambda_{\text{reg}}^* \geq \frac{\gamma^2 \sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda_{\text{reg}}^* + \sigma_j^2)^3}}{\theta_{\max}^* \sum_{j=1}^{r} \frac{\sigma_j^2}{(\lambda_{\text{reg}}^* + \sigma_j^2)^3}} \geq \frac{\gamma^2}{\theta_{\max}^*}, \tag{60}$$

where $\theta_{\max}^* = \max_{j\in\{1,\dots,r\}} \theta_j^*$. Using this, if $\gamma^2 > \frac{\theta_{\max}^*}{\nu - 1}$, then $\lambda_{\text{reg}}^* \geq \frac{\gamma^2}{\theta_{\max}^*} > \frac{1}{\nu-1} > \frac{1}{z-1}$. This completes the proof. ■

G. Proof of Theorem 3.10

Proof. We provide a 2-dimensional example, i.e., $d = 2$. Suppose $n > 2$. Take $\boldsymbol{\theta}^* = \frac{1}{\sqrt{2}}(\boldsymbol{u}_1 + \boldsymbol{u}_2)$; so, $\theta_1^* = \theta_2^* = \frac{1}{2}$. Also, suppose $\sigma_1 = 1$ and $\sigma_2 = \frac{1}{2}$. For this case, we get (by using the formulas in Theorem E.1):

$$e_{\text{reg}}(\lambda) = \frac{\lambda^2}{2}\left(\frac{1}{(\lambda+1)^2} + \frac{16}{(4\lambda+1)^2}\right) + \gamma^2\left(\frac{1}{(\lambda+1)^2} + \frac{4}{(4\lambda+1)^2}\right), \tag{61}$$

and

$$e'_{\text{reg}}(\lambda) = (\lambda - 2\gamma^2)\left(\frac{1}{(\lambda+1)^3} + \frac{16}{(4\lambda+1)^3}\right). \tag{62}$$

From Equation (62), we have that $\lambda_{\text{reg}}^* = \arg\min_{\lambda>0} e_{\text{reg}}(\lambda) = 2\gamma^2$.

From Theorem E.1, we have that:

$$e'_{\text{sd}}(\lambda) = e'_{\text{reg}}(\lambda)\left(1 - \frac{2e''_{\text{reg}}(\lambda)}{h(\lambda)} + \frac{e'_{\text{reg}}(\lambda)h'(\lambda)}{(h(\lambda))^2}\right), \text{ where } h(\lambda) = \left(\frac{4\gamma^2+2}{(\lambda+1)^4} + \frac{256\gamma^2+32}{(4\lambda+1)^4}\right). \tag{63}$$

After a lot of algebraic heavy lifting, we get:

$$e'_{\text{sd}}(\lambda) = \frac{288(\lambda - 2\gamma^2)^3}{(\lambda+1)^5(4\lambda+1)^5\left(\frac{2\gamma^2+1}{(\lambda+1)^4} + \frac{128\gamma^2+16}{(4\lambda+1)^4}\right)^2}\left(\frac{1}{(\lambda+1)^3} + \frac{16}{(4\lambda+1)^3}\right). \tag{64}$$

Using Equation (64), we can conclude that $\arg\min_{\lambda>0} e_{\text{sd}}(\lambda) = 2\gamma^2 = \lambda_{\text{reg}}^*$. ■

H. Empirical Motivation for Assumption 4.2

We consider the same logistic regression setting as Section 4. Note that the kernel matrix $\boldsymbol{K} \in \mathbb{R}^{2n\times 2n}$ (w.r.t. $\phi(.)$) is of the form $\boldsymbol{K} = \begin{bmatrix} \boldsymbol{K}_1 & \mathbf{0}_{n\times n} \\ \mathbf{0}_{n\times n} & \boldsymbol{K}_0 \end{bmatrix}$, where $\mathbf{0}_{n\times n}$ is the $n \times n$ matrix of all 0's and $\boldsymbol{K}_1$ and $\boldsymbol{K}_0$ are both PSD matrices with diagonal entries = 1. For our simulations, the diagonal elements of $\boldsymbol{K}_1$ are set equal to 1 and the off-diagonal elements are set equal to the corresponding off-diagonal element of $\frac{1}{n}\boldsymbol{Z}_1\boldsymbol{Z}_1^T$, where each element of $\boldsymbol{Z}_1 \in \mathbb{R}^{n\times n}$ is drawn i.i.d. from **(i)** Unif$[0,1]$, and **(ii)** Bernoulli(0.8)[15]. $\boldsymbol{K}_0$ is constructed in the same way. Note that $\boldsymbol{K}$ is PSD. In the case of **(i)** (resp., **(ii)**), the expected off-diagonal element of both $\boldsymbol{K}_1$ and $\boldsymbol{K}_0$ is 0.25 (resp., 0.64), and so we compare against Assumption 4.2 with $c = 0.25$ (resp., $c = 0.64$) and $n \to \infty$. We consider four values of n, namely, 1000, 5000, 10000 and 50000.

In Table 4, we show results for **(i)** when $p = 0.45$ (top) and $p = 0.35$ (bottom) with $\hat{\lambda} = 1 - c$ (recall that $\hat{\lambda} \in \left[\frac{1-c}{2.16}, \frac{1-c}{0.40}\right]$ as per Theorem 4.3). In Table 5, we show results for **(ii)** when $p = 0.3$ (top) and $p = 0.2$ (bottom) with $\hat{\lambda} = \frac{1-c}{0.50} = 2(1-c)$. Please see the table captions for a detailed discussion, but in summary, we conclude that Assumption 4.2 is a reasonable assumption to analyze the average behavior of a linear model on a large dataset under random label corruption.

I. Proof of Theorem 4.3

I.1. Step 1 in Detail

The teacher's estimated parameter $\boldsymbol{\theta}_{\mathrm{T}}^* := \arg\min_{\boldsymbol{\theta}} f_{\mathrm{T}}(\boldsymbol{\theta})$ satisfies $\nabla f_{\mathrm{T}}(\boldsymbol{\theta}_{\mathrm{T}}^*) = \frac{1}{2n}\sum_{i=1}^{2n}\Big(\sigma(\langle \boldsymbol{\theta}_{\mathrm{T}}^*, \phi(\boldsymbol{x}_i)\rangle) - \hat{y}_i\Big)\phi(\boldsymbol{x}_i) + \lambda\boldsymbol{\theta}_{\mathrm{T}}^* = \vec{0}$. From this, we get:

$$\boldsymbol{\theta}_{\mathrm{T}}^* = \sum_{i=1}^{2n} \underbrace{\frac{1}{2n\lambda}\Big(\hat{y}_i - \sigma(\langle \boldsymbol{\theta}_{\mathrm{T}}^*, \phi(\boldsymbol{x}_i)\rangle)\Big)}_{:=\alpha_i} \phi(\boldsymbol{x}_i) = \sum_{i=1}^{2n} \alpha_i \phi(\boldsymbol{x}_i), \tag{65}$$

for some real numbers $\{\alpha_i\}_{i=1}^{2n}$ which are known as the teacher's dual-space coordinates. Recall that we defined $\hat{\lambda} := 2n\lambda$ in the theorem statement.

Lemma I.1 (**Teacher's Dual-Space Coordinates and Predictions**). *Suppose Assumptions 4.1 and 4.2 hold. Then:*

$$\alpha_i = \begin{cases} -\hat{\alpha} \textit{ for } i \in \mathcal{S}_{1,\text{bad}}, \\ \alpha \textit{ for } i \in \mathcal{S}_{1,\text{good}}, \\ \hat{\alpha} \textit{ for } i \in \mathcal{S}_{0,\text{bad}}, \\ -\alpha \textit{ for } i \in \mathcal{S}_{0,\text{good}}, \end{cases} \tag{66}$$

where $\alpha \geq 0$ and $\hat{\alpha} \geq 0$ are obtained by jointly solving:

$$\sigma\Big(cn\big(\alpha - (\alpha + \hat{\alpha})p\big) - (1-c)\hat{\alpha}\Big) = \hat{\lambda}\hat{\alpha}, \tag{67}$$

and

$$\sigma\Big(cn\big(\alpha - (\alpha + \hat{\alpha})p\big) + (1-c)\alpha\Big) = 1 - \hat{\lambda}\alpha. \tag{68}$$

Also, the teacher's prediction for the ith sample, $y_i^{(\mathrm{T})}$, turns out to be:

$$y_i^{(\mathrm{T})} = \begin{cases} \hat{\lambda}\hat{\alpha} \textit{ for } i \in \mathcal{S}_{1,\text{bad}}, \\ 1 - \hat{\lambda}\alpha \textit{ for } i \in \mathcal{S}_{1,\text{good}}, \\ 1 - \hat{\lambda}\hat{\alpha} \textit{ for } i \in \mathcal{S}_{0,\text{bad}}, \\ \hat{\lambda}\alpha \textit{ for } i \in \mathcal{S}_{0,\text{good}}. \end{cases} \tag{69}$$

Lemma I.1 is proved next in Appendix I.2.

[15] If $X \sim$ Bernoulli(p), then $\mathbb{P}(X = 1) = p$ and $\mathbb{P}(X = 0) = 1 - p$.

Table 4. **(i) Unif**$[0, 1]$**:** Results (up to fourth decimal point) for $p = 0.45$ (top) and $p = 0.35$ (bottom) with $\hat{\lambda} = 1 - c$ on points with true label = 1; points with true label = 0 follow the same trend by symmetry of the problem. In the table, "bad" (resp., "good") points mean incorrectly (resp., correctly) labeled points, and A4.2 is Assumption 4.2. Also, "pred." is the predicted probability of the label being 1 and "Avg. pred. for bad points" (resp., "Avg. pred. for good points") is the empirical average over all bad (resp., good) points with true label = 1. Under Assumption 4.2, all bad/good points have the same prediction (see Equations (11) and (14) or Lemmas I.1 and I.2) due to which the corresponding columns do not have the word "Avg.". *Observe that as n increases, the average prediction for both good and bad points (with the simulated kernel matrix) matches the corresponding predictions under Assumption 4.2 (and $n \to \infty$). Thus, Assumption 4.2 is a reasonable assumption to analyze the average behavior of a linear model on a large dataset under random label corruption.*

	n	Avg. pred. for bad points	Pred. for bad points under A4.2 & $n \to \infty$	Avg. pred. for good points	Pred. for good points under A4.2 & $n \to \infty$
`Teacher`	1k	0.4413	**0.4400**	0.6372	**0.6400**
	5k	0.4399		0.6397	
	10k	0.4399		0.6399	
	50k	**0.4400**		**0.6400**	
	n	Avg. pred. for bad points	Pred. for bad points under A4.2 & $n \to \infty$	Avg. pred. for good points	Pred. for good points under A4.2 & $n \to \infty$
`Student`	1k	0.5287	**0.5280**	0.5645	**0.5680**
	5k	0.5279		0.5676	
	10k	0.5279		0.5679	
	50k	**0.5280**		**0.5680**	

(a) $p = 0.45$

	n	Avg. pred. for bad points	Pred. for bad points under A4.2 & $n \to \infty$	Avg. pred. for good points	Pred. for good points under A4.2 & $n \to \infty$
`Teacher`	1k	0.5243	**0.5200**	0.7146	**0.7200**
	5k	0.5198		0.7195	
	10k	0.5198		0.7198	
	50k	**0.5200**		**0.7200**	
	n	Avg. pred. for bad points	Pred. for bad points under A4.2 & $n \to \infty$	Avg. pred. for good points	Pred. for good points under A4.2 & $n \to \infty$
`Student`	1k	0.6264	**0.6240**	0.6568	**0.6640**
	5k	0.6235		0.6631	
	10k	0.6236		0.6636	
	50k	**0.6240**		**0.6640**	

(b) $p = 0.35$

As mentioned in the proof sketch in the main text, we shall focus on the interesting case of:
(a) p being large enough so that the teacher misclassifies the incorrectly labeled points because otherwise, there is no need for SD, and
(b) $\hat{\lambda}$ being chosen sensibly so that the teacher at least correctly classifies the correctly labeled points because otherwise, SD is hopeless.

