RATT: Leveraging Unlabeled Data to Guarantee Generalization

Saurabh Garg [1] **Sivaraman Balakrishnan** [1] **J. Zico Kolter** [1] **Zachary C. Lipton** [1]

Abstract

To assess generalization, machine learning scientists typically either (i) bound the generalization gap and then (after training) plug in the empirical risk to obtain a bound on the true risk; or (ii) validate empirically on holdout data. However, (i) typically yields vacuous guarantees for overparameterized models; and (ii) shrinks the training set and its guarantee erodes with each reuse of the holdout set. In this paper, we leverage unlabeled data to produce generalization bounds. After augmenting our (labeled) training set with randomly labeled data, we train in the standard fashion. Whenever classifiers achieve low error on the clean data but high error on the random data, our bound ensures that the true risk is low. We prove that our bound is valid for 0-1 empirical risk minimization and with linear classifiers trained by gradient descent. Our approach is especially useful in conjunction with deep learning due to the early learning phenomenon whereby networks fit true labels before noisy labels but requires one intuitive assumption. Empirically, on canonical computer vision and NLP tasks, our bound provides non-vacuous generalization guarantees that track actual performance closely. This work enables practitioners to certify generalization even when (labeled) holdout data is unavailable and provides insights into the relationship between random label noise and generalization. Code is available at https://github.com/acmi-lab/RATT_generalization_bound.

1. Introduction

Typically, machine learning scientists establish generalization in one of two ways. One approach, favored by learning theorists, places an *a priori* bound on the gap between the empirical and true risks, usually in terms of the complexity of the hypothesis class. After fitting the model on the available data, one can plug in the empirical risk to obtain a guarantee on the true risk. The second approach, favored by practitioners, involves splitting the available data into training and holdout partitions, fitting the models on the former and estimating the population risk with the latter.

Surely, both approaches are useful, with the former providing theoretical insights and the latter guiding the development of a vast array of practical technology. Nevertheless, both methods have drawbacks. Most *a priori* generalization bounds rely on uniform convergence and thus fail to explain the ability of overparameterized networks to generalize (Zhang et al., 2016; Nagarajan & Kolter, 2019b). On the other hand, provisioning a holdout dataset restricts the amount of labeled data available for training. Moreover, risk estimates based on holdout sets lose their validity with successive re-use of the holdout data due to adaptive overfitting (Murphy, 2012; Dwork et al., 2015; Blum & Hardt, 2015). However, recent empirical studies suggest that on large benchmark datasets, adaptive overfitting is surprisingly absent (Recht et al., 2019).

In this paper, we propose Randomly Assign, Train and Track (RATT), a new method that leverages unlabeled data to provide a *post-training* bound on the true risk (i.e., the population error). Here, we assign random labels to a fresh batch of unlabeled data, augmenting the clean training dataset with these randomly labeled points. Next, we train on this data, following standard risk minimization practices. Finally, we track the error on the randomly labeled portion of training data, estimating the error on the mislabeled portion and using this quantity to upper bound the population error.

Counterintuitively, we guarantee generalization by guaranteeing overfitting. Specifically, we prove that Empirical Risk Minimization (ERM) with 0-1 loss leads to lower error on the *mislabeled training data* than on the *mislabeled population*. Thus, if despite minimizing the loss on the combined training data, we nevertheless have high error on the mislabeled portion, then the (mislabeled) population error will be even higher. Then, by complementarity, the (clean) population error must be low. Finally, we show how to obtain this guarantee using randomly labeled (vs mislabeled data), thus enabling us to incorporate unlabeled data.

To expand the applicability of our idea beyond ERM on 0-1

[1]Carnegie Mellon University. Correspondence to: Saurabh Garg <sgarg2@andrew.cmu.edu>.

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

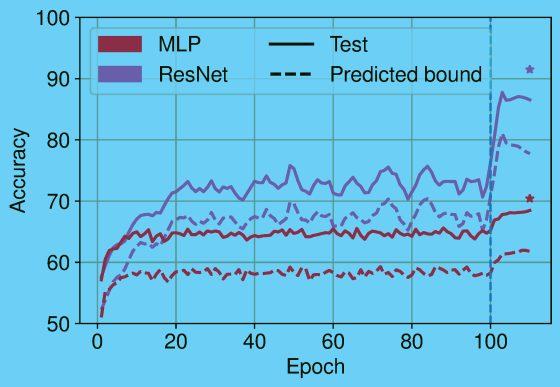


Figure 1. **Predicted lower bound on the clean population error** with ResNet and MLP on binary CIFAR. Results aggregated over 5 seeds. '*' denotes the best test performance achieved when training with only clean data and the same hyperparameters (except for the stopping point). The bound predicted by RATT (RHS in (2)) closely tracks the population accuracy on clean data.

error, we prove corresponding results for a linear classifier trained by gradient descent to minimize squared loss. Furthermore, leveraging the connection between early stopping and ℓ_2-regularization in linear models (Ali et al., 2018; 2020; Suggala et al., 2018), our results extend to early-stopped gradient descent. Because we make no assumptions on the data distribution, our results on linear models hold for more complex models such as kernel regression and neural networks in the Neural Tangent Kernel (NTK) regime (Jacot et al., 2018; Du et al., 2018; 2019; Allen-Zhu et al., 2019b; Chizat et al., 2019).

Addressing practical deep learning models, our guarantee requires an additional (reasonable) assumption. Our experiments show that the bound yields non-vacuous guarantees that track test error across several major architectures on a range of benchmark datasets for computer vision and Natural Language Processing (NLP). Because, in practice, overparameterized deep networks exhibit an *early learning phenomenon*, fitting clean data before mislabeled data (Liu et al., 2020; Arora et al., 2019; Li et al., 2019), our procedure yields tight bounds in the early phases of learning. Experimentally, we confirm the early learning phenomenon in standard Stochastic Gradient Descent (SGD) training and illustrate the effectiveness of weight decay combined with large initial learning rates in avoiding interpolation to mislabeled data while maintaining fit on the training data, strengthening the guarantee provided by our method.

To be clear, we do not advocate RATT as a blanket replacement for the holdout approach. Our main contribution is to introduce a new theoretical perspective on generalization and to provide a method that may be applicable even when the holdout approach is unavailable. Of interest, unlike generalization bounds based on uniform-convergence that restrict the complexity of the hypothesis class (Neyshabur et al., 2018; 2015; 2017b; Bartlett et al., 2017; Nagarajan & Kolter, 2019a), our *post hoc* bounds depend only on the fit to mislabeled data. We emphasize that our theory does not guarantee *a priori* that early learning should take place but only *a posteriori* that when it does, we can provide non-vacuous bounds on the population error. Conceptually, this finding underscores the significance of the early learning phenomenon in the presence of noisy labels and motivates further work to explain why it occurs.

2. Preliminaries

By $\|\cdot\|$, and $\langle\cdot,\cdot\rangle$ we denote the Euclidean norm and inner product, respectively. For a vector $v \in \mathbb{R}^d$, we use v_j to denote its j^{th} entry, and for an event E we let $\mathbb{I}[E]$ denote the binary indicator of the event.

Suppose we have a multiclass classification problem with the input domain $\mathcal{X} \subseteq \mathbb{R}^d$ and label space $\mathcal{Y} = \{1, 2, \ldots, k\}$[1]. By $\mathcal{D}$, we denote the distribution over $\mathcal{X} \times \mathcal{Y}$. A dataset $S := \{(x_i, y_i)\}_{i=1}^n \sim \mathcal{D}^n$ contains n points sampled i.i.d. from $\mathcal{D}$. By $\mathcal{S}$, $\mathcal{T}$, and $\widetilde{\mathcal{S}}$, we denote the (uniform) empirical distribution over points in datasets S, T, and $\widetilde{S}$, respectively. Let $\mathcal{F}$ be a class of hypotheses mapping $\mathcal{X}$ to $\mathbb{R}^k$. A *training algorithm* $\mathcal{A}$: takes a dataset S and returns a classifier $f(\mathcal{A}, S) \in \mathcal{F}$. When the context is clear, we drop the parentheses for convenience. Given a classifier f and datum (x, y), we denote the 0-1 error (i.e., classification error) on that point by $\mathcal{E}(f(x), y) := \mathbb{I}\left[y \notin \arg\max_{j \in \mathcal{Y}} f_j(x)\right]$, We express the *population error* on $\mathcal{D}$ as $\mathcal{E}_{\mathcal{D}}(f) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\mathcal{E}(f(x), y)\right]$ and the *empirical error* on S as $\mathcal{E}_{\mathcal{S}}(f) := \mathbb{E}_{(x,y)\sim\mathcal{S}}\left[\mathcal{E}(f(x), y)\right] = \frac{1}{n}\sum_{i=1}^n \mathcal{E}(f(x_i), y_i)$.

Throughout, we consider a *random label assignment* procedure: draw $x \sim \mathcal{D}_{\mathcal{X}}$ (the underlying distribution over $\mathcal{X}$), and then assign a label sampled uniformly at random. We denote a randomly labeled dataset by $\widetilde{S} := \{(x_i, y_i)\}_{i=1}^m \sim \widetilde{\mathcal{D}}^m$, where $\widetilde{\mathcal{D}}$ is the distribution of randomly labeled data. By $\mathcal{D}'$, we denote the mislabeled distribution that corresponds to selecting examples (x, y) according to $\mathcal{D}$ and then re-assigning the label by sampling among the incorrect labels $y' \neq y$ (renormalizing the label marginal).

3. Generalization Bound for RATT with ERM

We now present our generalization bound and proof sketches for ERM on the 0-1 loss (full proofs in App. A). For any dataset T, ERM returns the classifier $\widehat{f}$ that minimizes the empirical error:

$$\widehat{f} := \arg\min_{f \in \mathcal{F}} \mathcal{E}_{\mathcal{T}}(f)\,. \tag{1}$$

[1]For binary classification, we use $\mathcal{Y} = \{-1, 1\}$.

We focus first on binary classification. Assume we have a clean dataset $S \sim \mathcal{D}^n$ of n points and a randomly labeled dataset $\widetilde{S} \sim \widetilde{\mathcal{D}}^m$ of m ($< n$) points with labels in $\widetilde{S}$ are assigned uniformly at random. We show that with 0-1 loss minimization on the union of S and $\widetilde{S}$, we obtain a classifier whose error on $\mathcal{D}$ is upper bounded by a function of the empirical errors on clean data $\mathcal{E}_S$ (lower is better) and on randomly labeled data $\mathcal{E}_{\widetilde{S}}$ (higher is better):

Theorem 1. *For any classifier $\widehat{f}$ obtained by ERM (1) on dataset $S \cup \widetilde{S}$, for any $\delta > 0$, with probability at least $1-\delta$, we have*

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_S(\widehat{f}) + 1 - 2\mathcal{E}_{\widetilde{S}}(\widehat{f}) + \left(\sqrt{2}\mathcal{E}_{\widetilde{S}}(\widehat{f}) + 2 + \frac{m}{2n}\right)\sqrt{\frac{\log(4/\delta)}{m}}. \quad (2)$$

In short, this theorem tells us that if after training on both clean and randomly labeled data, we achieve low error on the clean data but high error (close to $1/2$) on the randomly labeled data, then low population error is guaranteed. Note that because the labels in $\widetilde{S}$ are assigned randomly, the error $\mathcal{E}_{\widetilde{S}}(f)$ for any fixed predictor f (not dependent on $\widetilde{S}$) will be approximately 1/2. Thus, if ERM produces a classifier that has not fit to the randomly labeled data, then $(1 - 2\mathcal{E}_{\widetilde{S}}(\widehat{f}))$ will be approximately 0, and our error will be determined by the fit to clean data. The final term accounts for finite sample error—notably, it (i) does not depend on the complexity of the hypothesis class; and (ii) approaches 0 at a $\mathcal{O}(1/\sqrt{m})$ rate (for $m < n$).

Our proof strategy unfolds in three steps. First, in Lemma 1 we bound $\mathcal{E}_{\mathcal{D}}(\widehat{f})$ in terms of the error on the mislabeled subset of $\widetilde{S}$. Next, in Lemmas 2 and 3, we show that the error on the mislabeled subset can be accurately estimated using only clean and randomly labeled data.

To begin, assume that we actually knew the original labels for the randomly labeled data. By $\widetilde{S}_C$ and $\widetilde{S}_M$, we denote the clean and mislabeled portions of the randomly labeled data, respectively (with $\widetilde{S} = \widetilde{S}_M \cup \widetilde{S}_C$). Note that for binary classification, a lower bound on mislabeled population error $\mathcal{E}_{\mathcal{D}'}(\widehat{f})$ directly upper bounds the error on the original population $\mathcal{E}_{\mathcal{D}}(\widehat{f})$. Thus we only need to prove that the empirical error on the mislabeled portion of our data is lower than the error on unseen mislabeled data, i.e., $\mathcal{E}_{\widetilde{S}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) = 1 - \mathcal{E}_{\widetilde{S}_M}(\widehat{f})$ (upto $\mathcal{O}(1/\sqrt{m})$).