Later in Appendix I.5, we shall impose a lower bound on p (in terms of c and $\hat{\lambda}$) so that (a) is ensured. Specifically, the teacher misclassifies the incorrectly labeled points (with indices $\mathcal{S}_{1,\text{bad}} = \{1, \ldots, \hat{n}\}$ and $\mathcal{S}_{0,\text{bad}} = \{n+1, \ldots, n+\hat{n}\}$) when

$$\hat{\lambda}\hat{\alpha} < \frac{1}{2}. \tag{70}$$

Table 5. **(ii) Bernoulli**(0.8)**:** Same as Table 4 except for $p = 0.3$ (top) and $p = 0.2$ (bottom) with $\hat{\lambda} = 2(1-c)$. Just like in Table 4, *as n increases, the average prediction for both good and bad points (with the simulated kernel matrix) matches the corresponding predictions under Assumption 4.2 (and $n \to \infty$). Thus, Assumption 4.2 is a reasonable assumption to analyze the average behavior of a linear model on a large dataset under random label corruption.*

	n	Avg. pred. for bad points	Pred. for bad points under A4.2 & $n \to \infty$	Avg. pred. for good points	Pred. for good points under A4.2 & $n \to \infty$
`Teacher`	1k	0.6213	**0.6222**	0.7324	**0.7333**
	5k	0.6220		0.7332	
	10k	0.6221		0.7332	
	50k	**0.6222**		**0.7333**	
`Student`	1k	0.6895	**0.6913**	0.7018	**0.7037**
	5k	0.6910		0.7033	
	10k	0.6911		0.7035	
	50k	**0.6913**		**0.7037**	

(a) $p = 0.3$

	n	Avg. pred. for bad points	Pred. for bad points under A4.2 & $n \to \infty$	Avg. pred. for good points	Pred. for good points under A4.2 & $n \to \infty$
`Teacher`	1k	0.7097	**0.7111**	0.8208	**0.8222**
	5k	0.7108		0.8219	
	10k	0.7109		0.8221	
	50k	**0.7111**		**0.8222**	
`Student`	1k	0.7872	**0.7901**	0.7995	**0.8024**
	5k	0.7895		0.8019	
	10k	0.7898		0.8021	
	50k	**0.7901**		**0.8024**	

(b) $p = 0.2$

Moreover, in Appendix I.5, we shall also restrict $\hat{\lambda}$ (in terms of c) so that (b) is ensured. Specifically, the teacher correctly classifies the correctly labeled points (with indices $\mathcal{S}_{1,\text{good}} = \{\hat{n}+1, \dots, n\}$ and $\mathcal{S}_{0,\text{good}} = \{n+\hat{n}+1, \dots, 2n\}$) when

$$1 - \hat{\lambda}\alpha > \frac{1}{2} \implies \hat{\lambda}\alpha < \frac{1}{2}. \tag{71}$$

I.2. Proof of Lemma I.1

Proof. From Equation (65), we have:

$$2n\lambda\alpha_i = \hat{y}_i - \sigma\Big(\sum_{j=1}^{2n} \alpha_j \langle \phi(\boldsymbol{x}_j), \phi(\boldsymbol{x}_i)\rangle\Big), \tag{72}$$

for all $i \in \{1, \dots, 2n\}$. For ease of notation, let us define $v_i := \sum_{j=1}^{2n} \alpha_j \langle \phi(\boldsymbol{x}_j), \phi(\boldsymbol{x}_i)\rangle$. Then, the above equation can be rewritten as:

$$2n\lambda\alpha_i = \hat{y}_i - \sigma(v_i). \tag{73}$$

Note here that the teacher's predictions are:

$$y_i^{(\mathrm{T})} := \sigma(v_i) = \hat{y}_i - 2n\lambda\alpha_i, \tag{74}$$

for $i \in \{1, \dots, 2n\}$. Next, using Assumptions 4.1 and 4.2, we have:

$$v_i = \begin{cases} \alpha_i + c\sum_{j\in\{1,\dots,n\}\setminus i}\alpha_j = \alpha_i(1-c) + c\sum_{j=1}^{n}\alpha_j \text{ for } i \in \{1,\dots,n\}, \\ \alpha_i + c\sum_{j\in\{n+1,\dots,2n\}\setminus i}^{n}\alpha_j = \alpha_i(1-c) + c\sum_{j=n+1}^{2n}\alpha_j \text{ for } i \in \{n+1,\dots,2n\}. \end{cases} \tag{75}$$

Let us focus on $i \in \{1, \dots, n\}$. Let $S = \sum_{j=1}^{n}\alpha_j$. Then, we have the following equations:

$$2n\lambda\alpha_i = -\sigma(\alpha_i(1-c) + cS) \text{ for } i \in \{1,\dots,\hat{n}\}, \tag{76}$$

and

$$2n\lambda\alpha_i = 1 - \sigma(\alpha_i(1-c) + cS) \text{ for } i \in \{\hat{n}+1,\dots,n\}. \tag{77}$$

Using the monotonicity of the sigmoid function, we conclude that:

$$\alpha_i = \begin{cases} -\hat{\alpha} \text{ for } i \in \{1,\dots,\hat{n}\} \\ \alpha \text{ for } i \in \{\hat{n}+1,\dots,n\}, \end{cases} \tag{78}$$

for some $\alpha, \hat{\alpha} \geq 0$. Using a similar argument, we can conclude that for $i \in \{n+1, \dots, 2n\}$:

$$\alpha_i = \begin{cases} \hat{\alpha}_2 \text{ for } i \in \{n+1,\dots,n+\hat{n}\} \\ -\alpha_2 \text{ for } i \in \{n+\hat{n}+1,\dots,2n\}, \end{cases} \tag{79}$$

for some $\alpha_2, \hat{\alpha}_2 \geq 0$. We further claim that:

$$\alpha_2 = \alpha \text{ and } \hat{\alpha}_2 = \hat{\alpha}. \tag{80}$$

Let us verify if this indeed holds up. Note that with such a solution:

$$\sum_{j=1}^{n}\alpha_j = -\sum_{j=n+1}^{2n}\alpha_j = \alpha(n-\hat{n}) - \hat{\alpha}\hat{n} = \alpha n - (\alpha+\hat{\alpha})\hat{n}. \tag{81}$$

Plugging this back in Equation (75) for $i \in \{1, \dots, n\}$ and then in Equation (73), we get (after a bit of rewriting):

$$\sigma\big(-(1-c)\hat{\alpha} + c\alpha n - c(\alpha+\hat{\alpha})\hat{n}\big) = 2n\lambda\hat{\alpha}. \tag{82}$$

$$\sigma\big((1-c)\alpha + c\alpha n - c(\alpha+\hat{\alpha})\hat{n}\big) = 1 - 2n\lambda\alpha. \tag{83}$$

Doing the same but for $i \in \{n+1, \dots, 2n\}$ with $\alpha_2 = \alpha$ and $\hat{\alpha}_2 = \hat{\alpha}$, we get (again, after a bit of rewriting):

$$\sigma\big((1-c)\hat{\alpha} - c\alpha n + c(\alpha+\hat{\alpha})\hat{n}\big) = 1 - 2n\lambda\hat{\alpha}. \tag{84}$$

$$\sigma\big(-(1-c)\alpha - c\alpha n + c(\alpha+\hat{\alpha})\hat{n}\big) = 2n\lambda\alpha. \tag{85}$$

Now note that Equation (82) and Equation (84), and Equation (83) and Equation (85) are the same – this is because $\sigma(-z) = 1 - \sigma(z)$ for all $z \in \mathbb{R}$. Thus, our claim in Equation (80) is true.

Hence, we can consider only Equation (82) and Equation (83), and solve them to find the two unknown variables α and $\hat{\alpha}$ in order to obtain $\boldsymbol{\theta}_{\mathrm{T}}^*$. Recalling $\hat{n} = np$, we can rewrite Equation (82) and Equation (83) as follows:

$$\sigma\Big(cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\hat{\alpha}\Big) = 2n\lambda\hat{\alpha}. \tag{86}$$

$$\sigma\Big(cn\big(\alpha - (\alpha+\hat{\alpha})p\big) + (1-c)\alpha\Big) = 1 - 2n\lambda\alpha. \tag{87}$$

Thus, we have:

$$\alpha_i = \begin{cases} -\hat{\alpha} \text{ for } i \in \{1, \dots, \hat{n}\}, \\ \alpha \text{ for } i \in \{\hat{n}+1, \dots, n\}, \\ \hat{\alpha} \text{ for } i \in \{n+1, \dots, n+\hat{n}\}, \\ -\alpha \text{ for } i \in \{n+\hat{n}+1, \dots, 2n\}, \end{cases} \tag{88}$$

where α and $\hat{\alpha}$ are obtained by solving Equation (86) and Equation (87).