Lemma 1. *Assume the same setup as in Theorem 1. Then for any $\delta > 0$, with probability at least $1-\delta$ over the random draws of mislabeled data $\widetilde{S}_M$, we have*

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant 1 - \mathcal{E}_{\widetilde{S}_M}(\widehat{f}) + \sqrt{\frac{\log(1/\delta)}{m}}. \quad (3)$$

Proof Sketch. The main idea of our proof is to regard the clean portion of the data ($S \cup \widetilde{S}_C$) as fixed. Then, there exists a classifier f^* that is optimal over draws of the mislabeled data $\widetilde{S}_M$. Formally,

$$f^* := \arg\min_{f \in \mathcal{F}} \mathcal{E}_{\breve{\mathcal{D}}}(f),$$

where $\breve{\mathcal{D}}$ is a combination of the *empirical distribution* over correctly labeled data $S \cup \widetilde{S}_C$ and the (population) distribution over mislabeled data $\mathcal{D}'$. Recall that $\widehat{f} := \arg\min_{f \in \mathcal{F}} \mathcal{E}_{\mathcal{S} \cup \widetilde{\mathcal{S}}}(f)$. Since, $\widehat{f}$ minimizes 0-1 error on $S \cup \widetilde{S}$, we have $\mathcal{E}_{\mathcal{S} \cup \widetilde{\mathcal{S}}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S} \cup \widetilde{\mathcal{S}}}(f^*)$. Moreover, since f^* is independent of $\widetilde{S}_M$, we have with probability at least $1-\delta$ that

$$\mathcal{E}_{\widetilde{\mathcal{S}}_M}(f^*) \leqslant \mathcal{E}_{\mathcal{D}'}(f^*) + \sqrt{\frac{\log(1/\delta)}{m}}.$$

Finally, since f^* is the optimal classifier on $\breve{\mathcal{D}}$, we have $\mathcal{E}_{\breve{\mathcal{D}}}(f^*) \leqslant \mathcal{E}_{\breve{\mathcal{D}}}(\widehat{f})$. Combining the above steps and using the fact that $\mathcal{E}_{\mathcal{D}} = 1 - \mathcal{E}_{\mathcal{D}'}$, we obtain the desired result. □

While the LHS in (3) depends on the unknown portion $\widetilde{S}_M$, our goal is to use unlabeled data (with randomly assigned labels) for which the mislabeled portion cannot be readily identified. Fortunately, we do not need to identify the mislabeled points to estimate the error on these points in aggregate $\mathcal{E}_{\widetilde{S}_M}(\widehat{f})$. Note that because the label marginal is uniform, approximately half of the data will be correctly labeled and the remaining half will be mislabeled. Consequently, we can utilize the value of $\mathcal{E}_{\widetilde{S}}(\widehat{f})$ and an estimate of $\mathcal{E}_{\widetilde{S}_C}(\widehat{f})$ to lower bound $\mathcal{E}_{\widetilde{S}_M}(\widehat{f})$. We formalize this as follows:

Lemma 2. *Assume the same setup as Theorem 1. Then for any $\delta > 0$, with probability at least $1-\delta$ over the random draws of $\widetilde{S}$, we have $\left|2\mathcal{E}_{\widetilde{S}}(\widehat{f}) - \mathcal{E}_{\widetilde{S}_C}(\widehat{f}) - \mathcal{E}_{\widetilde{S}_M}(\widehat{f})\right| \leqslant 2\mathcal{E}_{\widetilde{S}}(\widehat{f})\sqrt{\frac{\log(4/\delta)}{2m}}$.*

To complete the argument, we show that due to the exchangeability of the clean data S and the clean portion of the randomly labeled data S_C, we can estimate the error on the latter $\mathcal{E}_{\widetilde{S}_C}(\widehat{f})$ by the error on the former $\mathcal{E}_S(\widehat{f})$.

Lemma 3. *Assume the same setup as Theorem 1. Then for any $\delta > 0$, with probability at least $1-\delta$ over the random draws of $\widetilde{S}_C$ and S, we have $\left|\mathcal{E}_{\widetilde{S}_C}(\widehat{f}) - \mathcal{E}_S(\widehat{f})\right| \leqslant \left(1 + \frac{m}{2n}\right)\sqrt{\frac{\log(2/\delta)}{m}}$.*

Lemma 3 establishes a tight bound on the difference of the error of classifier $\widehat{f}$ on $\widetilde{S}_C$ and on S. The proof uses Hoeffding's inequality for randomly sampled points from a fixed population (Hoeffding, 1994; Bardenet et al., 2015).

Having established these core components, we can now summarize the proof strategy for Theorem 1. We bound the

population error on clean data (the term on the LHS of (2)) in three steps: (i) use Lemma 1 to upper bound the error on clean distribution $\mathcal{E}_{\mathcal{D}}(\widehat{f})$, by the error on mislabeled training data $\mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f})$; (ii) approximate $\mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f})$ by $\mathcal{E}_{\widetilde{\mathcal{S}}_C}(\widehat{f})$ and the error on randomly labeled training data (i.e., $\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f})$) using Lemma 2; and (iii) use Lemma 3 to estimate $\mathcal{E}_{\widetilde{\mathcal{S}}_C}(\widehat{f})$ using the error on clean training data ($\mathcal{E}_{\mathcal{S}}(\widehat{f})$).

Comparison with Rademacher bound Our bound in Theorem 1 shows that we can upper bound the clean population error of a classifier by estimating its accuracy on the clean and randomly labeled portions of the training data. Next, we show that our bound's dominating term is upper bounded by the *Rademacher complexity* (Shalev-Shwartz & Ben-David, 2014), a standard distribution-dependent complexity measure.

Proposition 1. *Fix a randomly labeled dataset $\widetilde{S} \sim \widetilde{\mathcal{D}}^m$. Then for any classifier $f \in \mathcal{F}$ (possibly dependent on $\widetilde{S}$)[2] and for any $\delta > 0$, with probability at least $1-\delta$ over random draws of $\widetilde{S}$, we have*

$$1 - 2\mathcal{E}_{\widetilde{S}}(f) \leqslant \mathbb{E}_{\epsilon,x}\left[\sup_{f\in\mathcal{F}}\left(\frac{\sum_i \epsilon_i f(x_i)}{m}\right)\right] + \sqrt{\frac{2\log(\frac{2}{\delta})}{m}},$$

where ϵ is drawn from a uniform distribution over $\{-1,1\}^m$ and x is drawn from $\mathcal{D}_{\mathcal{X}}^m$.

In other words, the proposition above highlights that the accuracy on the randomly labeled data is never larger than the Rademacher complexity of $\mathcal{F}$ w.r.t. the underlying distribution over $\mathcal{X}$, implying that our bound is never looser than a bound based on Rademacher complexity. The proof follows by application of the bounded difference condition and McDiarmid's inequality (McDiarmid, 1989). We now discuss extensions of Theorem 1 to regularized ERM and multiclass classification.

Extension to regularized ERM Consider any function $R : \mathcal{F} \to \mathbb{R}$, e.g., a regularizer that penalizes some measure of complexity for functions in class $\mathcal{F}$. Consider the following regularized ERM:

$$\widehat{f} := \arg\min_{f\in\mathcal{F}} \mathcal{E}_{\mathcal{S}}(f) + \lambda R(f)\,, \tag{4}$$

where λ is a regularization constant. If the regularization coefficient is independent of the training data $S \cup \widetilde{S}$, then our guarantee (Theorem 1) holds. Formally,

Theorem 2. *For any regularization function R, assume we perform regularized ERM as in (4) on $S \cup \widetilde{S}$ and obtain a classifier $\widehat{f}$. Then, for any $\delta > 0$, with probability at least $1-\delta$, we have $\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}}(\widehat{f}) + 1 - 2\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f}) + \left(\sqrt{2}\mathcal{E}_{\widetilde{S}}(\widehat{f}) + 2 + \frac{m}{2n}\right)\sqrt{\frac{\log(1/\delta)}{m}}$.*

A key insight here is that the proof of Theorem 1 treats the clean data S as fixed and considers random draws of the mislabeled portion. Thus a data-independent regularization function does not alter our chain of arguments and hence, has no impact on the resulting inequality. We prove this result formally in App. A.

We note one immediate corollary from Theorem 2: when learning any function f parameterized by w with L_2-norm penalty on the parameters w, the population error with $\widehat{f}$ is determined by the error on the clean training data as long as the error on randomly labeled data is high (close to $1/2$).

Extension to multiclass classification Thus far, we have addressed binary classification. We now extend these results to the multiclass setting. As before, we obtain datasets S and $\widetilde{S}$. Here, random labels are assigned uniformly among all classes.

Theorem 3. *For any regularization function R, assume we perform regularized ERM as in (4) on $S \cup \widetilde{S}$ and obtain a classifier $\widehat{f}$. For a multiclass classification problem with k classes, for any $\delta > 0$, with probability at least $1-\delta$, we have*

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}}(\widehat{f}) + (k-1)\left(1 - \tfrac{k}{k-1}\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f})\right) + c\sqrt{\frac{\log(\frac{4}{\delta})}{2m}}\,, \tag{5}$$

for some constant $c \leqslant (2k + \sqrt{k} + \frac{m}{n\sqrt{k}})$.

We first discuss the implications of Theorem 3. Besides empirical error on clean data, the dominating term in the above expression is given by $(k-1)\left(1 - \frac{k}{k-1}\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f})\right)$. For any predictor f (not dependent on $\widetilde{S}$), the term $\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f})$ would be approximately $(k-1)/k$ and for $\widehat{f}$, the difference now evaluates to the accuracy of the randomly labeled data. Note that for binary classification, (5) simplifies to Theorem 1.

The core of our proof involves obtaining an inequality similar to (3). While for binary classification, we could upper bound $\mathcal{E}_{\widetilde{\mathcal{S}}_M}$ with $1 - \mathcal{E}_{\mathcal{D}}$ (in the proof of Lemma 1), for multiclass classification, error on the mislabeled data and accuracy on the clean data in the population are not so directly related. To establish an inequality analogous to (3), we break the error on the (unknown) mislabeled data into two parts: one term corresponds to predicting the true label on mislabeled data, and the other corresponds to predicting neither the true label nor the assigned (mis-)label. Finally, we relate these errors to their population counterparts to establish an inequality similar to (3).

[2] We restrict $\mathcal{F}$ to functions which output a label in $\mathcal{Y} = \{-1, 1\}$.

4. Generalization Bound for RATT with Gradient Descent

In the previous section, we presented results with ERM on 0-1 loss. While minimizing the 0-1 loss is hard in general, these results provide important theoretical insights. In this section, we show parallel results for linear models trained with Gradient Descent (GD).

To begin, we introduce the setup and some additional notation. For simplicity, we begin discussion with binary classification with $\mathcal{X} = \mathbb{R}^d$. Define a linear function $f(x; w) := w^T x$ for some $w \in \mathbb{R}^d$ and $x \in \mathcal{X}$. Given training set S, we suppose that the parameters of the linear function are obtained via gradient descent on the following L_2 regularized problem:

$$\mathcal{L}_S(w; \lambda) := \sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2 , \qquad (6)$$

where $\lambda \geqslant 0$ is a regularization parameter. Our choice to analyze squared loss minimization for linear networks is motivated in part by its analytical convenience, and follows recent theoretical work which analyze neural networks trained via squared loss minimization in the Neural Tangent Kernel (NTK) regime when they are well approximated by linear networks (Jacot et al., 2018; Arora et al., 2019; Du et al., 2019; Hu et al., 2019). Moreover, recent research suggests that for classification tasks, squared loss minimization performs comparably to cross-entropy loss minimization (Muthukumar et al., 2020; Hui & Belkin, 2020).

For a given training set S, we use $S_{(i)}$ to denote the training set S with the i^{th} point removed. We now introduce one stability condition:

Condition 1 (Hypothesis Stability)**.** *We have β hypothesis stability if our training algorithm $\mathcal{A}$ satisfies the following for all $i \in \{1, 2, \ldots, n\}$:*

$$\mathbb{E}_{S,(x,y)\in\mathcal{D}} \left[\left| \mathcal{E}\left(f(x), y\right) - \mathcal{E}\left(f_{(i)}(x), y\right) \right| \right] \leqslant \frac{\beta}{n} ,$$

where $f_{(i)} := f(\mathcal{A}, S_{(i)})$ and $f := f(\mathcal{A}, S)$.

This condition is similar to a notion of stability called *hypothesis stability* (Bousquet & Elisseeff, 2002; Kearns & Ron, 1999; Elisseeff et al., 2003). Intuitively, Condition 1 states that empirical leave-one-out error and average population error of leave-one-out classifiers are close. This condition is mild and does not guarantee generalization. We discuss the implications in more detail in App. B.3.

Now we present the main result of this section. As before, we assume access to a clean dataset $S = \{(x_i, y_i)\}_{i=1}^{n} \sim \mathcal{D}^n$ and randomly labeled dataset $\widetilde{S} = \{(x_i, y_i)\}_{i=n+1}^{n+m} \sim \widetilde{\mathcal{D}}^m$. Let $\boldsymbol{X} = [x_1, x_2, \cdots, x_{m+n}]$ and $\boldsymbol{y} = [y_1, y_2, \cdots, y_{m+n}]$. Fix a positive learning rate η such that $\eta \leqslant 1/\left(\left\|\boldsymbol{X}^T\boldsymbol{X}\right\|_{\text{op}} + \lambda^2\right)$ and an initialization $w_0 = 0$. Consider the following gradient descent iterates to minimize objective (6) on $S \cup \widetilde{S}$:

$$w_t = w_{t-1} - \eta \nabla_w \mathcal{L}_{S\cup\widetilde{S}}(w_{t-1}; \lambda) \quad \forall t = 1, 2, \ldots . \qquad (7)$$

Then we have $\{w_t\}$ converge to the limiting solution $\widehat{w} = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{X}^T \boldsymbol{y}$. Define $\widehat{f}(x) := f(x; \widehat{w})$.

Theorem 4. *Assume that this gradient descent algorithm satisfies Condition 1 with $\beta = \mathcal{O}(1)$. Then for any $\delta > 0$, with probability at least $1 - \delta$ over the random draws of datasets $\widetilde{S}$ and S, we have:*

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_{S}(\widehat{f}) + 1 - 2\mathcal{E}_{\widetilde{S}}(\widehat{f}) + \sqrt{\frac{4}{\delta}\left(\frac{1}{m} + \frac{3\beta}{m+n}\right)} + \left(\sqrt{2}\mathcal{E}_{\widetilde{S}}(\widehat{f}) + 1 + \frac{m}{2n}\right)\sqrt{\frac{\log(4/\delta)}{m}} . \qquad (8)$$

With a mild regularity condition, we establish the same bound on GD training with squared loss, notably the same dominating term on the population error, as in Theorem 1. In App. B.2, we present the extension to multiclass classification, where we again obtain a result parallel to Theorem 3.

Proof Sketch. Because squared loss minimization does not imply 0-1 error minimization, we cannot use arguments from Lemma 1. This is the main technical difficulty. To compare the 0-1 error at a train point with an unseen point, we use the closed-form expression for $\widehat{w}$. We show that the train error on mislabeled points is less than the population error on the distribution of mislabeled data (parallel to Lemma 1).

For a mislabeled training point (x_i, y_i) in $\widetilde{S}$, we show that

$$\mathbb{I}\left[y_i x_i^T \widehat{w} \leqslant 0\right] \leqslant \mathbb{I}\left[y_i x_i^T \widehat{w}_{(i)} \leqslant 0\right] , \qquad (9)$$

where $\widehat{w}_{(i)}$ is the classifier obtained by leaving out the i^{th} point from the training set. Intuitively, this condition states that the train error at a training point is less than the leave-one-out error at that point, i.e. the error obtained by removing that point and re-training. Using Condition 1, we then relate the average leave-one-out error (over the index i of the RHS in (9)) to the population error on the mislabeled distribution to obtain an inequality similar to (3). □

Extensions to kernel regression Since the result in Theorem 4 does not impose any regularity conditions on the underlying distribution over $\mathcal{X} \times \mathcal{Y}$, our guarantees extend straightforwardly to kernel regression by using the transformation $x \to \phi(x)$ for some feature transform function ϕ. Furthermore, recent literature has pointed out a concrete connection between neural networks and kernel regression with

the so-called *Neural Tangent Kernel* (NTK) which holds in a certain regime where weights do not change much during training (Jacot et al., 2018; Du et al., 2019; 2018; Chizat et al., 2019). Using this concrete correspondence, our bounds on the clean population error (Theorem 4) extend to wide neural networks operating in the NTK regime.

Extensions to early stopped GD Often in practice, gradient descent is stopped early. We now provide theoretical evidence that our guarantees may continue to hold for an early stopped GD iterate. Concretely, we show that in expectation, the outputs of the GD iterates are close to that of a problem with data-independent regularization (as considered in Theorem 2). First, we introduce some notation. By $\mathcal{L}_S(w)$, we denote the objective in (6) with $\lambda = 0$. Consider the GD iterates defined in (7). Let $\widetilde{w}_\lambda = \arg\min_w \mathcal{L}_\mathcal{S}(w;\lambda)$. Define $f_t(x) := f(x; w_t)$ as the solution at the t^{th} iterate and $\widetilde{f}_\lambda(x) := f(x;\widetilde{w}_\lambda)$ as the regularized solution. Let κ be the condition number of the population covariance matrix and let $s_{\min}$ be the minimum positive singular value of the empirical covariance matrix.

Proposition 2 (informal)**.** *For $\lambda = \frac{1}{t\eta}$, we have*

$$\mathbb{E}_{x\sim\mathcal{D}_\mathcal{X}}\left[(f_t(x) - \widetilde{f}_\lambda(x))^2\right] \leqslant c(t,\eta)\cdot\mathbb{E}_{x\sim\mathcal{D}_\mathcal{X}}\left[f_t(x)^2\right],$$

where $c(t,\eta) \approx \kappa\cdot\min(0.25, \frac{1}{s_{min}^2 t^2\eta^2})$. An equivalent guarantee holds for a point x sampled from the training data.

The proposition above states that for large enough t, GD iterates stay close to a regularized solution with data-independent regularization constant. Together with our guarantees in Theorem 4 for regularization solution with $\lambda = \frac{1}{t\eta}$, Proposition 2 shows that our guarantees with RATT may hold on early stopped GD. See the formal result in App. B.4.

Remark Proposition 2 only bounds the expected squared difference between the t^{th} gradient descent iterate and a corresponding regularized solution. The expected squared difference and the expected difference of classification errors (what we wish to bound) are not related, in general. However, they can be related under standard low-noise (margin) assumptions. For instance, under the Tsybakov noise condition (Tsybakov et al., 1997; Yao et al., 2007), we can lower-bound the expression on the LHS of Proposition 2 with the difference of expected classification error.

Extensions to deep learning Note that the main lemma underlying our bound on (clean) population error states that when training on a mixture of clean and randomly labeled data, we obtain a classifier whose empirical error on the mislabeled training data is lower than its population error on the distribution of mislabeled data. We prove this for ERM on 0-1 loss (Lemma 1). For linear models (and networks in NTK regime), we obtained this result by assuming hypothesis stability and relating training error at a datum with the leave-one-out error (Theorem 4). However, to extend our bound to deep models we must assume that training on the mixture of random and clean data leads to overfitting on the random mixture. Formally:

Assumption 1. *Let $\widehat{f}$ be a model obtained by training with an algorithm $\mathcal{A}$ on a mixture of clean data S and randomly labeled data $\widetilde{S}$. Then with probability $1-\delta$ over the random draws of mislabeled data $\widetilde{S}_M$, we assume that the following condition holds:*

$$\mathcal{E}_{\widetilde{S}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) + c\sqrt{\frac{\log(1/\delta)}{2m}},$$

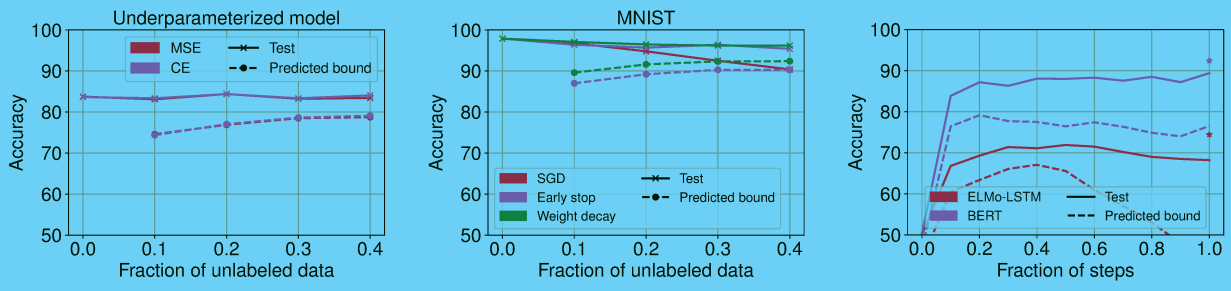
for a fixed constant $c > 0$.

Under Assumption 1, our results in Theorem 1, 2 and 3 extend beyond ERM with the 0-1 loss to general learning algorithms. We include the formal result in App. B.5. Note that given the ability of neural networks to interpolate the data, this assumption seems uncontroversial in the later stages of training. Moreover, concerning the early phases of training, recent research has shown that learning dynamics for complex deep networks resemble those for linear models (Nakkiran et al., 2019; Hu et al., 2020), much like the wide neural networks that we do analyze. Together, these arguments help to justify Assumption 1 and hence, the applicability of our bound in deep learning. Motivated by our analysis on linear models trained with gradient descent, we discuss conditions in App. B.6 which imply Assumption 1 for constant values $\delta > 0$. In the next section, we empirically demonstrate applicability of our bounds for deep models.

5. Empirical Study and Implications

Having established our framework theoretically, we now demonstrate its utility experimentally. First, for linear models and wide networks in the NTK regime where our guarantee holds, we confirm that our bound is not only valid, but closely tracks the generalization error. Next, we show that in practical deep learning settings, optimizing cross-entropy loss by SGD, the expression for our (0-1) ERM bound nevertheless tracks test performance closely and in numerous experiments on diverse models and datasets is never violated empirically.

Datasets To verify our results on linear models, we consider a toy dataset, where the class conditional distribution $p(x|y)$ for each label is Gaussian. For binary tasks, we use binarized CIFAR-10 (first 5 classes vs rest) (Krizhevsky & Hinton, 2009), binary MNIST (0-4 vs 5-9) (LeCun et al., 1998) and IMDb sentiment analysis dataset (Maas et al., 2011). For multiclass setup, we use MNIST and CIFAR-10.



(a) (b) (c)

Figure 2. We plot the accuracy and corresponding bound (RHS in (1)) at $\delta = 0.1$. for binary classification tasks. Results aggregated over 3 seeds. (a) Accuracy vs fraction of unlabeled data (w.r.t clean data) in the toy setup with a linear model trained with GD. (b) Accuracy vs fraction of unlabeled data for a 2-layer wide network trained with SGD on binary MNIST. With SGD and no regularization (red curve in (b)), we interpolate the training data and hence the predicted lower bound is 0. However, with early stopping (or weight decay) we obtain tight guarantees. (c) Accuracy vs gradient iteration on IMDb dataset with unlabeled fraction fixed at 0.2. In plot (c), '*' denotes the best test accuracy with the same hyperparameters and training only on clean data. See App. C for exact hyperparameter values.

Architectures To simulate the NTK regime, we experiment with 2-layered wide networks both (i) with the second layer fixed at random initialization; (ii) and updating both layers' weights. For vision datasets (e.g., MNIST and CIFAR10), we consider (fully connected) multilayer perceptrons (MLPs) with ReLU activations and ResNet18 (He et al., 2016). For the IMDb dataset, we train Long Short-Term Memory Networks (LSTMs; Hochreiter & Schmidhuber (1997)) with ELMo embeddings (Peters et al., 2018) and fine-tune an off-the-shelf uncased BERT model (Devlin et al., 2018; Wolf et al., 2020).

Methodology To bound the population error, we require access to both clean and unlabeled data. For toy datasets, we obtain unlabeled data by sampling from the underlying distribution over $\mathcal{X}$. For image and text datasets, we hold out a small fraction of the clean training data and discard their labels to simulate unlabeled data. We use the random labeling procedure described in Sec. 2. After augmenting clean training data with randomly labeled data, we train in the standard fashion. See App. C for experimental details.

Underparameterized linear models On toy Gaussian data, we train linear models with GD to minimize cross-entropy loss and mean squared error. Varying the fraction of randomly labeled data we observe that the accuracy on clean unseen data is barely impacted (Fig. 2(a)). This highlights that in low dimensional models adding randomly labeled data with the clean dataset (in toy setup) has minimal effect on the performance on unseen clean data. Moreover, we find that RATT offers a tight lower bound on the unseen clean data accuracy. We observe the same behavior with Stochastic Gradient Descent (SGD) training (ref. App. C). Observe that the predicted bound goes up as the fraction of unlabeled data increases. While the accuracy as dictated by the dominating term in the RHS of (2) decreases with an increase in the fraction of unlabeled data, we observe a relatively sharper decrease in $\mathcal{O}_p\left(1/\sqrt{m}\right)$ term of the bound, leading to an overall increase in the predicted accuracy bound. In this toy setup, we also evaluated a kernel regression bound from Bartlett & Mendelson (2002) (Theorem 21), however, the predicted kernel regression bound remains vacuous.

Wide Nets Next, we consider MNIST binary classification with a wide 2-layer fully-connected network. In experiments with SGD training on MSE loss without early stopping or weight decay regularization, we find that adding extra randomly label data hurts the unseen clean performance (Fig. 2(b)). Additionally, due to the perfect fit on the training data, our bound is rendered vacuous. However, with early stopping (or weight decay), we observe close to zero performance difference with additional randomly labeled data. Alongside, we obtain tight bounds on the accuracy on unseen clean data paying only a small price to negligible for incorporating randomly labeled data. Similar results hold for SGD and GD and when cross-entropy loss is substituted for MSE (ref. App. C).