From Equation (74), recall that the teacher's predictions for the ith sample is:

$$y_i^{(\text{T})} := \hat{y}_i - 2n\lambda\alpha_i. \tag{89}$$

Now using Equation (88) in Equation (89), we get:

$$y_i^{(\text{T})} = \begin{cases} 2n\lambda\hat{\alpha} \text{ for } i \in \{1, \dots, \hat{n}\}, \\ 1 - 2n\lambda\alpha \text{ for } i \in \{\hat{n}+1, \dots, n\}, \\ 1 - 2n\lambda\hat{\alpha} \text{ for } i \in \{n+1, \dots, n+\hat{n}\}, \\ 2n\lambda\alpha \text{ for } i \in \{n+\hat{n}+1, \dots, 2n\}. \end{cases} \tag{90}$$

Replacing $2n\lambda$ with $\hat{\lambda}$ in Equations (86), (87) and (90), and plugging in $\mathcal{S}_{1,\text{bad}} = \{1, \dots, \hat{n}\}$, $\mathcal{S}_{1,\text{good}} = \{\hat{n}+1, \dots, n\}$, $\mathcal{S}_{0,\text{bad}} = \{n+1, \dots, n+\hat{n}\}$ and $\mathcal{S}_{0,\text{good}} = \{n+\hat{n}+1, \dots, 2n\}$ throughout finishes the proof. ■

I.3. Step 2 in Detail

Just like Equation (65) for the teacher, it can be shown that:

$$\boldsymbol{\theta}_{\text{S}}^* = \sum_{i=1}^{2n} \beta_i \phi(\boldsymbol{x}_i), \tag{91}$$

for some real numbers $\{\beta_i\}_{i=1}^{2n}$ which are known as the student's dual-space coordinates.

Lemma I.2 (**Student's Dual-Space Coordinates and Predictions**). *Suppose Assumptions 4.1 and 4.2 hold, and the teacher correctly classifies the correctly labeled points but misclassifies the incorrectly labeled points, i.e., $\hat{\lambda}\alpha < \frac{1}{2}$ and $\hat{\lambda}\hat{\alpha} < \frac{1}{2}$ in Lemma I.1. Then:*

$$\beta_i = \begin{cases} -\hat{\beta} \textit{ for } i \in \mathcal{S}_{1,\text{bad}}, \\ \beta \textit{ for } i \in \mathcal{S}_{1,\text{good}}, \\ \hat{\beta} \textit{ for } i \in \mathcal{S}_{0,\text{bad}}, \\ -\beta \textit{ for } i \in \mathcal{S}_{0,\text{good}}, \end{cases} \tag{92}$$

where $\beta \geq 0$ and $\hat{\beta} \geq 0$ are obtained by jointly solving:

$$\sigma\Big(cn\big(\beta - (\beta + \hat{\beta})p\big) - (1-c)\hat{\beta}\Big) = \hat{\lambda}\hat{\alpha} + \hat{\lambda}\hat{\beta}, \tag{93}$$

and

$$\sigma\Big(cn\big(\beta - (\beta + \hat{\beta})p\big) + (1-c)\beta\Big) = 1 - \hat{\lambda}\alpha - \hat{\lambda}\beta. \tag{94}$$

Also, the student's prediction for the ith sample, $y_i^{(\text{S})}$, turns out to be:

$$y_i^{(\text{S})} = \begin{cases} \hat{\lambda}\hat{\alpha} + \hat{\lambda}\hat{\beta} \textit{ for } i \in \mathcal{S}_{1,\text{bad}}, \\ 1 - \hat{\lambda}\alpha - \hat{\lambda}\beta \textit{ for } i \in \mathcal{S}_{1,\text{good}}, \\ 1 - \hat{\lambda}\hat{\alpha} - \hat{\lambda}\hat{\beta} \textit{ for } i \in \mathcal{S}_{0,\text{bad}}, \\ \hat{\lambda}\alpha + \hat{\lambda}\beta \textit{ for } i \in \mathcal{S}_{0,\text{good}}. \end{cases} \tag{95}$$

We prove Lemma I.2 in Appendix I.4.

Now note that if $\hat{\lambda}\hat{\alpha} + \hat{\lambda}\hat{\beta} > \frac{1}{2}$ and $\hat{\lambda}\alpha + \hat{\lambda}\beta < \frac{1}{2}$, then the student has managed to correctly classify all the points in the training set. We ensure this in Appendix I.5 by imposing an upper bound on p.

I.4. Proof of Lemma I.2

Proof. The student's estimated parameter $\boldsymbol{\theta}_{\mathrm{S}}^* = \arg\min_{\boldsymbol{\theta}} f_{\mathrm{S}}(\boldsymbol{\theta})$ satisfies $\nabla f_{\mathrm{S}}(\boldsymbol{\theta}_{\mathrm{S}}^*) = \vec{0}$, from which we get:

$$\boldsymbol{\theta}_{\mathrm{S}}^* = \sum_{i=1}^{2n} \underbrace{\frac{1}{2n\lambda}\Big(y_i^{(\mathrm{T})} - \sigma(\langle \boldsymbol{\theta}_{\mathrm{S}}^*, \phi(\boldsymbol{x}_i)\rangle)\Big)}_{:=\beta_i} \phi(\boldsymbol{x}_i). \tag{96}$$

Thus the student's i^{th} dual coordinate β_i (as defined in Equation (91)) satisfies:

$$2n\lambda\beta_i = y_i^{(\mathrm{T})} - \sigma(\langle \boldsymbol{\theta}_{\mathrm{S}}^*, \phi(\boldsymbol{x}_i)\rangle). \tag{97}$$

By following the same approach as the one we took in the proof of Lemma I.1 for the teacher (with hard labels replaced by soft labels), we can show that:

$$\beta_i = \begin{cases} -\hat{\beta} \text{ for } i \in \{1, \ldots, \hat{n}\}, \\ \beta \text{ for } i \in \{\hat{n}+1, \ldots, n\}, \\ \hat{\beta} \text{ for } i \in \{n+1, \ldots, n+\hat{n}\}, \\ -\beta \text{ for } i \in \{n+\hat{n}+1, \ldots, 2n\}, \end{cases} \tag{98}$$

where $\beta \in \mathbb{R}$ and $\hat{\beta} \in \mathbb{R}$ are obtained by solving the following two equations:

$$\sigma\Big(cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\hat{\beta}\Big) = 2n\lambda\hat{\alpha} + 2n\lambda\hat{\beta}, \tag{99}$$

and

$$\sigma\Big(cn\big(\beta - (\beta+\hat{\beta})p\big) + (1-c)\beta\Big) = 1 - 2n\lambda\alpha - 2n\lambda\beta. \tag{100}$$

We shall now show that $\beta \geq 0$ and $\hat{\beta} \geq 0$. We shall prove this by contradiction – specifically, by showing that the other cases lead to a contradiction.

Case 1: $\beta \leq 0$ and $\hat{\beta} \leq 0$. In this case:

$$cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\hat{\beta} \geq cn\big(\beta - (\beta+\hat{\beta})p\big) + (1-c)\beta, \tag{101}$$

which implies (by the increasing nature of the sigmoid function):

$$\underbrace{\sigma\Big(cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\hat{\beta}\Big)}_{=2n\lambda\hat{\alpha}+2n\lambda\hat{\beta} \text{ from Equation (99)}} \geq \underbrace{\sigma\Big(cn\big(\beta - (\beta+\hat{\beta})p\big) + (1-c)\beta\Big)}_{=1-2n\lambda\alpha-2n\lambda\beta \text{ from Equation (100)}}. \tag{102}$$

Now using Equation (99) and Equation (100), we get:

$$2n\lambda\hat{\alpha} + 2n\lambda\hat{\beta} \geq 1 - 2n\lambda\alpha - 2n\lambda\beta \implies 2n\lambda\hat{\alpha} \geq 1 - 2n\lambda\alpha \underbrace{-2n\lambda(\beta+\hat{\beta})}_{\geq 0} \implies 2n\lambda\hat{\alpha} \geq 1 - 2n\lambda\alpha. \tag{103}$$

But this is a contradiction because as per Equation (70) and Equation (71), we had:

$$2n\lambda\hat{\alpha} < \frac{1}{2} \text{ and } 1 - 2n\lambda\alpha > \frac{1}{2} \implies 2n\lambda\hat{\alpha} < 1 - 2n\lambda\alpha. \tag{104}$$

Hence, $\beta \leq 0$ and $\hat{\beta} \leq 0$ is not possible.

Case 2: $\beta \geq 0$ and $\hat{\beta} \leq 0$. In this case:

$$cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\hat{\beta} \geq 0 \implies \underbrace{\sigma\Big(cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\hat{\beta}\Big)}_{=2n\lambda\hat{\alpha}+2n\lambda\hat{\beta} \text{ from Equation (99)}} \geq \frac{1}{2}. \tag{105}$$

Using the above and Equation (99), we get that:

$$2n\lambda\hat{\alpha} + \underbrace{2n\lambda\hat{\beta}}_{\leq 0} \geq \frac{1}{2} \implies 2n\lambda\hat{\alpha} \geq \frac{1}{2}. \tag{106}$$

But this is again a contradiction as $2n\lambda\hat{\alpha} < \frac{1}{2}$ as per Equation (70). Hence, $\beta \geq 0$ and $\hat{\beta} \leq 0$ is also ruled out.

Case 3: $\beta \leq 0$ **and** $\hat{\beta} \geq 0$**.** In this case:

$$cn\big(\beta - (\beta + \hat{\beta})p\big) + (1-c)\beta \leq 0 \implies \underbrace{\sigma\Big(cn\big(\beta - (\beta + \hat{\beta})p\big) + (1-c)\beta\Big)}_{=1-2n\lambda\alpha-2n\lambda\beta \text{ from Equation (100)}} \leq \frac{1}{2}. \tag{107}$$

Using the above and Equation (100), we get that:

$$1 - 2n\lambda\alpha - \underbrace{2n\lambda\beta}_{\leq 0} \leq \frac{1}{2} \implies 1 - 2n\lambda\alpha \leq \frac{1}{2}. \tag{108}$$

But this is also a contradiction as $1 - 2n\lambda\alpha > \frac{1}{2}$ as per Equation (71). Hence, $\beta \leq 0$ and $\hat{\beta} \geq 0$ is also ruled out.