Deep Nets We verify our findings on (i) ResNet-18 and 5-layer MLPs trained with binary CIFAR (Fig. 1); and (ii) ELMo-LSTM and BERT-Base models fine-tuned on the IMDb dataset (Fig. 2(c)). See App. C for additional results with deep models on binary MNIST. We fix the amount of unlabeled data at 20% of the clean dataset size and train all models with standard hyperparameters. Consistently, we find that our predicted bounds are never violated in practice. And as training proceeds, the fit on the mislabeled data increases with perfect overfitting in the interpolation regime rendering our bounds vacuous. However, with early stopping, our bound predicts test performance closely. For example, on IMDb dataset with BERT fine-tuning we predict 79.8 as the accuracy of the classifier, when the true

Dataset	Model	Pred. Acc	Test Acc.	Best Acc.
MNIST	MLP	93.1	97.4	97.9
	ResNet	96.8	98.8	98.9
CIFAR10	MLP	48.4	54.2	60.0
	ResNet	76.4	88.9	92.3

Table 1. Results on multiclass classification tasks. With pred. acc. we refer to the dominating term in RHS of (5). At the given sample size and $\delta = 0.1$, the remaining term evaluates to 30.7, decreasing our predicted accuracy by the same. We note that test acc. denotes the corresponding accuracy on unseen clean data. Best acc. is the best achievable accuracy with just training on just the clean data (and same hyperparamters except the stopping point). Note that across all tasks our predicted bound is tight and the gap between the best accuracy and test accuracy is small. Exact hyperparameters are included in App. C.

performance is 88.04 (and the best achievable performance on unseen data is 92.45). Additionally, we observe that our method tracks the performance from the beginning of the training and not just towards the end.

Finally, we verify our multiclass bound on MNIST and CIFAR10 with deep MLPs and ResNets (see results in Table 1 and per-epoch curves in App. C). As before, we fix the amount of unlabeled data at 20% of the clean dataset to minimize cross-entropy loss via SGD. In all four settings, our bound predicts non-vacuous performance on unseen data. In App. C, we investigate our approach on CIFAR100 showing that even though our bound grows pessimistic with greater numbers of classes, the error on the mislabeled data nevertheless tracks population accuracy.

6. Discussion and Connections to Prior Work

Implicit bias in deep learning Several recent lines of research attempt to explain the generalization of neural networks despite massive overparameterization via the *implicit bias* of gradient descent (Soudry et al., 2018; Gunasekar et al., 2018a;b; Ji & Telgarsky, 2019; Chizat & Bach, 2020). Noting that even for overparameterized linear models, there exist multiple parameters capable of overfitting the training data (with arbitrarily low loss), of which some generalize well and others do not, they seek to characterize the favored solution. Notably, Soudry et al. (2018) find that for linear networks, gradient descent converges (slowly) to the max margin solution. A complementary line of work focuses on the early phases of training, finding both empirically (Rolnick et al., 2017; Arpit et al., 2017) and theoretically (Arora et al., 2019; Li et al., 2020; Liu et al., 2020) that even in the presence of a small amount of mislabeled data, gradient descent is biased to fit the clean data first during initial phases of training. However, to best our knowledge, no prior work leverages this phenomenon to obtain generalization guarantees on the clean data, which is the primary focus of our work. Our method exploits this phenomenon to produce non-vacuous generalization bounds. Even when we cannot prove *a priori* that models will fit the clean data well while performing badly on the mislabeled data, we can observe that it indeed happens (often in practice), and thus, *a posteriori*, provide tight bounds on the population error. Moreover, by using regularizers like early stopping or weight decay, we can accentuate this phenomenon, enabling our framework to provide even tighter guarantees.

Generalization bounds Conventionally, generalization in machine learning has been studied through the lens of uniform convergence bounds (Blumer et al., 1989; Vapnik, 1999). Representative works on understanding generalization in overparameterized networks within this framework include Neyshabur et al. (2015; 2017b;a; 2018); Dziugaite & Roy (2017); Bartlett et al. (2017); Arora et al. (2018); Li & Liang (2018); Zhou et al. (2018); Allen-Zhu et al. (2019a); Nagarajan & Kolter (2019a). However, uniform convergence based bounds typically remain numerically loose relative to the true generalization error. Several works have also questioned the ability of uniform convergence based approaches to explain generalization in overparameterized models (Zhang et al., 2016; Nagarajan & Kolter, 2019b). Subsequently, recent works have proposed unconventional ways to derive generalization bounds (Negrea et al., 2020; Zhou et al., 2020). In a similar spirit, we take departure from complexity-based approaches to generalization bounds in our work. In particular, we leverage unlabeled data to derive a post-hoc generalization bound. Our work provides guarantees on overparameterized networks by using early stopping or weight decay regularization, preventing a perfect fit on the training data. Notably, in our framework, the model can perfectly fit the clean portion of the data, so long as they nevertheless fit the mislabeled data poorly.

Leveraging noisy data to provide generalization guarantees In parallel work, Bansal et al. (2020) presented an upper bound on the generalization gap of linear classifiers trained on representations learned via self-supervision. Under certain noise-robustness and rationality assumptions on the training procedure, the authors obtained bounds dependent on the complexity of the linear classifier and independent of the complexity of representations. By contrast, we present generalization bounds for supervised learning that are non-vacuous by virtue of the early learning phenomenon. While both frameworks highlight how robustness to random label corruptions can be leveraged to obtain bounds that do not depend directly on the complexity of the underlying hypothesis class, our framework, methodology, claims, and generalization results are very different from theirs.

Other related work. A long line of work relates early stopped GD to a corresponding regularized solution (Fried-

man & Popescu, 2003; Yao et al., 2007; Suggala et al., 2018; Ali et al., 2018; Neu & Rosasco, 2018; Ali et al., 2020). In the most relevant work, Ali et al. (2018) and Suggala et al. (2018) address a regression task, theoretically relating the solutions of early-stopped GD and a regularized problem, obtained with a data-independent regularization coefficient. Towards understanding generalization numerous stability conditions have been discussed (Kearns & Ron, 1999; Bousquet & Elisseeff, 2002; Mukherjee et al., 2006; Shalev-Shwartz et al., 2010). Hardt et al. (2016) studies the uniform stability property to obtain generalization guarantees with early-stopped SGD. While we assume a benign stability condition to relate leave-one-out performance with population error, we do not rely on any stability condition that implies generalization.

7. Conclusion and Future work

Our work introduces a new approach for obtaining generalization bounds that do not directly depend on the underlying complexity of the model class. For linear models, we provably obtain a bound in terms of the fit on randomly labeled data added during training. Our findings raise a number of questions to be explored next. While our empirical findings and theoretical results with 0-1 loss hold absent further assumptions and shed light on why the bound may apply for more general models, we hope to extend our proof that overfitting (in terms classification error) to the finite sample of mislabeled data occurs with SGD training on broader classes of models and loss functions. We hope to build on some early results (Nakkiran et al., 2019; Hu et al., 2020) which provide evidence that deep models behave like linear models in the early phases of training. We also wish to extend our framework to the interpolation regime. Since many important aspects of neural network learning take place within early epochs (Achille et al., 2017; Frankle et al., 2020), including gradient dynamics converging to very small subspace (Gur-Ari et al., 2018), we might imagine operationalizing our bounds in the interpolation regime by discarding the randomly labeled data after initial stages of training.

Acknowledgements

SG thanks Divyansh Kaushik for help with NLP code. This material is based on research sponsored by Air Force Research Laboratory (AFRL) under agreement number FA8750-19-1-1000. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation therein. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Air Force Laboratory, DARPA or the U.S. Government. SB acknowledges funding from the NSF grants DMS-1713003 and CIF-1763734, as well as Amazon AI and a Google Research Scholar Award. ZL acknowledges Amazon AI, Salesforce Research, Facebook, UPMC, Abridge, the PwC Center, the Block Center, the Center for Machine Learning and Health, and the CMU Software Engineering Institute (SEI), for their generous support of ACMI Lab's research on machine learning under distribution shift.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation*, 2016.

Abou-Moustafa, K. and Szepesvári, C. An exponential efron-stein inequality for lq stable learning rules. *arXiv preprint arXiv:1903.05457*, 2019.

Achille, A., Rovere, M., and Soatto, S. Critical learning periods in deep neural networks. *arXiv preprint arXiv:1711.08856*, 2017.

Ali, A., Kolter, J. Z., and Tibshirani, R. J. A continuous-time view of early stopping for least squares. *arXiv preprint arXiv:1810.10082*, 2018.

Ali, A., Dobriban, E., and Tibshirani, R. J. The implicit regularization of stochastic gradient flow for least squares. *arXiv preprint arXiv:2003.07802*, 2020.

Allen-Zhu, Z., Li, Y., and Liang, Y. Learning and generalization in overparameterized neural networks, going beyond two layers. In *Advances in neural information processing systems*, pp. 6158–6169, 2019a.

Allen-Zhu, Z., Li, Y., and Song, Z. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pp. 242–252. PMLR, 2019b.

Arora, S., Ge, R., Neyshabur, B., and Zhang, Y. Stronger generalization bounds for deep nets via a compression approach. *arXiv preprint arXiv:1802.05296*, 2018.

Arora, S., Du, S. S., Hu, W., Li, Z., and Wang, R. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. *arXiv preprint arXiv:1901.08584*, 2019.

Arpit, D., Jastrzebski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., et al. A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 233–242. JMLR. org, 2017.

Bansal, Y., Kaplun, G., and Barak, B. For self-supervised learning, rationality implies generalization, provably. *arXiv preprint arXiv:2010.08508*, 2020.

Bardenet, R., Maillard, O.-A., et al. Concentration inequalities for sampling without replacement. *Bernoulli*, 21(3): 1361–1385, 2015.

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. Spectrally-normalized margin bounds for neural networks. In *Advances in neural information processing systems*, pp. 6240–6249, 2017.

Blum, A. and Hardt, M. The ladder: A reliable leaderboard for machine learning competitions. In *International Conference on Machine Learning*, pp. 1006–1014. PMLR, 2015.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. Learnability and the vapnik-chervonenkis dimension. *Journal of the ACM (JACM)*, 36(4):929–965, 1989.

Bousquet, O. and Elisseeff, A. Stability and generalization. *Journal of machine learning research*, 2(Mar):499–526, 2002.

Chizat, L. and Bach, F. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Conference on Learning Theory*, pp. 1305–1338. PMLR, 2020.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, pp. 2937–2947, 2019.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pp. 1675–1685. PMLR, 2019.

Du, S. S., Zhai, X., Poczos, B., and Singh, A. Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*, 2018.

Dwork, C., Feldman, V., Hardt, M., Pitassi, T., Reingold, O., and Roth, A. L. Preserving statistical validity in adaptive data analysis. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pp. 117–126, 2015.

Dziugaite, G. K. and Roy, D. M. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv preprint arXiv:1703.11008*, 2017.

Elisseeff, A., Pontil, M., et al. Leave-one-out error and stability of learning algorithms with applications. *NATO science series sub series iii computer and systems sciences*, 190:111–130, 2003.

Frankle, J., Schwab, D. J., and Morcos, A. S. The early phase of neural network training. *arXiv preprint arXiv:2002.10365*, 2020.

Friedman, J. and Popescu, B. E. Gradient directed regularization for linear regression and classification. Technical report, Technical Report, Statistics Department, Stanford University, 2003.

Gunasekar, S., Lee, J., Soudry, D., and Srebro, N. Implicit bias of gradient descent on linear convolutional networks. *arXiv preprint arXiv:1806.00468*, 2018a.

Gunasekar, S., Woodworth, B., Bhojanapalli, S., Neyshabur, B., and Srebro, N. Implicit regularization in matrix factorization. In *2018 Information Theory and Applications Workshop (ITA)*, pp. 1–10. IEEE, 2018b.

Gur-Ari, G., Roberts, D. A., and Dyer, E. Gradient descent happens in a tiny subspace. *arXiv preprint arXiv:1812.04754*, 2018.

Hardt, M., Recht, B., and Singer, Y. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, pp. 1225–1234. PMLR, 2016.

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Hoeffding, W. Probability inequalities for sums of bounded random variables. In *The Collected Works of Wassily Hoeffding*, pp. 409–426. Springer, 1994.

Hu, W., Li, Z., and Yu, D. Simple and effective regularization methods for training on noisily labeled data with generalization guarantee. *arXiv preprint arXiv:1905.11368*, 2019.

Hu, W., Xiao, L., Adlam, B., and Pennington, J. The surprising simplicity of the early-time learning dynamics of neural networks. *arXiv preprint arXiv:2006.14599*, 2020.

Hui, L. and Belkin, M. Evaluation of neural architectures trained with square loss vs cross-entropy in classification tasks. *arXiv preprint arXiv:2006.07322*, 2020.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pp. 8571–8580, 2018.

Ji, Z. and Telgarsky, M. The implicit bias of gradient descent on nonseparable data. In *Conference on Learning Theory*, pp. 1772–1798. PMLR, 2019.

Kearns, M. and Ron, D. Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. *Neural computation*, 11(6):1427–1453, 1999.

Krizhevsky, A. and Hinton, G. Learning Multiple Layers of Features from Tiny Images. Technical report, Citeseer, 2009.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86, 1998.

Li, M., Soltanolkotabi, M., and Oymak, S. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. *arXiv preprint arXiv:1903.11680*, 2019.

Li, M., Soltanolkotabi, M., and Oymak, S. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *International Conference on Artificial Intelligence and Statistics*, pp. 4313–4324. PMLR, 2020.

Li, Y. and Liang, Y. Learning overparameterized neural networks via stochastic gradient descent on structured data. In *Advances in Neural Information Processing Systems*, pp. 8157–8166, 2018.

Liu, S., Niles-Weed, J., Razavian, N., and Fernandez-Granda, C. Early-learning regularization prevents memorization of noisy labels. *arXiv preprint arXiv:2007.00151*, 2020.

Maas, A., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 142–150, 2011.

McDiarmid, C. *On the method of bounded differences*, pp. 148–188. London Mathematical Society Lecture Note Series. Cambridge University Press, 1989.

Mukherjee, S., Niyogi, P., Poggio, T., and Rifkin, R. Learning theory: stability is sufficient for generalization and necessary and sufficient for consistency of empirical risk minimization. *Advances in Computational Mathematics*, 25(1):161–193, 2006.

Murphy, K. P. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

Muthukumar, V., Narang, A., Subramanian, V., Belkin, M., Hsu, D., and Sahai, A. Classification vs regression in overparameterized regimes: Does the loss function matter? *arXiv preprint arXiv:2005.08054*, 2020.

Nagarajan, V. and Kolter, J. Z. Deterministic pac-bayesian generalization bounds for deep networks via generalizing noise-resilience. *arXiv preprint arXiv:1905.13344*, 2019a.

Nagarajan, V. and Kolter, J. Z. Uniform convergence may be unable to explain generalization in deep learning. In *Advances in Neural Information Processing Systems*, pp. 11615–11626, 2019b.

Nakkiran, P., Kaplun, G., Kalimeris, D., Yang, T., Edelman, B. L., Zhang, F., and Barak, B. Sgd on neural networks learns functions of increasing complexity. *arXiv preprint arXiv:1905.11604*, 2019.

Negrea, J., Dziugaite, G. K., and Roy, D. In defense of uniform convergence: Generalization via derandomization with an application to interpolating predictors. In *International Conference on Machine Learning*, pp. 7263–7272. PMLR, 2020.

Neu, G. and Rosasco, L. Iterate averaging as regularization for stochastic gradient descent. In *Conference On Learning Theory*, pp. 3222–3242. PMLR, 2018.

Neyshabur, B., Tomioka, R., and Srebro, N. Norm-based capacity control in neural networks. In *Conference on Learning Theory*, pp. 1376–1401, 2015.

Neyshabur, B., Bhojanapalli, S., McAllester, D., and Srebro, N. Exploring generalization in deep learning. *arXiv preprint arXiv:1706.08947*, 2017a.

Neyshabur, B., Bhojanapalli, S., and Srebro, N. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017b.

Neyshabur, B., Li, Z., Bhojanapalli, S., LeCun, Y., and Srebro, N. The role of over-parametrization in generalization of neural networks. In *International Conference on Learning Representations*, 2018.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison,

M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, 2019.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. Deep contextualized word representations. In *Proc. of NAACL*, 2018.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pp. 5389–5400. PMLR, 2019.

Rolnick, D., Veit, A., Belongie, S., and Shavit, N. Deep learning is robust to massive label noise. *arXiv preprint arXiv:1705.10694*, 2017.

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Shalev-Shwartz, S., Shamir, O., Srebro, N., and Sridharan, K. Learnability, stability and uniform convergence. *The Journal of Machine Learning Research*, 11:2635–2670, 2010.

Sherman, J. and Morrison, W. J. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *The Annals of Mathematical Statistics*, 21 (1):124–127, 1950.

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

Suggala, A., Prasad, A., and Ravikumar, P. K. Connecting optimization and regularization paths. In *Advances in Neural Information Processing Systems*, pp. 10608–10619, 2018.

Tsybakov, A. B. et al. On nonparametric estimation of density level sets. *The Annals of Statistics*, 25(3):948–969, 1997.

Vapnik, V. N. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45. Association for Computational Linguistics, 2020.

Yao, Y., Rosasco, L., and Caponnetto, A. On early stopping in gradient descent learning. *Constructive Approximation*, 26(2):289–315, 2007.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Zhou, L., Sutherland, D. J., and Srebro, N. On uniform convergence and low-norm interpolation learning. *arXiv preprint arXiv:2006.05942*, 2020.

Zhou, W., Veitch, V., Austern, M., Adams, R. P., and Orbanz, P. Non-vacuous generalization bounds at the imagenet scale: a pac-bayesian compression approach. *arXiv preprint arXiv:1804.05862*, 2018.

Supplementary Material

Throughout this discussion, we will make frequently use of the following standard results concerning the exponential concentration of random variables:

Lemma 4 (Hoeffding's inequality for independent RVs (Hoeffding, 1994)). *Let $Z_1, Z_2, \ldots, Z_n$ be independent bounded random variables with $Z_i \in [a, b]$ for all i, then*

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^{n}(Z_i - \mathbb{E}[Z_i]) \geqslant t\right) \leqslant \exp\left(-\frac{2nt^2}{(b-a)^2}\right)$$

and

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^{n}(Z_i - \mathbb{E}[Z_i]) \leqslant -t\right) \leqslant \exp\left(-\frac{2nt^2}{(b-a)^2}\right)$$

for all $t \geqslant 0$.

Lemma 5 (Hoeffding's inequality for sampling with replacement (Hoeffding, 1994)). *Let $\mathcal{Z} = (Z_1, Z_2, \ldots, Z_N)$ be a finite population of N points with $Z_i \in [a.b]$ for all i. Let $X_1, X_2, \ldots X_n$ be a random sample drawn without replacement from $\mathcal{Z}$. Then for all $t \geqslant 0$, we have*

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^{n}(X_i - \mu) \geqslant t\right) \leqslant \exp\left(-\frac{2nt^2}{(b-a)^2}\right)$$

and

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^{n}(X_i - \mu) \leqslant -t\right) \leqslant \exp\left(-\frac{2nt^2}{(b-a)^2}\right),$$

where $\mu = \frac{1}{N}\sum_{i=1}^{N} Z_i$.

We now discuss one condition that generalizes the exponential concentration to dependent random variables.

Condition 2 (Bounded difference inequality). *Let $\mathcal{Z}$ be some set and $\phi : \mathcal{Z}^n \to \mathbb{R}$. We say that ϕ satisfies the bounded difference assumption if there exists $c_1, c_2, \ldots c_n \geqslant 0$ s.t. for all i, we have*

$$\sup_{Z_1, Z_2, \ldots, Z_n, Z_i' \in \mathcal{Z}^{n+1}} \left|\phi(Z_1, \ldots, Z_i, \ldots, Z_n) - \phi(Z_1, \ldots, Z_i', \ldots, Z_n)\right| \leqslant c_i\,.$$

Lemma 6 (McDiarmid's inequality (McDiarmid, 1989)). *Let $Z_1, Z_2, \ldots, Z_n$ be independent random variables on set $\mathcal{Z}$ and $\phi : \mathcal{Z}^n \to \mathbb{R}$ satisfy bounded difference inequality (Condition 2). Then for all $t > 0$, we have*

$$\mathbb{P}\left(\phi(Z_1, Z_2, \ldots, Z_n) - \mathbb{E}[\phi(Z_1, Z_2, \ldots, Z_n)] \geqslant t\right) \leqslant \exp\left(-\frac{2t^2}{\sum_{i=1}^{n} c_i^2}\right)$$

and

$$\mathbb{P}\left(\phi(Z_1, Z_2, \ldots, Z_n) - \mathbb{E}[\phi(Z_1, Z_2, \ldots, Z_n)] \leqslant -t\right) \leqslant \exp\left(-\frac{2t^2}{\sum_{i=1}^{n} c_i^2}\right).$$

A. Proofs from Sec. 3

Additional notation Let m_1 be the number of mislabeled points ($\widetilde{S}_M$) and m_2 be the number of correctly labeled points ($\widetilde{S}_C$). Note $m_1 + m_2 = m$.

A.1. Proof of Theorem 1

Proof of Lemma 1. The main idea of our proof is to regard the clean portion of the data ($S \cup \widetilde{S}_C$) as fixed. Then, there exists an (unknown) classifier f^* that minimizes the expected risk calculated on the (fixed) clean data and (random draws of) the mislabeled data $\widetilde{S}_M$. Formally,

$$f^* := \arg\min_{f \in \mathcal{F}} \mathcal{E}_{\breve{\mathcal{D}}}(f)\,, \tag{10}$$

where

$$\breve{\mathcal{D}} = \frac{n}{m+n}\mathcal{S} + \frac{m_2}{m+n}\widetilde{\mathcal{S}}_C + \frac{m_1}{m+n}\mathcal{D}'\,.$$

Note here that $\breve{\mathcal{D}}$ is a combination of the *empirical distribution* over correctly labeled data $S \cup \widetilde{S}_C$ and the (population) distribution over mislabeled data $\mathcal{D}'$. Recall that

$$\widehat{f} := \arg\min_{f \in \mathcal{F}} \mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}}(f)\,. \tag{11}$$

Since, $\widehat{f}$ minimizes 0-1 error on $S \cup \widetilde{S}$, using ERM optimality on (11), we have

$$\mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}}(f^*)\,. \tag{12}$$

Moreover, since f^* is independent of $\widetilde{S}_M$, using Hoeffding's bound, we have with probability at least $1-\delta$ that

$$\mathcal{E}_{\widetilde{\mathcal{S}}_M}(f^*) \leqslant \mathcal{E}_{\mathcal{D}'}(f^*) + \sqrt{\frac{\log(1/\delta)}{2m_1}}\,. \tag{13}$$

Finally, since f^* is the optimal classifier on $\breve{\mathcal{D}}$, we have

$$\mathcal{E}_{\breve{\mathcal{D}}}(f^*) \leqslant \mathcal{E}_{\breve{\mathcal{D}}}(\widehat{f})\,. \tag{14}$$

Now to relate (12) and (14), we multiply (13) by $\frac{m_1}{m+n}$ and add $\frac{n}{m+n}\mathcal{E}_{\mathcal{S}}(f) + \frac{m_2}{m+n}\mathcal{E}_{\widetilde{\mathcal{S}}_C}(f)$ both the sides. Hence, we can rewrite (13) as follows:

$$\mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}}(f^*) \leqslant \mathcal{E}_{\breve{\mathcal{D}}}(f^*) + \frac{m_1}{m+n}\sqrt{\frac{\log(1/\delta)}{2m_1}}\,. \tag{15}$$

Now we combine equations (12), (15), and (14), to get

$$\mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}}(\widehat{f}) \leqslant \mathcal{E}_{\breve{\mathcal{D}}}(\widehat{f}) + \frac{m_1}{m+n}\sqrt{\frac{\log(1/\delta)}{2m_1}}\,, \tag{16}$$

which implies

$$\mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) + \sqrt{\frac{\log(1/\delta)}{2m_1}}\,. \tag{17}$$

Since $\widetilde{S}$ is obtained by randomly labeling an unlabeled dataset, we assume $2m_1 \approx m$ [3]. Moreover, using $\mathcal{E}_{\mathcal{D}'} = 1 - \mathcal{E}_{\mathcal{D}}$ we obtain the desired result. □

Proof of Lemma 2. Recall $\mathcal{E}_{\widetilde{S}}(f) = \frac{m_1}{m}\mathcal{E}_{\widetilde{S}_M}(f) + \frac{m_2}{m}\mathcal{E}_{\widetilde{S}_C}(f)$. Hence, we have

$$2\mathcal{E}_{\widetilde{S}}(f) - \mathcal{E}_{\widetilde{S}_M}(f) - \mathcal{E}_{\widetilde{S}_C}(f) = \left(\frac{2m_1}{m}\mathcal{E}_{\widetilde{S}_M}(f) - \mathcal{E}_{\widetilde{S}_M}(f)\right) + \left(\frac{2m_2}{m}\mathcal{E}_{\widetilde{S}_C}(f) - \mathcal{E}_{\widetilde{S}_C}(f)\right) \tag{18}$$

$$= \left(\frac{2m_1}{m} - 1\right)\mathcal{E}_{\widetilde{S}_M}(f) + \left(\frac{2m_2}{m} - 1\right)\mathcal{E}_{\widetilde{S}_C}(f)\,. \tag{19}$$

[3] Formally, with probability at least $1-\delta$, we have $(m - 2m_1) \leqslant \sqrt{m\log(1/\delta)/2}$.