So, only $\beta \geq 0$ and $\hat{\beta} \geq 0$ is possible. Recall that β and $\hat{\beta}$ are solutions to:

$$\sigma\Big(cn\big(\beta - (\beta + \hat{\beta})p\big) - (1-c)\hat{\beta}\Big) = 2n\lambda\hat{\alpha} + 2n\lambda\hat{\beta}, \tag{109}$$

and

$$\sigma\Big(cn\big(\beta - (\beta + \hat{\beta})p\big) + (1-c)\beta\Big) = 1 - 2n\lambda\alpha - 2n\lambda\beta. \tag{110}$$

Just like we obtained the teacher's predictions $\left\{y_i^{(\mathrm{T})}\right\}_{i=1}^{2n}$, the student's predictions are:

$$y_i^{(\mathrm{S})} = \begin{cases} 2n\lambda\hat{\alpha} + 2n\lambda\hat{\beta} \text{ for } i \in \{1, \ldots, \hat{n}\}, \\ 1 - 2n\lambda\alpha - 2n\lambda\beta \text{ for } i \in \{\hat{n}+1, \ldots, n\}, \\ 1 - 2n\lambda\hat{\alpha} - 2n\lambda\hat{\beta} \text{ for } i \in \{n+1, \ldots, n+\hat{n}\}, \\ 2n\lambda\alpha + 2n\lambda\beta \text{ for } i \in \{n+\hat{n}+1, \ldots, 2n\}. \end{cases} \tag{111}$$

Finally, replacing $2n\lambda$ with $\hat{\lambda}$ in Equations (109), (110) and (111), and plugging in $\mathcal{S}_{1,\text{bad}} = \{1, \ldots, \hat{n}\}$, $\mathcal{S}_{1,\text{good}} = \{\hat{n}+1, \ldots, n\}$, $\mathcal{S}_{0,\text{bad}} = \{n+1, \ldots, n+\hat{n}\}$ and $\mathcal{S}_{0,\text{good}} = \{n+\hat{n}+1, \ldots, 2n\}$ throughout gives us the desired result. ■

I.5. Step 3 in Detail

Proof. Here, we shall obtain analytical expressions for the teacher's and student's predictions by solving Equation (67) and Equation (68) (in Lemma I.1) for the teacher and then Equation (93) and Equation (94) (in Lemma I.2) for the student. Our approach will involve employing the first-order Maclaurin series expansion of the sigmoid function; specifically, we will use:

$$\sigma(z) = \frac{1}{2} + \frac{z}{4} + \varepsilon(z), \tag{112}$$

where $\varepsilon(z)$ is the residual error function. Note that:

$$\varepsilon(z) \begin{cases} < 0 \text{ for } z > 0 \text{ or equivalently when } \sigma(z) > \frac{1}{2} \\ = 0 \text{ for } z = 0 \text{ or equivalently when } \sigma(z) = \frac{1}{2} \\ > 0 \text{ for } z < 0 \text{ or equivalently when } \sigma(z) < \frac{1}{2}. \end{cases} \tag{113}$$

It also holds that $\varepsilon(z)$ is a decreasing function. So,

$$\sup_{z \in [-1,0]} \varepsilon(z) = \varepsilon(-1) < 0.02 \text{ or equivalently } \sup_{z:\sigma(z) \in [\sigma(-1), 0.5]} \varepsilon(z) < 0.02, \tag{114}$$

and

$$\inf_{z\in[0,1]} \varepsilon(z) = \varepsilon(1) > -0.02 \text{ or equivalently } \inf_{z:\sigma(z)\in[0.5,\sigma(1)]} \varepsilon(z) > -0.02. \tag{115}$$

Let us start with the **teacher**. Rewriting Equation (67) and Equation (68) while using the Maclaurin series expansion of the sigmoid function (from Equation (112)) and the fact that $\sigma(-z) = 1 - \sigma(z)\ \forall\ z \in \mathbb{R}$, we have:

$$\hat{\lambda}\hat{\alpha} = \sigma\Big(cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\hat{\alpha}\Big) = \frac{1}{2} + \left(\frac{cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\hat{\alpha}}{4}\right) + \varepsilon_1, \tag{116}$$

and

$$\hat{\lambda}\alpha = \sigma\Big(-cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\alpha\Big) = \frac{1}{2} - \left(\frac{cn\big(\alpha - (\alpha+\hat{\alpha})p\big) + (1-c)\alpha}{4}\right) + \varepsilon_2, \tag{117}$$

for some real numbers $\varepsilon_1, \varepsilon_2$. Solving the above two equations in the limit of $n \to \infty$, when $c = \Theta(1)$ and $\hat{\lambda} < \mathcal{O}(n)$ (this will be ensured subsequently), gives us:

$$\lim_{n\to\infty} \alpha = \frac{p(1+\varepsilon_1+\varepsilon_2)}{\hat{\lambda} + \frac{1-c}{4}} \text{ and } \lim_{n\to\infty} \hat{\alpha} = \frac{(1-p)(1+\varepsilon_1+\varepsilon_2)}{\hat{\lambda} + \frac{1-c}{4}}. \tag{118}$$

Henceforth, we shall drop the $\lim_{n\to\infty}$ notation, and it is implied directly.

Let us now bound $\varepsilon_1 + \varepsilon_2$ by imposing some more constraints. First, recall from Equation (70) and Equation (71) that we want $\hat{\lambda}\hat{\alpha} < \frac{1}{2}$ (i.e., the teacher does *not* correctly classify the incorrectly labeled points) and $\hat{\lambda}\alpha < \frac{1}{2}$ (i.e., the teacher correctly classifies the correctly labeled points). Now since we are solving Equation (116) and Equation (117), we must have $\hat{\lambda}\hat{\alpha} = \sigma\big(cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\hat{\alpha}\big) < \frac{1}{2}$ and $\hat{\lambda}\alpha = \sigma\big(-cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\alpha\big) < \frac{1}{2}$; in this case, we must have that $\varepsilon_1 > 0$ and $\varepsilon_2 > 0$ from Equation (113). Next, we shall obtain upper bounds for ε_1 and ε_2. Using Equation (117), if $\sigma\Big(-cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\alpha\Big) = \hat{\lambda}\alpha > \sigma(-1)$, then $\varepsilon_2 < 0.02$ from Equation (114). Note that since $\varepsilon_1 + \varepsilon_2 > 0$ and $p < \frac{1}{2}$, $\hat{\alpha} > \alpha$. So if $\hat{\lambda}\alpha > \sigma(-1)$ holds, then so does $\hat{\lambda}\hat{\alpha} > \sigma(-1)$, in which case $\varepsilon_1 < 0.02$. But using the fact that $\varepsilon_1 + \varepsilon_2 > 0$, having

$$\frac{\hat{\lambda}p}{\hat{\lambda} + \frac{1-c}{4}} > \sigma(-1), \tag{119}$$

ensures $\hat{\lambda}\alpha > \sigma(-1)$ (as well as, $\hat{\lambda}\hat{\alpha} > \sigma(-1)$). Recalling that $r = \frac{(1-c)/4}{\hat{\lambda}}$ and using the fact that $\sigma(-1) = \frac{1}{1+e}$, we get:

$$p > \frac{1+r}{1+e}. \tag{120}$$

But we must also have $p < \frac{1}{2}$ due to which we should have $\frac{1+r}{1+e} < \frac{1}{2}$; this holds when:

$$r = \frac{(1-c)/4}{\hat{\lambda}} < \frac{e-1}{2} \implies \hat{\lambda} > \frac{1-c}{2(e-1)}. \tag{121}$$

The above two conditions can be evaluated and simplified a bit more to get:

$$p > \frac{1+r}{3.7} \text{ and } r < 0.85 \text{ or } \hat{\lambda} > \frac{1-c}{3.4}, \tag{122}$$

and under these conditions, $\varepsilon_1 < 0.02$ and $\varepsilon_2 < 0.02$. Combining all this, Equation (118) can be rewritten as (while also dropping the $\lim_{n\to\infty}$ notation):

$$\alpha = \frac{p(1+\zeta)}{\hat{\lambda} + \frac{1-c}{4}} \text{ and } \hat{\alpha} = \frac{(1-p)(1+\zeta)}{\hat{\lambda} + \frac{1-c}{4}}, \tag{123}$$

where $\zeta \in (0, 0.04)$. Next, recall that we want $\hat{\lambda}\alpha < \frac{1}{2}$ and $\hat{\lambda}\hat{\alpha} < \frac{1}{2}$. Since, $\hat{\alpha} > \alpha$, both these conditions can be satisfied by just ensuring $\hat{\lambda}\hat{\alpha} < \frac{1}{2}$ which itself can be ensured by imposing:

$$\frac{1.04\hat{\lambda}(1-p)}{\hat{\lambda} + \frac{1-c}{4}} = \frac{1.04(1-p)}{1+r} < \frac{1}{2}. \tag{124}$$

The above is obtained by making use of Equation (123) and the fact that $\zeta < 0.04$. This gives us:

$$p > 1 - \Big(\frac{1+r}{2.08}\Big). \tag{125}$$

But again, we must have $p < \frac{1}{2}$ due to which we should also have $1 - \Big(\frac{1+r}{2.08}\Big) < \frac{1}{2}$; this holds when:

$$r = \frac{(1-c)/4}{\hat{\lambda}} > 0.04 \implies \hat{\lambda} < \frac{1-c}{0.16}. \tag{126}$$

So to recap, for the teacher, we have:

$$\alpha = \frac{p(1+\zeta)}{\hat{\lambda} + \frac{1-c}{4}} \text{ and } \hat{\alpha} = \frac{(1-p)(1+\zeta)}{\hat{\lambda} + \frac{1-c}{4}}, \tag{127}$$

where $\zeta \in (0, 0.04)$, with $\hat{\lambda}\alpha < \hat{\lambda}\hat{\alpha} < \frac{1}{2}$ for $p > \max\Big(1 - \Big(\frac{1+r}{2.08}\Big), \frac{1+r}{3.7}\Big)$. All this is valid when $r \in \big(0.04, 0.85\big)$ or equivalently when $\hat{\lambda} \in \Big(\frac{1-c}{3.4}, \frac{1-c}{0.16}\Big)$.

Let us do a sanity check to verify that the above range of p ensures $\hat{\lambda}\alpha < \hat{\lambda}\hat{\alpha} < \frac{1}{2}$. First, we shall show that $\zeta = \varepsilon_1 + \varepsilon_2 \geq 0$ by contradiction; so suppose $\zeta < 0$. Then using Equation (123), we have $\hat{\lambda}\hat{\alpha} = \frac{(1+\zeta)(1-p)}{1+r} < \frac{1.04(1-p)}{1+r} < \frac{1}{2}$, where the last step follows because $p > 1 - \big(\frac{1+r}{2.08}\big)$. But if $\hat{\lambda}\hat{\alpha} < \frac{1}{2}$, we must have $\varepsilon_1 > 0$ (using Equation (113)) as we are solving $\hat{\lambda}\hat{\alpha} = \sigma\big(cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\hat{\alpha}\big)$. Similarly, we must also have $\varepsilon_2 > 0$ as $\hat{\lambda}\alpha$ is also $< \frac{1}{2}$ (which is easy to see because $0 < \alpha < \hat{\alpha}$ since $p < \frac{1}{2}$). But then $\zeta = \varepsilon_1 + \varepsilon_2 > 0$, which is a contradiction to our earlier supposition of $\zeta < 0$. Hence, we must have $\zeta \geq 0$. But then using Equation (123), we have $\hat{\lambda}\alpha = \frac{(1+\zeta)p}{1+r} > \frac{p}{1+r} > \sigma(-1)$, where the last step follows because $p > \frac{1+r}{3.7}$. But if $\hat{\lambda}\alpha > \sigma(-1)$, we must have $\varepsilon_2 < 0.02$ (using Equation (114)) as we are solving $\hat{\lambda}\alpha = \sigma\big(-cn\big(\alpha - (\alpha+\hat{\alpha})p\big) - (1-c)\alpha\big)$. Similarly, we must also have $\varepsilon_1 < 0.02$ as $\hat{\lambda}\hat{\alpha}$ is also $> \sigma(-1)$ (again, because $\alpha < \hat{\alpha}$). Combining all this, we get $\zeta = \varepsilon_1 + \varepsilon_2 < 0.04$. So, $\hat{\lambda}\alpha < \hat{\lambda}\hat{\alpha} = \frac{(1+\zeta)(1-p)}{1+r} < \frac{1.04(1-p)}{1+r} < \frac{1}{2}$, where the last step follows because $p > 1 - \big(\frac{1+r}{2.08}\big)$. So our prescribed range of p indeed ensures $\hat{\lambda}\alpha < \hat{\lambda}\hat{\alpha} < \frac{1}{2}$.