Since the dataset is labeled uniformly at random, with probability at least $1-\delta$, we have $\left(\frac{2m_1}{m}-1\right) \leqslant \sqrt{\frac{\log(1/\delta)}{2m}}$. Similarly, we have with probability at least $1-\delta$, $\left(\frac{2m_2}{m}-1\right) \leqslant \sqrt{\frac{\log(1/\delta)}{2m}}$. Using union bound, with probability at least $1-\delta$, we have

$$2\mathcal{E}_{\widetilde{S}} - \mathcal{E}_{\widetilde{S}_M}(f) - \mathcal{E}_{\widetilde{S}_C}(f) \leqslant \sqrt{\frac{\log(2/\delta)}{2m}} \left(\mathcal{E}_{\widetilde{S}_M}(f) + \mathcal{E}_{\widetilde{S}_C}(f)\right) . \tag{20}$$

With re-arranging $\mathcal{E}_{\widetilde{S}_M}(f) + \mathcal{E}_{\widetilde{S}_C}(f)$ and using the inequality $1-a \leqslant \frac{1}{1+a}$, we have

$$2\mathcal{E}_{\widetilde{S}} - \mathcal{E}_{\widetilde{S}_M}(f) - \mathcal{E}_{\widetilde{S}_C}(f) \leqslant 2\mathcal{E}_{\widetilde{S}} \sqrt{\frac{\log(2/\delta)}{2m}} . \tag{21}$$

□

Proof of Lemma 3. In the set of correctly labeled points $S \cup \widetilde{S}_C$, we have S as a random subset of $S \cup \widetilde{S}_C$. Hence, using Hoeffding's inequality for sampling without replacement (Lemma 5), we have with probability at least $1-\delta$

$$\mathcal{E}_{\widetilde{S}_C}(\widehat{f}) - \mathcal{E}_{\mathcal{S}\cup\widetilde{S}_C}(\widehat{f}) \leqslant \sqrt{\frac{\log(1/\delta)}{2m_2}} . \tag{22}$$

Re-writing $\mathcal{E}_{\mathcal{S}\cup\widetilde{S}_C}(\widehat{f})$ as $\frac{m_2}{m_2+n}\mathcal{E}_{\widetilde{S}_C}(\widehat{f}) + \frac{n}{m_2+n}\mathcal{E}_{\mathcal{S}}(\widehat{f})$, we have with probability at least $1-\delta$

$$\left(\frac{n}{n+m_2}\right)\left(\mathcal{E}_{\widetilde{S}_C}(\widehat{f}) - \mathcal{E}_{\mathcal{S}}(\widehat{f})\right) \leqslant \sqrt{\frac{\log(1/\delta)}{2m_2}} . \tag{23}$$

As before, assuming $2m_2 \approx m$, we have with probability at least $1-\delta$

$$\mathcal{E}_{\widetilde{S}_C}(\widehat{f}) - \mathcal{E}_{\mathcal{S}}(\widehat{f}) \leqslant \left(1+\frac{m_2}{n}\right)\sqrt{\frac{\log(1/\delta)}{m}} \leqslant \left(1+\frac{m}{2n}\right)\sqrt{\frac{\log(1/\delta)}{m}} . \tag{24}$$

□

Proof of Theorem 1. Having established these core intermediate results, we can now combine above three lemmas to prove the main result. In particular, we bound the population error on clean data ($\mathcal{E}_{\mathcal{D}}(\widehat{f})$) as follows:

(i) First, use (17), to obtain an upper bound on the population error on clean data, i.e., with probability at least $1-\delta/4$, we have

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant 1 - \mathcal{E}_{\widetilde{S}_M}(\widehat{f}) + \sqrt{\frac{\log(4/\delta)}{m}} . \tag{25}$$

(ii) Second, use (21), to relate the error on the mislabeled fraction with error on clean portion of randomly labeled data and error on whole randomly labeled dataset, i.e., with probability at least $1-\delta/2$, we have

$$-\mathcal{E}_{\widetilde{S}_M}(f) \leqslant \mathcal{E}_{\widetilde{S}_C}(f) - 2\mathcal{E}_{\widetilde{S}} + 2\mathcal{E}_{\widetilde{S}}\sqrt{\frac{\log(4/\delta)}{2m}} . \tag{26}$$

(iii) Finally, use (24) to relate the error on the clean portion of randomly labeled data and error on clean training data, i.e., with probability $1-\delta/4$, we have

$$\mathcal{E}_{\widetilde{S}_C}(\widehat{f}) \leqslant -\mathcal{E}_{\mathcal{S}}(\widehat{f}) + \left(1+\frac{m}{2n}\right)\sqrt{\frac{\log(4/\delta)}{m}} . \tag{27}$$

Using union bound on the above three steps, we have with probability at least $1-\delta$:

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}}(\widehat{f}) + 1 - 2\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f}) + \left(\sqrt{2}\mathcal{E}_{\widetilde{S}} + 2 + \frac{m}{2n}\right)\sqrt{\frac{\log(4/\delta)}{m}} . \tag{28}$$

□

A.2. Proof of Proposition 1

Proof of Proposition 1. For a classifier $f : \mathcal{X} \to \{-1, 1\}$, we have $1 - 2\,\mathbb{I}\left[f(x) \neq y\right] = y \cdot f(x)$. Hence, by definition of $\mathcal{E}$, we have

$$1 - 2\mathcal{E}_{\widetilde{S}}(f) = \frac{1}{m}\sum_{i=1}^{m} y_i \cdot f(x_i) \leqslant \sup_{f\in\mathcal{F}} \frac{1}{m}\sum_{i=1}^{m} y_i \cdot f(x_i)\,. \tag{29}$$

Note that for fixed inputs $(x_1, x_2, \ldots, x_m)$ in $\widetilde{S}$, $(y_1, y_2, \ldots y_m)$ are random labels. Define $\phi_1(y_1, y_2, \ldots, y_m) := \sup_{f\in\mathcal{F}} \frac{1}{m}\sum_{i=1}^{m} y_i \cdot f(x_i)$. We have the following bounded difference condition on ϕ_1. For all i,

$$\sup_{y_1,\ldots y_m, y_i'\in\{-1,1\}^{m+1}} \left|\phi_1(y_1, \ldots, y_i, \ldots, y_m) - \phi_1(y_1, \ldots, y_i', \ldots, y_m)\right| \leqslant 1/m\,. \tag{30}$$

Similarly, we define $\phi_2(x_1, x_2, \ldots, x_m) := \mathbb{E}_{y_i\sim U\{-1,1\}}\left[\sup_{f\in\mathcal{F}} \frac{1}{m}\sum_{i=1}^{m} y_i \cdot f(x_i)\right]$. We have the following bounded difference condition on ϕ_2. For all i,

$$\sup_{x_1,\ldots x_m, x_i'\in\mathcal{X}^{m+1}} \left|\phi_2(x_1, \ldots, x_i, \ldots, x_m) - \phi_1(x_1, \ldots, x_i', \ldots, x_m)\right| \leqslant 1/m\,. \tag{31}$$

Using McDiarmid's inequality (Lemma 6) twice with Condition (30) and (31), with probability at least $1 - \delta$, we have

$$\sup_{f\in\mathcal{F}} \frac{1}{m}\sum_{i=1}^{m} y_i \cdot f(x_i) - \mathbb{E}_{x,y}\left[\sup_{f\in\mathcal{F}} \frac{1}{m}\sum_{i=1}^{m} y_i \cdot f(x_i)\right] \leqslant \sqrt{\frac{2\log(2/\delta)}{m}}\,. \tag{32}$$

Combining (29) and (32), we obtain the desired result. □

A.3. Proof of Theorem 2

Proof of Theorem 2 follows similar to the proof of Theorem 1. Note that the same results in Lemma 1, Lemma 2, and Lemma 3 hold in the regularized ERM case. However, the arguments in the proof of Lemma 1 change slightly. Hence, we state the lemma for regularized ERM and prove it here for completeness.

Lemma 7. *Assume the same setup as Theorem 2. Then for any $\delta > 0$, with probability at least $1 - \delta$ over the random draws of mislabeled data $\widetilde{S}_M$, we have*

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant 1 - \mathcal{E}_{\widetilde{S}_M}(\widehat{f}) + \sqrt{\frac{\log(1/\delta)}{m}}\,. \tag{33}$$

Proof. The main idea of the proof remains the same, i.e. regard the clean portion of the data ($S \cup \widetilde{S}_C$) as fixed. Then, there exists a classifier f^* that is optimal over draws of the mislabeled data $\widetilde{S}_M$.

Formally,

$$f^* := \arg\min_{f\in\mathcal{F}} \mathcal{E}_{\breve{\mathcal{D}}}(f) + \lambda R(f)\,, \tag{34}$$

where

$$\breve{\mathcal{D}} = \frac{n}{m+n}\mathcal{S} + \frac{m_1}{m+n}\widetilde{\mathcal{S}}_C + \frac{m_2}{m+n}\mathcal{D}'\,.$$

That is, $\breve{\mathcal{D}}$ a combination of the *empirical distribution* over correctly labeled data $S \cup \widetilde{S}_C$ and the (population) distribution over mislabeled data $\mathcal{D}'$. Recall that

$$\widehat{f} := \arg\min_{f\in\mathcal{F}} \mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}}(f) + \lambda R(f)\,. \tag{35}$$

Since, $\widehat{f}$ minimizes 0-1 error on $S \cup \widetilde{S}$, using ERM optimality on (11), we have

$$\mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}}(\widehat{f}) + \lambda R(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}}(f^*) + \lambda R(f^*)\,. \tag{36}$$

Moreover, since f^* is independent of $\widetilde{S}_M$, using Hoeffding's bound, we have with probability at least $1-\delta$ that

$$\mathcal{E}_{\widetilde{S}_M}(f^*) \leqslant \mathcal{E}_{\mathcal{D}'}(f^*) + \sqrt{\frac{\log(1/\delta)}{2m_1}}\,. \tag{37}$$

Finally, since f^* is the optimal classifier on $\breve{\mathcal{D}}$, we have

$$\mathcal{E}_{\breve{\mathcal{D}}}(f^*) + \lambda R(f^*) \leqslant \mathcal{E}_{\breve{\mathcal{D}}}(\widehat{f}) + \lambda R(\widehat{f})\,. \tag{38}$$

Now to relate (36) and (38), we can re-write the (37) as follows:

$$\mathcal{E}_{S\cup\widetilde{\mathcal{S}}}(f^*) \leqslant \mathcal{E}_{\breve{\mathcal{D}}}(f^*) + \frac{m_1}{m+n}\sqrt{\frac{\log(1/\delta)}{2m_1}}\,. \tag{39}$$

After adding $\lambda R(f^*)$ on both sides in (39), we combine equations (36), (39), and (38), to get

$$\mathcal{E}_{S\cup\widetilde{\mathcal{S}}}(\widehat{f}) \leqslant \mathcal{E}_{\breve{\mathcal{D}}}(\widehat{f}) + \frac{m_1}{m+n}\sqrt{\frac{\log(1/\delta)}{2m_1}}\,, \tag{40}$$

which implies

$$\mathcal{E}_{\widetilde{S}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) + \sqrt{\frac{\log(1/\delta)}{2m_1}}\,. \tag{41}$$

Similar as before, since $\widetilde{S}$ is obtained by randomly labeling an unlabeled dataset, we assume $2m_1 \approx m$. Moreover, using $\mathcal{E}_{\mathcal{D}'} = 1 - \mathcal{E}_{\mathcal{D}}$ we obtain the desired result. □

A.4. Proof of Theorem 3

To prove our results in the multiclass case, we first state and prove lemmas parallel to those used in the proof of balanced binary case. We then combine these results to obtain the result in Theorem 3.

Before stating the result, we define mislabeled distribution $\mathcal{D}'$ for any $\mathcal{D}$. While $\mathcal{D}'$ and $\mathcal{D}$ share the same marginal distribution over inputs $\mathcal{X}$, the conditional distribution over labels y given an input $x \sim \mathcal{D}_{\mathcal{X}}$ is changed as follows: For any x, the Probability Mass Function (PMF) over y is defined as: $p_{\mathcal{D}'}(\cdot|x) := \frac{1-p_{\mathcal{D}}(\cdot|x)}{k-1}$, where $p_{\mathcal{D}}(\cdot|x)$ is the PMF over y for the distribution $\mathcal{D}$.

Lemma 8. *Assume the same setup as Theorem 3. Then for any $\delta > 0$, with probability at least $1-\delta$ over the random draws of mislabeled data $\widetilde{S}_M$, we have*

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant (k-1)\left(1 - \mathcal{E}_{\widetilde{S}_M}(\widehat{f})\right) + (k-1)\sqrt{\frac{\log(1/\delta)}{m}}\,. \tag{42}$$

Proof. The main idea of the proof remains the same. We begin by regarding the clean portion of the data ($S \cup \widetilde{S}_C$) as fixed. Then, there exists a classifier f^* that is optimal over draws of the mislabeled data $\widetilde{S}_M$.

However, in the multiclass case, we cannot as easily relate the population error on mislabeled data to the population accuracy on clean data. While for binary classification, we could lower bound the population accuracy $1-\mathcal{E}_{\mathcal{D}}$ with the empirical error on mislabeled data $\mathcal{E}_{\widetilde{S}_M}$ (in the proof of Lemma 1), for multiclass classification, error on the mislabeled data and accuracy on the clean data in the population are not so directly related. To establish (42), we break the error on the (unknown) mislabeled data into two parts: one term corresponds to predicting the true label on mislabeled data, and the other corresponds to predicting neither the true label nor the assigned (mis-)label. Finally, we relate these errors to their population counterparts to establish (42).

Formally,

$$f^* := \arg\min_{f\in\mathcal{F}} \mathcal{E}_{\breve{\mathcal{D}}}(f) + \lambda R(f)\,, \tag{43}$$

where

$$\breve{\mathcal{D}} = \frac{n}{m+n}\mathcal{S} + \frac{m_1}{m+n}\widetilde{\mathcal{S}}_C + \frac{m_2}{m+n}\mathcal{D}' \,.$$

That is, $\breve{\mathcal{D}}$ is a combination of the *empirical distribution* over correctly labeled data $S \cup \widetilde{S}_C$ and the (population) distribution over mislabeled data $\mathcal{D}'$. Recall that

$$\widehat{f} := \arg\min_{f \in \mathcal{F}} \mathcal{E}_{\mathcal{S} \cup \widetilde{S}}(f) + \lambda R(f) \,. \tag{44}$$

Following the exact steps from the proof of Lemma 7, with probability at least $1-\delta$, we have

$$\mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) + \sqrt{\frac{\log(1/\delta)}{2m_1}} \,. \tag{45}$$

Similar to before, since $\widetilde{S}$ is obtained by randomly labeling an unlabeled dataset, we assume $\frac{k}{k-1}m_1 \approx m$.