Let us now move onto the **student**. Rewriting Equation (93) and Equation (94) while using the Maclaurin series expansion of the sigmoid function (from Equation (112)) and the fact that $\sigma(-z) = 1 - \sigma(z)\ \forall\ z \in \mathbb{R}$, we get:

$$\hat{\lambda}\hat{\alpha} + \hat{\lambda}\hat{\beta} = \sigma\Big(cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\hat{\beta}\Big) = \frac{1}{2} + \left(\frac{cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\hat{\beta}}{4}\right) + \varepsilon_3, \tag{128}$$

and

$$\hat{\lambda}\alpha + \hat{\lambda}\beta = \sigma\Big(-cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\beta\Big) = \frac{1}{2} - \left(\frac{cn\big(\beta - (\beta+\hat{\beta})p\big) + (1-c)\beta}{4}\right) + \varepsilon_4, \tag{129}$$

for some real numbers ε_3 and ε_4. Solving the above two equations in the limit of $n \to \infty$ (when $c = \Theta(1)$ and $\hat{\lambda} < \mathcal{O}(n)$) while using the values of α and $\hat{\alpha}$ from Equation (127), we get:

$$\lim_{n\to\infty} \beta = \frac{p}{\hat{\lambda} + \frac{1-c}{4}}\Big(-\frac{\hat{\lambda}(1+\zeta)}{\hat{\lambda} + \frac{1-c}{4}} + (1+\zeta')\Big) \text{ and } \lim_{n\to\infty} \hat{\beta} = \frac{1-p}{\hat{\lambda} + \frac{1-c}{4}}\Big(-\frac{\hat{\lambda}(1+\zeta)}{\hat{\lambda} + \frac{1-c}{4}} + (1+\zeta')\Big), \tag{130}$$

with $\zeta' := \varepsilon_3 + \varepsilon_4$. Again, we shall drop the $\lim_{n\to\infty}$ notation subsequently, and it is implied directly.

Next, we get:

$$\alpha + \beta = \frac{p}{\hat{\lambda} + \frac{1-c}{4}}\left(\frac{(\frac{1-c}{4})(1+\zeta)}{\hat{\lambda} + \frac{1-c}{4}} + (1+\zeta')\right), \tag{131}$$

and

$$\hat{\alpha} + \hat{\beta} = \frac{1-p}{\hat{\lambda} + \frac{1-c}{4}}\left(\frac{(\frac{1-c}{4})(1+\zeta)}{\hat{\lambda} + \frac{1-c}{4}} + (1+\zeta')\right). \tag{132}$$

Now, recall that if $\hat{\lambda}(\hat{\alpha}+\hat{\beta}) > \frac{1}{2}$ and $\hat{\lambda}(\alpha+\beta) < \frac{1}{2}$, then the student has managed to correctly classify all the points in the training set. Let us first impose $\hat{\lambda}(\hat{\alpha}+\hat{\beta}) \in \big(\frac{1}{2}, \sigma(1)\big)$. Then, since we are solving Equation (128), $\sigma\Big(cn\big(\beta - (\beta + \hat{\beta})p\big) - (1-c)\hat{\beta}\Big) \in \big(\frac{1}{2}, \sigma(1)\big)$, and so $\varepsilon_3 \in (-0.02, 0)$ using Equation (115). Now, we shall be imposing $\hat{\lambda}(\alpha+\beta) < \frac{1}{2}$. Additionally, we ensured earlier that $\hat{\lambda}\alpha > \sigma(-1)$ and showed in Lemma I.2 that $\beta \geq 0$. Therefore, we will have $\hat{\lambda}(\alpha+\beta) \in \big(\sigma(-1), \frac{1}{2}\big)$. Since we are solving Equation (129), $\sigma\Big(-cn\big(\beta - (\beta+\hat{\beta})p\big) - (1-c)\beta\Big) \in \big(\sigma(-1), \frac{1}{2}\big)$, due to which $\varepsilon_4 \in (0, 0.02)$ using Equation (114). Thus, $\zeta' = \varepsilon_3 + \varepsilon_4 \in (-0.02, 0.02)$.

Now, using Equation (131) and Equation (132), and plugging in $r = \frac{(1-c)/4}{\hat{\lambda}}$, we get:

$$\hat{\lambda}(\alpha+\beta) = \frac{p}{1+r}\Big(\frac{r(1+\zeta)}{1+r} + (1+\zeta')\Big), \tag{133}$$

and

$$\hat{\lambda}(\hat{\alpha}+\hat{\beta}) = \frac{1-p}{1+r}\Big(\frac{r(1+\zeta)}{1+r} + (1+\zeta')\Big), \tag{134}$$

with $\zeta \in (0, 0.04)$ and $\zeta' \in (-0.02, 0.02)$. Let us first ensure $\hat{\lambda}(\hat{\alpha}+\hat{\beta}) \in \big(\frac{1}{2}, \sigma(1)\big)$. Using the bounds on ζ and ζ', this can be ensured by having:

$$\frac{1-p}{1+r}\Big(\frac{1.04r}{1+r} + 1.02\Big) < \sigma(1) = \frac{e}{1+e}, \tag{135}$$

and

$$\frac{1-p}{1+r}\Big(\frac{r}{1+r} + 0.98\Big) > \frac{1}{2}. \tag{136}$$

Solving and simplifying the above two equations gives us:

$$p \in \Big(1 - \frac{0.7(1+r)^2}{1+2r}, 1 - \frac{0.51(1+r)^2}{1+2r}\Big). \tag{137}$$

Note that:

$$1 - \frac{0.7(1+r)^2}{1+2r} < 1 - \frac{0.51(1+r)^2}{1+2r} < \frac{1}{2} \tag{138}$$

for all $r > 0$, and so we are good here. But recall that from the teacher's analysis (see the discussion after Equation (127)), we had $p > \max\Big(1 - \big(\frac{1+r}{2.08}\big), \frac{1+r}{3.7}\Big)$. Combining everything, our current bound on p is:

$$p \in \left(\max\Big(1 - \Big(\frac{1+r}{2.08}\Big), \frac{1+r}{3.7}, 1 - \frac{0.7(1+r)^2}{1+2r}\Big), 1 - \frac{0.51(1+r)^2}{1+2r}\right). \tag{139}$$

But the above is only meaningful when the lower bound on p is smaller than the upper bound on it. So we must find the range of r for which:

$$1 - \Big(\frac{1+r}{2.08}\Big) < 1 - \frac{0.51(1+r)^2}{1+2r} \text{ and } \frac{1+r}{3.7} < 1 - \frac{0.51(1+r)^2}{1+2r}.$$

$1 - \frac{0.7(1+r)^2}{1+2r}$ is trivially smaller than $1 - \frac{0.51(1+r)^2}{1+2r}$ so we do not need to worry about that. Combining the range of r obtained from the above equation with the previous range of $r \in (0.04, 0.85)$ (that we obtained from the teacher), we get:

$$r \in [0.07, 0.54] \implies \hat{\lambda} \in \Big[\frac{1-c}{2.16}, \frac{1-c}{0.28}\Big]. \tag{140}$$

Finally, we need to ensure $\hat{\lambda}(\alpha+\beta) < \frac{1}{2}$. Using Equation (133) and the bounds on ζ and ζ', this can be ensured by imposing:

$$\frac{p}{1+r}\Big(\frac{1.04r}{1+r} + 1.02\Big) < \frac{1}{2}. \tag{141}$$

This can be simplified to:

$$p < \frac{0.485(1+r)^2}{1+2r}.$$

But recall that we already have an upper bound on p of $1 - \frac{0.51(1+r)^2}{1+2r}$. It can be checked that $1 - \frac{0.51(1+r)^2}{1+2r} < \frac{0.485(1+r)^2}{1+2r}$ for $r \geq 0.08$. Thus, for $r \in [0.08, 0.54]$ or $\hat{\lambda} \in \left[\frac{1-c}{2.16}, \frac{1-c}{0.32}\right]$, our bound on p remains the same as Equation (139), i.e.,

$$p \in \left(\max\left(1 - \left(\frac{1+r}{2.08}\right), \frac{1+r}{3.7}, 1 - \frac{0.7(1+r)^2}{1+2r}\right), 1 - \frac{0.51(1+r)^2}{1+2r} \right). \tag{142}$$

Finally, to simplify our bound on p a bit, we consider $r \in [0.10, 0.54]$, where:

$$\max\left(1 - \left(\frac{1+r}{2.08}\right), \frac{1+r}{3.7}, 1 - \frac{0.7(1+r)^2}{1+2r}\right) = \max\left(1 - \left(\frac{1+r}{2.08}\right), \frac{1+r}{3.7}\right). \tag{143}$$

Thus, our final bound on p is:

$$p \in \left(\max\left(1 - \left(\frac{1+r}{2.08}\right), \frac{1+r}{3.7}\right), 1 - \frac{0.51(1+r)^2}{1+2r} \right), \tag{144}$$

for

$$r \in [0.10, 0.54] \text{ or } \hat{\lambda} \in \left[\frac{1-c}{2.16}, \frac{1-c}{0.40}\right]. \tag{145}$$

Finally, note that the prescribed range of $\hat{\lambda}$ is $< \mathcal{O}(n)$ (as required in Equation (118) and Equation (130)) since $c = \Theta(1)$. So we are good here.

Also, since $n \to \infty$, the generalization gap (i.e., population accuracy - training accuracy) $\to 0$; see for e.g., the margin bounds (with ℓ_2-regularization) in Kakade et al. (2008) where it is shown that the generalization gap goes down as $\mathcal{O}(1/\sqrt{n})$. Therefore, the population accuracy of the student (resp., teacher) is the same as the training accuracy of the student (resp., teacher).

This finishes the proof. ■

J. Variability of Predictions of Teacher and Student

Recall the setup of Section 4.

Corollary J.1 (Variability of predictions of points within the same class). *Define* $\Delta_{\mathrm{T}} := \max_{i \neq i', y_i = y_{i'}} |y_i^{(\mathrm{T})} - y_{i'}^{(\mathrm{T})}|$ *as the teacher's variability, i.e., the maximum difference between the teacher's predictions on two points having the same ground truth label. Similarly,* $\Delta_{\mathrm{S}} := \max_{i \neq i', y_i = y_{i'}} |y_i^{(\mathrm{S})} - y_{i'}^{(\mathrm{S})}|$ *is defined as the student's variability. Under the conditions of Theorem 4.3,* $\Delta_{\mathrm{S}} < \Delta_{\mathrm{T}}$.