Now we will relate $\mathcal{E}_{\mathcal{D}'}(\widehat{f})$ with $\mathcal{E}_{\mathcal{D}}(\widehat{f})$. Let y^T denote the (unknown) true label for a mislabeled point (x, y) (i.e., label before replacing it with a mislabel).

$$\begin{aligned}\mathbb{E}_{(x,y)\in\sim\mathcal{D}'}\left[\mathbb{I}\left[\widehat{f}(x) \neq y\right]\right] = \underbrace{\mathbb{E}_{(x,y)\in\sim\mathcal{D}'}\left[\mathbb{I}\left[\widehat{f}(x) \neq y \wedge \widehat{f}(x) \neq y^T\right]\right]}_{\text{I}} \\ + \underbrace{\mathbb{E}_{(x,y)\in\sim\mathcal{D}'}\left[\mathbb{I}\left[\widehat{f}(x) \neq y \wedge \widehat{f}(x) = y^T\right]\right]}_{\text{II}} \,.\end{aligned} \tag{46}$$

Clearly, term 2 is one minus the accuracy on the clean unseen data, i.e.,

$$\text{II} = 1 - \mathbb{E}_{x,y\sim\mathcal{D}}\left[\mathbb{I}\left[\widehat{f}(x) \neq y\right]\right] = 1 - \mathcal{E}_{\mathcal{D}}(\widehat{f}) \,. \tag{47}$$

Next, we relate term 1 with the error on the unseen clean data. We show that term 1 is equal to the error on the unseen clean data scaled by $\frac{k-2}{k-1}$, where k is the number of labels. Using the definition of mislabeled distribution $\mathcal{D}'$, we have

$$\text{I} = \frac{1}{k-1}\left(\mathbb{E}_{(x,y)\in\sim\mathcal{D}}\left[\sum_{i\in\mathcal{Y}\wedge i\neq y}\mathbb{I}\left[\widehat{f}(x) \neq i \wedge \widehat{f}(x) \neq y\right]\right]\right) = \frac{k-2}{k-1}\mathcal{E}_{\mathcal{D}}(\widehat{f}) \,. \tag{48}$$

Combining the result in (47), (48) and (46), we have

$$\mathcal{E}_{\mathcal{D}'}(\widehat{f}) = 1 - \frac{1}{k-1}\mathcal{E}_{\mathcal{D}}(\widehat{f}) \,. \tag{49}$$

Finally, combining the result in (49) with equation (45), we have with probability $1-\delta$,

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant (k-1)\left(1 - \mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f})\right) + (k-1)\sqrt{\frac{k\log(1/\delta)}{2(k-1)m}} \,. \tag{50}$$

□

Lemma 9. *Assume the same setup as Theorem 3. Then for any $\delta > 0$, with probability at least $1-\delta$ over the random draws of $\widetilde{S}$, we have*

$$\left|k\mathcal{E}_{\widetilde{S}}(\widehat{f}) - \mathcal{E}_{\widetilde{S}_C}(\widehat{f}) - (k-1)\mathcal{E}_{\widetilde{S}_M}(\widehat{f})\right| \leqslant 2k\sqrt{\frac{\log(4/\delta)}{2m}} \,.$$

Proof. Recall $\mathcal{E}_{\widetilde{S}}(f) = \frac{m_1}{m}\mathcal{E}_{\widetilde{S}_M}(f) + \frac{m_2}{m}\mathcal{E}_{\widetilde{S}_C}(f)$. Hence, we have

$$\begin{aligned} k\mathcal{E}_{\widetilde{S}}(f) - (k-1)\mathcal{E}_{\widetilde{S}_M}(f) - \mathcal{E}_{\widetilde{S}_C}(f) &= (k-1)\left(\frac{km_1}{(k-1)m}\mathcal{E}_{\widetilde{S}_M}(f) - \mathcal{E}_{\widetilde{S}_M}(f)\right) \\ &\quad + \left(\frac{km_2}{m}\mathcal{E}_{\widetilde{S}_C}(f) - \mathcal{E}_{\widetilde{S}_C}(f)\right) \\ &= k\left[\left(\frac{m_1}{m} - \frac{k-1}{k}\right)\mathcal{E}_{\widetilde{S}_M}(f) + \left(\frac{m_2}{m} - \frac{1}{k}\right)\mathcal{E}_{\widetilde{S}_C}(f)\right]. \end{aligned}$$

Since the dataset is randomly labeled, we have with probability at least $1-\delta$, $\left(\frac{m_1}{m} - \frac{k-1}{k}\right) \leqslant \sqrt{\frac{\log(1/\delta)}{2m}}$. Similarly, we have with probability at least $1-\delta$, $\left(\frac{m_2}{m} - \frac{1}{k}\right) \leqslant \sqrt{\frac{\log(1/\delta)}{2m}}$. Using union bound, we have with probability at least $1-\delta$

$$k\mathcal{E}_{\widetilde{S}}(f) - (k-1)\mathcal{E}_{\widetilde{S}_M}(f) - \mathcal{E}_{\widetilde{S}_C}(f) \leqslant k\sqrt{\frac{\log(2/\delta)}{2m}}\left(\mathcal{E}_{\widetilde{S}_M}(f) + \mathcal{E}_{\widetilde{S}_C}(f)\right). \tag{51}$$

□

Lemma 10. *Assume the same setup as Theorem 3. Then for any $\delta > 0$, with probability at least $1-\delta$ over the random draws of $\widetilde{S}_C$ and S, we have*

$$\left|\mathcal{E}_{\widetilde{S}_C}(\widehat{f}) - \mathcal{E}_{\mathcal{S}}(\widehat{f})\right| \leqslant 1.5\sqrt{\frac{k\log(2/\delta)}{2m}}.$$

Proof. In the set of correctly labeled points $S \cup \widetilde{S}_C$, we have S as a random subset of $S \cup \widetilde{S}_C$. Hence, using Hoeffding's inequality for sampling without replacement (Lemma 5), we have with probability at least $1-\delta$

$$\mathcal{E}_{\widetilde{\mathcal{S}}_c}(\widehat{f}) - \mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}_C}(\widehat{f}) \leqslant \sqrt{\frac{\log(1/\delta)}{2m_2}}. \tag{52}$$

Re-writing $\mathcal{E}_{\mathcal{S}\cup\widetilde{\mathcal{S}}_C}(\widehat{f})$ as $\frac{m_2}{m_2+n}\mathcal{E}_{\widetilde{\mathcal{S}}_C}(\widehat{f}) + \frac{n}{m_2+n}\mathcal{E}_{\mathcal{S}}(\widehat{f})$, we have with probability at least $1-\delta$

$$\left(\frac{n}{n+m_2}\right)\left(\mathcal{E}_{\widetilde{\mathcal{S}}_c}(\widehat{f}) - \mathcal{E}_{\mathcal{S}}(\widehat{f})\right) \leqslant \sqrt{\frac{\log(1/\delta)}{2m_2}}. \tag{53}$$

As before, assuming $km_2 \approx m$, we have with probability at least $1-\delta$

$$\mathcal{E}_{\widetilde{\mathcal{S}}_c}(\widehat{f}) - \mathcal{E}_{\mathcal{S}}(\widehat{f}) \leqslant \left(1 + \frac{m_2}{n}\right)\sqrt{\frac{k\log(1/\delta)}{2m}} \leqslant \left(1 + \frac{1}{k}\right)\sqrt{\frac{k\log(1/\delta)}{2m}}. \tag{54}$$

□

Proof of Theorem 3. Having established these core intermediate results, we can now combine above three lemmas. In particular, we bound the population error on clean data ($\mathcal{E}_{\mathcal{D}}(\widehat{f})$) as follows:

(i) First, use (50), to obtain an upper bound on the population error on clean data, i.e., with probability at least $1-\delta/4$, we have

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant (k-1)\left(1 - \mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f})\right) + (k-1)\sqrt{\frac{k\log(4/\delta)}{2(k-1)m}}. \tag{55}$$

(ii) Second, use (51) to relate the error on the mislabeled fraction with error on clean portion of randomly labeled data and error on whole randomly labeled dataset, i.e., with probability at least $1-\delta/2$, we have

$$-(k-1)\mathcal{E}_{\widetilde{S}_M}(f) \leqslant \mathcal{E}_{\widetilde{S}_C}(f) - k\mathcal{E}_{\widetilde{S}} + k\sqrt{\frac{\log(4/\delta)}{2m}}. \tag{56}$$

(iii) Finally, use (54) to relate the error on the clean portion of randomly labeled data and error on clean training data, i.e., with probability $1 - \delta/4$, we have

$$\mathcal{E}_{\widetilde{S}_C}(\widehat{f}) \leqslant -\mathcal{E}_{\mathcal{S}}(\widehat{f}) + \left(1 + \frac{m}{kn}\right)\sqrt{\frac{k\log(4/\delta)}{2m}}\,. \tag{57}$$

Using union bound on the above three steps, we have with probability at least $1 - \delta$:

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}}(\widehat{f}) + (k-1) - k\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f}) + (\sqrt{k(k-1)} + k + \sqrt{k} + \frac{m}{n\sqrt{k}})\sqrt{\frac{\log(4/\delta)}{2m}}\,. \tag{58}$$

Simplifying the term in RHS of (58), we get the desired result. in the final bound. □

B. Proofs from Sec. 4

We suppose that the parameters of the linear function are obtained via gradient descent on the following L_2 regularized problem:

$$\mathcal{L}_S(w;\lambda) := \sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2 \,, \tag{59}$$

where $\lambda \geqslant 0$ is a regularization parameter. We assume access to a clean dataset $S = \{(x_i, y_i)\}_{i=1}^{n} \sim \mathcal{D}^n$ and randomly labeled dataset $\widetilde{S} = \{(x_i, y_i)\}_{i=n+1}^{n+m} \sim \widetilde{\mathcal{D}}^m$. Let $\boldsymbol{X} = [x_1, x_2, \cdots, x_{m+n}]$ and $\boldsymbol{y} = [y_1, y_2, \cdots, y_{m+n}]$. Fix a positive learning rate η such that $\eta \leqslant 1/\left(\left\|\boldsymbol{X}^T\boldsymbol{X}\right\|_{\text{op}} + \lambda^2\right)$ and an initialization $w_0 = 0$. Consider the following gradient descent iterates to minimize objective (59) on $S \cup \widetilde{S}$:

$$w_t = w_{t-1} - \eta \nabla_w \mathcal{L}_{S\cup\widetilde{S}}(w_{t-1};\lambda) \quad \forall t = 1, 2, \ldots \tag{60}$$

Then we have $\{w_t\}$ converge to the limiting solution $\widehat{w} = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{X}^T \boldsymbol{y}$. Define $\widehat{f}(x) := f(x;\widehat{w})$.

B.1. Proof of Theorem 4

We use a standard result from linear algebra, namely the Shermann-Morrison formula (Sherman & Morrison, 1950) for matrix inversion:

Lemma 11 (Sherman & Morrison (1950))**.** *Suppose $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is an invertible square matrix and $u, v \in \mathbb{R}^n$ are column vectors. Then $\boldsymbol{A} + uv^T$ is invertible iff $1 + v^T\boldsymbol{A}u \neq 0$ and in particular*

$$(\boldsymbol{A} + uv^T)^{-1} = \boldsymbol{A}^{-1} - \frac{\boldsymbol{A}^{-1}uv^T\boldsymbol{A}^{-1}}{1 + v^T\boldsymbol{A}^{-1}u} \,. \tag{61}$$

For a given training set $S \cup \widetilde{S}_C$, define leave-one-out error on mislabeled points in the training data as

$$\mathcal{E}_{\text{LOO}(\widetilde{S}_M)} = \frac{\sum_{(x_i,y_i)\in\widetilde{S}_M} \mathcal{E}(f_{(i)}(x_i), y_i)}{\left|\widetilde{S}_M\right|} \,,$$

where $f_{(i)} := f(\mathcal{A}, (S \cup \widetilde{S})_{(i)})$. To relate empirical leave-one-out error and population error with hypothesis stability condition, we use the following lemma:

Lemma 12 (Bousquet & Elisseeff (2002))**.** *For the leave-one-out error, we have*

$$\mathbb{E}\left[\left(\mathcal{E}_{\mathcal{D}'}(\widehat{f}) - \mathcal{E}_{LOO(\widetilde{S}_M)}\right)^2\right] \leqslant \frac{1}{2m_1} + \frac{3\beta}{n+m} \,. \tag{62}$$

Proof of the above lemma is similar to the proof of Lemma 9 in Bousquet & Elisseeff (2002) and can be found in App. D. Before presenting the proof of Theorem 4, we introduce some more notation. Let $\boldsymbol{X}_{(i)}$ denote the matrix of covariates with the i^{th} point removed. Similarly, let $\boldsymbol{y}_{(i)}$ be the array of responses with the i^{th} point removed. Define the corresponding regularized GD solution as $\widehat{w}_{(i)} = \left(\boldsymbol{X}_{(i)}^T\boldsymbol{X}_{(i)} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{X}_{(i)}^T\boldsymbol{y}_{(i)}$. Define $\widehat{f}_{(i)}(x) := f(x;\widehat{w}_{(i)})$.