In other words, the student's predictions are more homogeneous than the teacher's predictions as per Corollary J.1. This is analogous to **SD mitigating the variance due to label noise** in linear regression (Remark 3.3) leading to smaller variability. We corroborate Corollary J.1 with empirical evidence in Appendix K.2.

Now, we shall prove Corollary J.1.

Proof. From Equation (69), we have:

$$\Delta_{\mathrm{T}} = 1 - \hat{\lambda}(\alpha + \hat{\alpha}), \tag{146}$$

where $\hat{\lambda} = 2n\lambda$. Similarly, using Equation (95), we have:

$$\Delta_{\mathrm{S}} = 1 - \hat{\lambda}(\alpha + \beta + \hat{\alpha} + \hat{\beta}). \tag{147}$$

Next, using Equation (123) in Equation (146), we get:

$$\Delta_{\mathrm{T}} = 1 - \left(\frac{1+\zeta}{1+r}\right), \tag{148}$$

where $\zeta \in (0, 0.04)$ and $r = \frac{(1-c)}{4\hat{\lambda}}$. Similarly, using Equation (133) and Equation (134), we get:

$$\Delta_{\text{S}} = 1 - \frac{1}{1+r}\Big(\frac{r(1+\zeta)}{1+r} + (1+\zeta')\Big), \tag{149}$$

where $\zeta' \in (-0.02, 0.02)$. Rewriting Equation (149) slightly, we get:

$$\begin{aligned}
\Delta_{\text{S}} &= 1 - \Big(\frac{1+\zeta}{1+r}\Big)\Big(\frac{r}{1+r} + \frac{1+\zeta'}{1+\zeta}\Big) && (150)\\
&\leq 1 - \Big(\frac{1+\zeta}{1+r}\Big)\Big(\frac{0.1}{1.1} + \frac{0.98}{1.04}\Big) && (151)\\
&< 1 - \Big(\frac{1+\zeta}{1+r}\Big) && (152)\\
&= \Delta_{\text{T}}. && (153)
\end{aligned}$$

In Equation (151), we have used the fact that $r \geq 0.1$ (from the condition of Theorem 4.3), $\zeta' \geq -0.02$ and $\zeta \leq 0.04$. ■

K. More Empirical Results

K.1. Verifying Remark 3.6 Continued

Here, we shall show that $\xi > 1$ can be suboptimal and detrimental *when there is no label noise*; specifically, on CIFAR-100 and Food-101 with ResNet-34. All other details are the same as (i) in Section 5. In Table 6, we list the student's improvement over the teacher (i.e., student's test accuracy - teacher's test accuracy) averaged across 3 different runs with different values of ξ for CIFAR-100 and Food-101 without any label noise. In both the cases, note that the best performing value of ξ is less than 1 and $\xi > 1$ harms performance.

Table 6. **No Corruption:** Average ($\pm$ 1 std.) improvement of student over teacher (i.e., student's test set accuracy - teacher's test set accuracy) with different values of the imitation parameter ξ. Unlike Tables 1 and 2, note that the value of ξ yielding the biggest improvement here is *less* than 1. In fact, $\xi > 1$ is significantly detrimental here.

ξ	Improvement of student over teacher (i.e., $\xi = 0$)
0.2	$\mathbf{0.22 \pm 0.03}$ **%**
0.5	$\mathbf{0.19 \pm 0.01}$ **%**
0.7	0.03 ± 0.09 %
1.0	-0.20 ± 0.04 %
1.2	-0.43 ± 0.06 %
1.5	-0.96 ± 0.08 %
1.7	-1.33 ± 0.05 %

(a) CIFAR-100 with ResNet-34

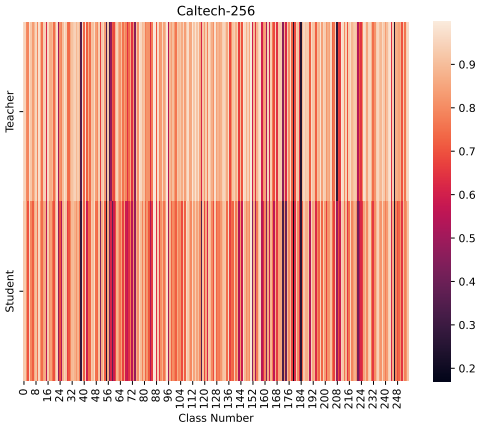
ξ	Improvement of student over teacher (i.e., $\xi = 0$)
0.2	$\mathbf{0.24 \pm 0.08}$ **%**
0.5	0.12 ± 0.09 %
0.7	-0.04 ± 0.05 %
1.0	-0.43 ± 0.05 %
1.2	-0.80 ± 0.06 %
1.5	-1.39 ± 0.07 %
1.7	-1.86 ± 0.04 %

(b) Food-101 with ResNet-34

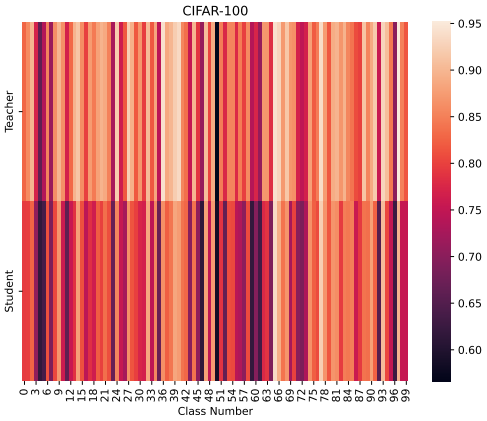
The individual accuracies of the teacher and student can be found in Appendix L.

K.2. Verifying Corollary J.1

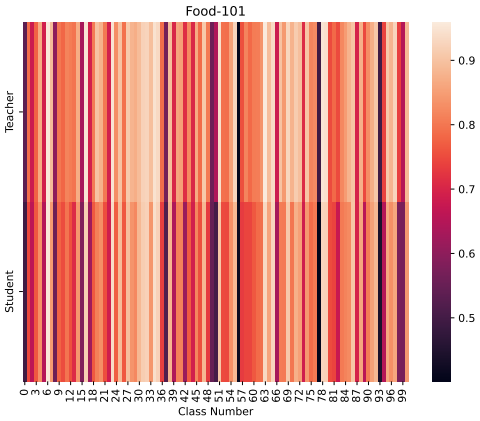
We now provide empirical evidence for our claim of the student's predictions being more homogeneous than the teacher's predictions in Corollary J.1. Since our experiments are for the multi-class (and not binary) case, we look at a slightly different metric to quantify variability which we introduce next. For a sample $\boldsymbol{x}$ belonging to class $c(\boldsymbol{x})$, let $\hat{p}_{\text{T}}(\boldsymbol{x})$ and $\hat{p}_{\text{S}}(\boldsymbol{x})$ be the teacher's and student's predicted probability of $\boldsymbol{x}$ belonging to $c(\boldsymbol{x})$, respectively. Also, let $\mathcal{X}'_c$ be the set of samples in the test set belonging to class c. To quantify the variability of the teacher and student for class c, we look at $\max_{\boldsymbol{x}_1,\boldsymbol{x}_2 \in \mathcal{X}'_c} |\hat{p}_{\text{T}}(\boldsymbol{x}_1) - \hat{p}_{\text{T}}(\boldsymbol{x}_2)|$ and $\max_{\boldsymbol{x}_1,\boldsymbol{x}_2 \in \mathcal{X}'_c} |\hat{p}_{\text{S}}(\boldsymbol{x}_1) - \hat{p}_{\text{S}}(\boldsymbol{x}_2)|$, i.e., the range of $\hat{p}_{\text{T}}(\boldsymbol{x})$ and $\hat{p}_{\text{S}}(\boldsymbol{x})$ w.r.t. $\boldsymbol{x} \in \mathcal{X}'_c$, respectively. In Figure 3, we plot the per-class variability as defined here for three of the cases of Table 3 covering all three types of label corruption; please see the caption for discussion.



(a) 50% Random Corruption in Caltech-256



(b) 50% Hierarchical Corruption in CIFAR-100



(c) 50% Adversarial Corruption in Food-101

Figure 3. **ResNet-34 with $\xi = 1$:** Comparison of the per-class variability of the teacher and student (i.e., range of the teacher's and student's predictions of belonging to the correct class, as defined in Appendix K.2) as a heat map. Note that a darker shade corresponds to a lower value; in all the cases, the student's heat map has a darker shade than the teacher's heat map which means that the student has a smaller variability than the teacher. This is consistent with the claim in Corollary J.1.

K.3. Results with Other Weight Decay Values

All our previous results were with weight decay $= 5 \times 10^{-4}$. Here, we verify Remarks 3.6 and 3.7 for two other weight decay values which are 1×10^{-3} and 1×10^{-4}.

(i) Verifying Remark 3.6: In Table 7, we list the student's improvement over the teacher (i.e., student's test accuracy - teacher's test accuracy) averaged across 3 different runs for different values of ξ in the case of (a) Caltech-256 with 50% random corruption & weight decay $= 1 \times 10^{-4}$ and (b) CIFAR-100 with 50% hierarchical corruption & weight decay $= 1 \times 10^{-3}$. As was the case with weight decay $= 5 \times 10^{-4}$ in Tables 1 and 2, note that the value of ξ yielding the biggest improvement here is also > 1.

(ii)Verifying Remark 3.7: The setup is the same as (ii) in Section 5, i.e., the student is trained with $\xi = 1$. In Table 8, we show the student's improvement over the teacher averaged across 3 different runs for varying degrees of label corruption in the case of (a) Caltech-256 with random corruption & weight decay $= 1 \times 10^{-4}$ and (b) CIFAR-100 with hierarchical corruption & weight decay $= 1 \times 10^{-3}$. As was the case with weight decay $= 5 \times 10^{-4}$ in Table 3, note that the improvement of the student (trained with $\xi = 1$) over the teacher increases as the corruption level increases.

Table 7. Average ($\pm$ 1 std.) improvement of student over teacher (i.e., student's test set accuracy - teacher's test set accuracy) with different values of ξ. Just like with weight decay $= 5 \times 10^{-4}$ (Tables 1 and 2), note that the value of ξ yielding the biggest improvement with both weight decay values here is more than 1. This is consistent with our message in Remark 3.6.

ξ	Improvement of student over teacher
0.2	2.04 ± 0.16 %
0.5	5.05 ± 0.10 %
0.7	6.82 ± 0.16 %
1.0	9.07 ± 0.17 %
1.2	10.43 ± 0.15 %
1.5	11.78 ± 0.16 %
1.7	12.30 ± 0.19 %
2.0	$\mathbf{13.07 \pm 0.20}$ %
2.2	$\mathbf{12.89 \pm 0.43}$ %
2.5	11.74 ± 0.60 %

(a) **ResNet-34:** 50% Random Corruption in Caltech-256 with weight decay $= 1 \times 10^{-4}$

ξ	Improvement of student over teacher
0.2	0.39 ± 0.09 %
0.5	1.90 ± 0.08 %
0.7	2.80 ± 0.09 %
1.0	3.82 ± 0.04 %
1.2	4.17 ± 0.05 %
1.5	$\mathbf{4.51 \pm 0.07}$ %
1.7	$\mathbf{4.56 \pm 0.02}$ %
2.0	4.15 ± 0.07 %

(b) **ResNet-34:** 50% Hierarchical Corruption in CIFAR-100 with weight decay $= 1 \times 10^{-3}$

Table 8. **ResNet-34 with $\xi = 1$:** Average ($\pm$ 1 std.) improvement of student over teacher (i.e., student's test set accuracy - teacher's test set accuracy) with varying levels of label corruption. Just like with weight decay $= 5 \times 10^{-4}$ (Table 3), note that the improvement of the student over the teacher increases as the corruption level increases. This is consistent with our claim in Remark 3.7.

Corruption level	Improvement of student over teacher
0%	0.25 ± 0.04 %
10%	1.39 ± 0.10 %
30%	6.31 ± 0.11 %
50%	9.07 ± 0.17 %

(a) Random Corruption in Caltech-256 with weight decay $= 1 \times 10^{-4}$

Corruption level	Improvement of student over teacher
0%	-0.73 ± 0.09 %
10%	0.03 ± 0.10 %
30%	1.77 ± 0.19 %
50%	3.82 ± 0.04 %

(b) Hierarchical Corruption in CIFAR-100 with weight decay $= 1 \times 10^{-3}$

The individual accuracies of the teacher and student and the experimental details appear in Appendix L.

L. Detailed Empirical Results

We list the individual accuracies of the teacher and student (along with the student's improvement) corresponding to the results of Table 1 in Tables 9-14, Table 2 in Tables 15-16, Table 3 in Tables 17-22, Table 6 in Tables 23-24, Table 7 in Tables 25-26 and Table 8 in Tables 27-28.

Experimental details: In all the cases, we use SGD with momentum = 0.9 and batch size = 128 for training. Since we are training only the softmax layer (i.e., doing logistic regression), we use an exponentially decaying learning rate scheme with decay parameter = 0.98 (for every epoch) and the initial learning rate is tuned[16] over $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5\}$. The maximum number of epochs is 200.

Table 9. Detailed Version of Table 1a (50% Random Corruption in Caltech-256 with ResNet-34)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	57.61 ± 0.03 %	0 %
0.2	59.83 ± 0.12 %	2.22 ± 0.12 %
0.5	62.79 ± 0.04 %	5.18 ± 0.03 %
0.7	64.45 ± 0.09 %	6.84 ± 0.06 %
1.0	66.15 ± 0.27 %	8.54 ± 0.29 %
1.2	67.27 ± 0.25 %	9.66 ± 0.23 %
1.5	$\mathbf{67.65 \pm 0.54}$ %	$\mathbf{10.04 \pm 0.51}$ %
1.7	$\mathbf{67.42 \pm 0.58}$ %	$\mathbf{9.81 \pm 0.55}$ %
2.0	66.17 ± 0.77 %	8.56 ± 0.73 %

Table 10. Detailed Version of Table 1b (50% Random Corruption in Caltech-256 with VGG-16)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	61.15 ± 0.09 %	0 %
0.5	62.04 ± 0.02 %	0.89 ± 0.10 %
1.0	63.16 ± 0.06 %	2.01 ± 0.14 %
1.5	64.28 ± 0.06 %	3.13 ± 0.11 %
2.0	65.37 ± 0.12 %	4.22 ± 0.20 %
2.5	66.43 ± 0.05 %	5.28 ± 0.13 %
3.0	$\mathbf{66.93 \pm 0.03}$ %	$\mathbf{5.78 \pm 0.12}$ %
3.5	$\mathbf{67.01 \pm 0.13}$ %	$\mathbf{5.86 \pm 0.18}$ %
4.0	66.47 ± 0.25 %	5.32 ± 0.33 %

[16]The tuning is done by picking the learning rate which yields the lowest training loss with the observed (noisy) labels. This is consistent with our theory setup where we assume convergence to the optimum of the training loss w.r.t. the observed labels.

Table 11. Detailed Version of Table 1c (50% Hierarchical Corruption in CIFAR-100 with ResNet-34)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	50.80 ± 0.04 %	0 %
0.2	51.78 ± 0.14 %	0.98 ± 0.12 %
0.5	53.26 ± 0.14 %	2.46 ± 0.11 %
0.7	54.18 ± 0.03 %	3.38 ± 0.02 %
1.0	54.99 ± 0.08 %	4.19 ± 0.09 %
1.2	$\mathbf{55.26 \pm 0.18}$ %	$\mathbf{4.46 \pm 0.19}$ %
1.5	$\mathbf{55.26 \pm 0.15}$ %	$\mathbf{4.46 \pm 0.17}$ %
1.7	$\mathbf{55.12 \pm 0.16}$ %	$\mathbf{4.32 \pm 0.18}$ %
2.0	54.32 ± 0.20 %	3.52 ± 0.23 %

Table 12. Detailed Version of Table 1d (50% Hierarchical Corruption in CIFAR-100 with VGG-16)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	41.60 ± 0.08 %	0 %
0.2	42.70 ± 0.03 %	1.10 ± 0.09 %
0.5	44.29 ± 0.06 %	2.69 ± 0.02 %
0.7	45.32 ± 0.05 %	3.72 ± 0.05 %
1.0	46.89 ± 0.05 %	5.29 ± 0.11 %
1.2	47.86 ± 0.06 %	6.26 ± 0.09 %
1.5	$\mathbf{48.80 \pm 0.16}$ %	$\mathbf{7.20 \pm 0.14}$ %
1.7	$\mathbf{48.83 \pm 0.18}$ %	$\mathbf{7.23 \pm 0.17}$ %
2.0	48.02 ± 0.25 %	6.42 ± 0.26 %

Table 13. Detailed Version of Table 1e (50% Adversarial Corruption in Food-101 with ResNet-34)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	48.93 ± 0.08 %	0 %
0.2	49.06 ± 0.05 %	0.13 ± 0.08 %
0.5	49.90 ± 0.04 %	0.97 ± 0.04 %
0.7	50.38 ± 0.09 %	1.45 ± 0.01 %
1.0	$\mathbf{50.78 \pm 0.07}$ %	$\mathbf{1.85 \pm 0.09}$ %
1.2	$\mathbf{50.80 \pm 0.06}$ %	$\mathbf{1.87 \pm 0.06}$ %
1.5	$\mathbf{50.79 \pm 0.03}$ %	$\mathbf{1.86 \pm 0.08}$ %
1.7	$\mathbf{50.73 \pm 0.04}$ %	$\mathbf{1.80 \pm 0.05}$ %
2.0	50.46 ± 0.09 %	1.53 ± 0.02 %

Table 14. Detailed Version of Table 1f (50% Adversarial Corruption in Food-101 with VGG-16)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	37.01 ± 0.46 %	0 %
0.2	37.80 ± 0.23 %	0.79 ± 0.23 %
0.5	39.15 ± 0.37 %	2.14 ± 0.09 %
0.7	39.97 ± 0.42 %	2.96 ± 0.04 %
1.0	40.86 ± 0.51 %	3.85 ± 0.05 %
1.2	$\mathbf{41.23 \pm 0.60}$ %	$\mathbf{4.22 \pm 0.15}$ %
1.5	$\mathbf{41.40 \pm 0.71}$ %	$\mathbf{4.39 \pm 0.29}$ %
1.7	$\mathbf{41.21 \pm 0.76}$ %	$\mathbf{4.20 \pm 0.34}$ %
2.0	40.54 ± 0.90 %	3.53 ± 0.49 %

Table 15. Detailed Version of Table 2a (30% Random Corruption in Stanford Cars with ResNet-34)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	25.01 ± 0.20 %	0 %
0.2	25.90 ± 0.09 %	0.89 ± 0.15 %
0.5	27.16 ± 0.16 %	2.15 ± 0.06 %
0.7	27.76 ± 0.21 %	2.75 ± 0.10 %
1.0	28.33 ± 0.14 %	3.32 ± 0.11 %
1.2	$\mathbf{28.54 \pm 0.11}$ %	$\mathbf{3.53 \pm 0.16}$ %
1.5	$\mathbf{28.47 \pm 0.10}$ %	$\mathbf{3.46 \pm 0.12}$ %
1.7	27.97 ± 0.20 %	2.96 ± 0.24 %
2.0	26.80 ± 0.27 %	1.79 ± 0.29 %

Table 16. Detailed Version of Table 2b (30% Adversarial Corruption in Flowers-102 with ResNet-34)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	50.34 ± 0.23 %	0 %
0.5	50.22 ± 0.23 %	-0.12 ± 0.20 %
1.0	50.88 ± 0.24 %	0.54 ± 0.02 %
1.5	51.20 ± 0.22 %	0.86 ± 0.01 %
2.0	51.91 ± 0.41 %	1.57 ± 0.34 %
2.5	52.39 ± 0.44 %	2.05 ± 0.27 %
3.0	52.83 ± 0.28 %	2.49 ± 0.25 %
3.5	52.96 ± 0.28 %	2.62 ± 0.12 %
4.0	53.21 ± 0.31 %	2.87 ± 0.09 %
4.5	$\mathbf{53.35 \pm 0.15}$ %	$\mathbf{3.01 \pm 0.22}$ %
5.0	$\mathbf{53.55 \pm 0.25}$ %	$\mathbf{3.21 \pm 0.07}$ %
5.5	$\mathbf{53.28 \pm 0.50}$ %	$\mathbf{2.94 \pm 0.33}$ %
6.0	$\mathbf{53.40 \pm 0.28}$ %	$\mathbf{3.06 \pm 0.09}$ %

Table 17. Detailed Version of Random Corruption in Caltech-256 with ResNet-34 and $\xi = 1$ (Table 3a)

Corruption level	Teacher's test acc.	Student's test acc.	Improvement of student over teacher
0%	83.97 ± 0.10 %	83.93 ± 0.12 %	-0.04 ± 0.02 %
10%	77.86 ± 0.14 %	80.37 ± 0.04 %	2.51 ± 0.11 %
30%	68.09 ± 0.21 %	74.23 ± 0.08 %	6.14 ± 0.16 %
50%	57.61 ± 0.03 %	66.15 ± 0.27 %	8.54 ± 0.29 %

Table 18. Detailed Version of Adversarial Corruption in Caltech-256 with ResNet-34 and $\xi = 1$ (Table 3a)

Corruption level	Teacher's test acc.	Student's test acc.	Improvement of student over teacher
0%	83.97 ± 0.10 %	83.93 ± 0.12 %	-0.04 ± 0.02 %
10%	77.01 ± 0.23 %	79.33 ± 0.13 %	2.32 ± 0.10 %
30%	64.21 ± 0.36 %	69.29 ± 0.12 %	5.08 ± 0.25 %
50%	48.66 ± 0.10 %	54.43 ± 0.29 %	5.77 ± 0.19 %

Table 19. Detailed Version of Random Corruption in CIFAR-100 with ResNet-34 and $\xi = 1$ (Table 3b)

Corruption level	Teacher's test acc.	Student's test acc.	Improvement of student over teacher
0%	72.77 ± 0.07 %	72.54 ± 0.07 %	-0.23 ± 0.06 %
10%	70.57 ± 0.14 %	71.20 ± 0.02 %	0.63 ± 0.11 %
30%	66.80 ± 0.06 %	68.14 ± 0.07 %	1.34 ± 0.13 %
50%	62.47 ± 0.10 %	64.58 ± 0.10 %	2.11 ± 0.15 %

Table 20. Detailed Version of Hierarchical Corruption in CIFAR-100 with ResNet-34 and $\xi = 1$ (Table 3b)

Corruption level	Teacher's test acc.	Student's test acc.	Improvement of student over teacher
0%	72.77 ± 0.07 %	72.54 ± 0.07 %	-0.23 ± 0.06 %
10%	69.39 ± 0.09 %	70.58 ± 0.08 %	1.19 ± 0.08 %
30%	62.18 ± 0.12 %	64.98 ± 0.10 %	2.80 ± 0.06 %
50%	50.80 ± 0.04 %	54.99 ± 0.08 %	4.19 ± 0.09 %

Table 21. Detailed Version of Random Corruption in Food-101 with ResNet-34 and $\xi = 1$ (Table 3c)

Corruption level	Teacher's test acc.	Student's test acc.	Improvement of student over teacher
0%	63.65 ± 0.08 %	63.28 ± 0.04 %	-0.37 ± 0.10 %
10%	62.44 ± 0.03 %	62.54 ± 0.03 %	0.10 ± 0.04 %
30%	59.38 ± 0.20 %	59.85 ± 0.18 %	0.47 ± 0.04 %
50%	54.76 ± 0.13 %	55.88 ± 0.06 %	1.12 ± 0.08 %

Table 22. Detailed Version of Adversarial Corruption in Food-101 with ResNet-34 and $\xi = 1$ (Table 3c)

Corruption level	Teacher's test acc.	Student's test acc.	Improvement of student over teacher
0%	63.65 ± 0.08 %	63.28 ± 0.04 %	-0.37 ± 0.10 %
10%	61.92 ± 0.13 %	62.16 ± 0.11 %	0.25 ± 0.05 %
30%	57.03 ± 0.16 %	57.80 ± 0.22 %	0.77 ± 0.06 %
50%	48.93 ± 0.08 %	50.78 ± 0.07 %	1.85 ± 0.09 %

Table 23. Detailed Version of Table 6a (No Corruption in CIFAR-100 with ResNet-34)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	72.73 ± 0.04 %	0 %
0.2	$\mathbf{72.95 \pm 0.01}$ %	$\mathbf{0.22 \pm 0.03}$ %
0.5	$\mathbf{72.92 \pm 0.05}$ %	$\mathbf{0.19 \pm 0.01}$ %
0.7	72.76 ± 0.05 %	0.03 ± 0.09 %
1.0	72.53 ± 0.02 %	-0.20 ± 0.04 %
1.2	72.30 ± 0.06 %	-0.43 ± 0.06 %
1.5	71.77 ± 0.07 %	-0.96 ± 0.08 %
1.7	71.40 ± 0.05 %	-1.33 ± 0.05 %

Table 24. Detailed Version of Table 6b (No Corruption in Food-101 with ResNet-34)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	63.62 ± 0.10 %	0 %
0.2	$\mathbf{63.86 \pm 0.02}$ %	$\mathbf{0.24 \pm 0.08}$ %
0.5	63.74 ± 0.01 %	0.12 ± 0.09 %
0.7	63.58 ± 0.07 %	-0.04 ± 0.05 %
1.0	63.19 ± 0.13 %	-0.43 ± 0.05 %
1.2	62.82 ± 0.14 %	-0.80 ± 0.06 %
1.5	62.23 ± 0.08 %	-1.39 ± 0.07 %
1.7	61.76 ± 0.13 %	-1.86 ± 0.04 %

Table 25. Detailed Version of Table 7a (50% Random Corruption in Caltech-256 w/ ResNet-34 and wt. decay $= 1 \times 10^{-4}$)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	39.78 ± 0.18 %	0 %
0.2	41.82 ± 0.06 %	2.04 ± 0.16 %
0.5	44.83 ± 0.11 %	5.05 ± 0.10 %
0.7	46.60 ± 0.09 %	6.82 ± 0.16 %
1.0	48.85 ± 0.06 %	9.07 ± 0.17 %
1.2	50.21 ± 0.03 %	10.43 ± 0.15 %
1.5	51.56 ± 0.10 %	11.78 ± 0.16 %
1.7	52.08 ± 0.03 %	12.30 ± 0.19 %
2.0	$\mathbf{52.85 \pm 0.05}$ %	$\mathbf{13.07 \pm 0.20}$ %
2.2	$\mathbf{52.67 \pm 0.33}$ %	$\mathbf{12.89 \pm 0.43}$ %
2.5	51.52 ± 0.51 %	11.74 ± 0.60 %

Table 26. Detailed Version of Table 7b (50% Hierarchical Corruption in CIFAR-100 w/ ResNet-34 and wt. decay $= 1 \times 10^{-3}$)

ξ	Student's test acc.	Improvement of student over teacher
0.0 (=Teacher)	54.71 ± 0.05 %	0 %
0.2	55.10 ± 0.04 %	0.39 ± 0.09 %
0.5	56.61 ± 0.04 %	1.90 ± 0.08 %
0.7	57.51 ± 0.04 %	2.80 ± 0.09 %
1.0	58.53 ± 0.05 %	3.82 ± 0.04 %
1.2	58.88 ± 0.03 %	4.17 ± 0.05 %
1.5	$\mathbf{59.22 \pm 0.11}$ %	$\mathbf{4.51 \pm 0.07}$ %
1.7	$\mathbf{59.27 \pm 0.06}$ %	$\mathbf{4.56 \pm 0.02}$ %
2.0	58.86 ± 0.02 %	4.15 ± 0.07 %

Table 27. Detailed Version of Table 8a (Random Corruption in Caltech-256 w/ ResNet-34, $\xi = 1$ and wt. decay $= 1 \times 10^{-4}$)

Corruption level	Teacher's test acc.	Student's test acc.	Improvement of student over teacher
0%	82.95 ± 0.02 %	83.20 ± 0.04 %	0.25 ± 0.04 %
10%	74.29 ± 0.12 %	75.68 ± 0.03 %	1.39 ± 0.10 %
30%	54.65 ± 0.13 %	60.96 ± 0.18 %	6.31 ± 0.11 %
50%	39.78 ± 0.18 %	48.85 ± 0.06 %	9.07 ± 0.17 %

Table 28. Detailed Version of Table 8b (Hierarchical Corruption in CIFAR-100 w/ ResNet-34, $\xi = 1$ and wt. decay $= 1 \times 10^{-3}$)

Corruption level	Teacher's test acc.	Student's test acc.	Improvement of student over teacher
0%	72.99 ± 0.09 %	72.26 ± 0.01 %	-0.73 ± 0.09 %
10%	70.59 ± 0.04 %	70.62 ± 0.07 %	0.03 ± 0.10 %
30%	64.64 ± 0.12 %	66.41 ± 0.09 %	1.77 ± 0.19 %
50%	54.71 ± 0.05 %	58.53 ± 0.05 %	3.82 ± 0.04 %

M. Preliminary Empirical Results for Full Network Training

We provide some preliminary empirical evidence to show that the insights that we verified for linear probing can also hold for *full network training from scratch*; specifically, on Caltech-256 with random corruption trained with ResNet-34 from scratch. We assume that we have a small *correctly labeled* validation set which is $(1/24)$ times the size of the training set. This is because if we let the teacher train until convergence and do not stop early, it will completely memorize the noisy labels and its output $(y_T) \approx$ the hard label (y) in which case there will be no effect of SD (as $\xi y_T + (1-\xi)y \approx y$). Moreover, it will perform very poorly on the test set. Ideally, we can train the teacher with noise correction techniques to prevent overfitting, but since the goal of our proof-of-concept experiment is to show the abilities of SD, using a small clean validation set instead is reasonable.

The weight decay value is set to 5×10^{-3} here. We use SGD with momentum = 0.9 and batch size = 128. We train for a maximum of 60 epochs with a step decay learning rate scheme wherein the learning rate is decimated by $\frac{1}{5}$, $\frac{1}{25}$ and $\frac{1}{125}$ after 15, 30 and 45 epochs, respectively. The initial learning rate is tuned over $\{0.001, 0.005, 0.01, 0.05, 0.1\}$. We stop based on the performance observed on the clean validation set.

The results for full network training from scratch – analogous to Tables 1 and 3 for linear probing – are shown in Tables 29 and 30, respectively. The table captions discuss the results. In summary, the results here are consistent with the results in Tables 1 and 3 for linear probing. However, more extensive experimentation with full network training is required; this is left for future work.

Table 29. Average ($\pm$ 1 std.) improvement of student over teacher (i.e., student's test set accuracy - teacher's test set accuracy) for full network (viz., ResNet-34) training from scratch in the case of Caltech-256 with *50%* random corruption with different values of the imitation parameter ξ (analogous to Table 1). Note that $\xi = 1.2 > 1$ performs the best here. This is consistent with the results in Table 1 for linear probing.

ξ	Improvement of student over teacher
0.2	2.01 ± 0.65 %
0.5	5.22 ± 0.38 %
0.7	6.60 ± 0.25 %
1.0	7.71 ± 0.22 %
1.2	$\mathbf{8.11 \pm 0.06}$ %
1.5	6.32 ± 0.92 %
1.7	3.97 ± 0.42 %

Table 30. Average ($\pm$ 1 std.) improvement of student over teacher (i.e., student's test set accuracy - teacher's test set accuracy) for full network (viz., ResNet-34) training from scratch in the case of Caltech-256 with varying levels of random corruption and $\xi = 1$ (analogous to Table 3). Just like in the case of linear probing (Table 3), as the noise level increases, so does the improvement of the student over the teacher.

Corruption level	Improvement of student over teacher
0%	0.18 ± 0.10 %
10%	0.98 ± 0.16 %
30%	6.32 ± 0.36 %
50%	7.71 ± 0.22 %