Proof of Theorem 4. Because squared loss minimization does not imply 0-1 error minimization, we cannot use arguments from Lemma 1. This is the main technical difficulty. To compare the 0-1 error at a train point with an unseen point, we use the closed-form expression for $\widehat{w}$ and Shermann-Morrison formula to upper bound training error with leave-one-out cross validation error.

The proof is divided into three parts: In part one, we show that 0-1 error on mislabeled points in the training set is lower than the error obtained by leave-one-out error at those points. In part two, we relate this leave-one-out error with the population error on mislabeled distribution using Condition 1. While the empirical leave-one-out error is an unbiased estimator of the average population error of leave-one-out classifiers, we need hypothesis stability to control the variance of empirical

leave-one-out error. Finally, in part three, we show that the error on the mislabeled training points can be estimated with just the randomly labeled and clean training data (as in proof of Theorem 1).

Part 1 First we relate training error with leave-one-out error. For any training point (x_i, y_i) in $\widetilde{S} \cup S$, we have

$$\mathcal{E}(\widehat{f}(x_i), y_i) = \mathbb{I}\left[y_i \cdot x_i^T \widehat{w} < 0\right] = \mathbb{I}\left[y_i \cdot x_i^T \left(\boldsymbol{X}^T \boldsymbol{X} + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{X}^T \boldsymbol{y} < 0\right] \tag{63}$$

$$= \mathbb{I}\left[y_i \cdot x_i^T \underbrace{\left(\boldsymbol{X}_{(i)}^T \boldsymbol{X}_{(i)} + x_i^T x_i + \lambda \boldsymbol{I}\right)^{-1}}_{\mathrm{I}} (\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{(i)} + y_i \cdot x_i) < 0\right]. \tag{64}$$

Letting $\boldsymbol{A} = \left(\boldsymbol{X}_{(i)}^T \boldsymbol{X}_{(i)} + \lambda \boldsymbol{I}\right)$ and using Lemma 11 on term 1, we have

$$\mathcal{E}(\widehat{f}(x_i), y_i) = \mathbb{I}\left[y_i \cdot x_i^T \left[\boldsymbol{A}^{-1} - \frac{\boldsymbol{A}^{-1} x_i x_i^T \boldsymbol{A}^{-1}}{1 + x_i^T \boldsymbol{A}^{-1} x_i}\right] (\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{(i)} + y_i \cdot x_i) < 0\right] \tag{65}$$

$$= \mathbb{I}\left[y_i \cdot \left[\frac{x_i^T \boldsymbol{A}^{-1}(1 + x_i^T \boldsymbol{A}^{-1} x_i) - x_i^T \boldsymbol{A}^{-1} x_i x_i^T \boldsymbol{A}^{-1}}{1 + x_i^T \boldsymbol{A}^{-1} x_i}\right] (\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{(i)} + y_i \cdot x_i) < 0\right] \tag{66}$$

$$= \mathbb{I}\left[y_i \cdot \left[\frac{x_i^T \boldsymbol{A}^{-1}}{1 + x_i^T \boldsymbol{A}^{-1} x_i}\right] (\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{(i)} + y_i \cdot x_i) < 0\right]. \tag{67}$$

Since $1 + x_i^T \boldsymbol{A}^{-1} x_i > 0$, we have

$$\mathcal{E}(\widehat{f}(x_i), y_i) = \mathbb{I}\left[y_i \cdot x_i^T \boldsymbol{A}^{-1} (\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{(i)} + y_i \cdot x_i) < 0\right] \tag{68}$$

$$= \mathbb{I}\left[x_i^T \boldsymbol{A}^{-1} x_i + y_i \cdot x_i^T \boldsymbol{A}^{-1} (\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{(i)}) < 0\right] \tag{69}$$

$$\leqslant \mathbb{I}\left[y_i \cdot x_i^T \boldsymbol{A}^{-1} (\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{(i)}) < 0\right] = \mathcal{E}(\widehat{f}_{(i)}(x_i), y_i). \tag{70}$$

Using (70), we have

$$\mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathrm{LOO}(\widetilde{S}_M)} := \frac{\sum_{(x_i, y_i) \in \widetilde{S}_M} \mathcal{E}(\widehat{f}_{(i)}(x_i), y_i)}{\left|\widetilde{\mathcal{S}}_M\right|}. \tag{71}$$

Part 2 We now relate RHS in (71) with the population error on mislabeled distribution. To do this, we leverage Condition 1 and Lemma 12. In particular, we have

$$\mathbb{E}_{\mathcal{S} \cup \widetilde{\mathcal{S}}_M}\left[\left(\mathcal{E}_{\mathcal{D}'}(\widehat{f}) - \mathcal{E}_{\mathrm{LOO}(\widetilde{S}_M)}\right)^2\right] \leqslant \frac{1}{2m_1} + \frac{3\beta}{m+n}. \tag{72}$$

Using Chebyshev's inequality, with probability at least $1 - \delta$, we have

$$\mathcal{E}_{\mathrm{LOO}(\widetilde{S}_M)} \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) + \sqrt{\frac{1}{\delta}\left(\frac{1}{2m_1} + \frac{3\beta}{m+n}\right)}. \tag{73}$$

Part 3 Combining (73) and (71), we have

$$\mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) + \sqrt{\frac{1}{\delta}\left(\frac{1}{2m_1} + \frac{3\beta}{m+n}\right)}. \tag{74}$$

Compare (74) with (17) in the proof of Lemma 1. We obtain a similar relationship between $\mathcal{E}_{\widetilde{\mathcal{S}}_M}$ and $\mathcal{E}_{\mathcal{D}'}$ but with a polynomial concentration instead of exponential concentration. In addition, since we just use concentration arguments to relate mislabeled error to the errors on the clean and unlabeled portions of the randomly labeled data, we can directly use the results in Lemma 2 and Lemma 3. Therefore, combining results in Lemma 2, Lemma 3, and (74) with union bound, we have with probability at least $1-\delta$

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}}(\widehat{f}) + 1 - 2\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f}) + \left(\sqrt{2}\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f}) + 1 + \frac{m}{2n}\right)\sqrt{\frac{\log(4/\delta)}{m}} + \sqrt{\frac{4}{\delta}\left(\frac{1}{m} + \frac{3\beta}{m+n}\right)}. \tag{75}$$

□

B.2. Extension to multiclass classification

For multiclass problems with squared loss minimization, as standard practice, we consider one-hot encoding for the underlying label, i.e., a class label $c \in [k]$ is treated as $(0, \cdot, 0, 1, 0, \cdot, 0) \in \mathbb{R}^k$ (with c-th coordinate being 1). As before, we suppose that the parameters of the linear function are obtained via gradient descent on the following L_2 regularized problem:

$$\mathcal{L}_S(w;\lambda) := \sum_{i=1}^{n} \left\| w^T x_i - y_i \right\|_2^2 + \lambda \sum_{j=1}^{k} \|w_j\|_2^2 \,, \tag{76}$$

where $\lambda \geqslant 0$ is a regularization parameter. We assume access to a clean dataset $S = \{(x_i, y_i)\}_{i=1}^{n} \sim \mathcal{D}^n$ and randomly labeled dataset $\widetilde{S} = \{(x_i, y_i)\}_{i=n+1}^{n+m} \sim \widetilde{\mathcal{D}}^m$. Let $\boldsymbol{X} = [x_1, x_2, \cdots, x_{m+n}]$ and $\boldsymbol{y} = [e_{y_1}, e_{y_2}, \cdots, e_{y_{m+n}}]$. Fix a positive learning rate η such that $\eta \leqslant 1/\left(\left\|\boldsymbol{X}^T\boldsymbol{X}\right\|_{\text{op}} + \lambda^2\right)$ and an initialization $w_0 = 0$. Consider the following gradient descent iterates to minimize objective (59) on $S \cup \widetilde{S}$:

$$w_j{}^t = w_j{}^{t-1} - \eta \nabla_{w_j} \mathcal{L}_{S\cup\widetilde{S}}(w^{t-1};\lambda) \quad \forall t = 1, 2, \ldots \text{ and } j = 1, 2, \ldots, k\,. \tag{77}$$

Then we have $\{w_j{}^t\}$ for all $j = 1, 2, \cdots, k$ converge to the limiting solution $\widehat{w}_j = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda \boldsymbol{I}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}_j$. Define $\widehat{f}(x) := f(x;\widehat{w})$.

Theorem 5. *Assume that this gradient descent algorithm satisfies Condition 1 with $\beta = \mathcal{O}(1)$. Then for a multiclass classification problem wth k classes, for any $\delta > 0$, with probability at least $1-\delta$, we have:*

$$\begin{aligned}\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}}(\widehat{f}) &+ (k-1)\left(1 - \frac{k}{k-1}\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f})\right) \\ &+ \left(k + \sqrt{k} + \frac{m}{n\sqrt{k}}\right)\sqrt{\frac{\log(4/\delta)}{2m}} + \sqrt{k(k-1)}\sqrt{\frac{4}{\delta}\left(\frac{1}{m} + \frac{3\beta}{m+n}\right)}.\end{aligned} \tag{78}$$

Proof. The proof of this theorem is divided into two parts. In the first part, we relate the error on the mislabeled samples with the population error on the mislabeled data. Similar to the proof of Theorem 4, we use Shermann-Morrison formula to upper bound training error with leave-one-out error on each $\widehat{w}^j$. Second part of the proof follows entirely from the proof of Theorem 3. In essence, the first part derives an equivalent of (45) for GD training with squared loss and then the second part follows from the proof of Theorem 3.

Part-1: Consider a training point (x_i, y_i) in $\widetilde{S} \cup S$. For simplicity, we use c_i to denote the class of i-th point and use y_i as the corresponding one-hot embedding. Recall error in multiclass point is given by $\mathcal{E}(\widehat{f}(x_i), y_i) = \mathbb{I}\left[c_i \notin \arg\max x_i^T \widehat{w}\right]$. Thus, there exists a $j \neq c_i \in [k]$, such that we have

$$\mathcal{E}(\widehat{f}(x_i), y_i) = \mathbb{I}\left[c_i \notin \arg\max x_i^T\widehat{w}\right] = \mathbb{I}\left[x_i^T\widehat{w}_{c_i} < x_i^T\widehat{w}_j\right] \tag{79}$$

$$= \mathbb{I}\left[x_i^T\left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}_{c_i} < x_i^T\left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}_j\right] \tag{80}$$

$$= \mathbb{I}\left[x_i^T\underbrace{\left(\boldsymbol{X}_{(i)}^T\boldsymbol{X}_{(i)} + x_i^Tx_i + \lambda\boldsymbol{I}\right)^{-1}}_{\text{I}}\left(\boldsymbol{X}_{(i)}^T\boldsymbol{y}_{c_i\,(i)} + x_i - \boldsymbol{X}_{(i)}^T\boldsymbol{y}_{j\,(i)}\right) < 0\right]. \tag{81}$$

Letting $\boldsymbol{A} = \left(\boldsymbol{X}_{(i)}^T \boldsymbol{X}_{(i)} + \lambda \boldsymbol{I}\right)$ and using Lemma 11 on term 1, we have

$$\mathcal{E}(\widehat{f}(x_i), y_i) = \mathbb{I}\left[x_i^T \left[\boldsymbol{A}^{-1} - \frac{\boldsymbol{A}^{-1} x_i x_i^T \boldsymbol{A}^{-1}}{1 + x_i^T \boldsymbol{A}^{-1} x_i}\right] \left(\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{c_i(i)} + x_i - \boldsymbol{X}_{(i)}^T \boldsymbol{y}_{j(i)}\right) < 0\right] \tag{82}$$

$$= \mathbb{I}\left[\left[\frac{x_i^T \boldsymbol{A}^{-1}(1 + x_i^T \boldsymbol{A}^{-1} x_i) - x_i^T \boldsymbol{A}^{-1} x_i x_i^T \boldsymbol{A}^{-1}}{1 + x_i^T \boldsymbol{A}^{-1} x_i}\right] \left(\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{c_i(i)} + x_i - \boldsymbol{X}_{(i)}^T \boldsymbol{y}_{j(i)}\right) < 0\right] \tag{83}$$

$$= \mathbb{I}\left[\left[\frac{x_i^T \boldsymbol{A}^{-1}}{1 + x_i^T \boldsymbol{A}^{-1} x_i}\right] \left(\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{c_i(i)} + x_i - \boldsymbol{X}_{(i)}^T \boldsymbol{y}_{j(i)}\right) < 0\right]. \tag{84}$$

Since $1 + x_i^T \boldsymbol{A}^{-1} x_i > 0$, we have

$$\mathcal{E}(\widehat{f}(x_i), y_i) = \mathbb{I}\left[x_i^T \boldsymbol{A}^{-1} \left(\boldsymbol{X}_{(i)}^T \boldsymbol{y}_{c_i(i)} + x_i - \boldsymbol{X}_{(i)}^T \boldsymbol{y}_{j(i)}\right) < 0\right] \tag{85}$$

$$= \mathbb{I}\left[x_i^T \boldsymbol{A}^{-1} x_i + x_i^T \boldsymbol{A}^{-1} \boldsymbol{X}_{(i)}^T \boldsymbol{y}_{c_i(i)} - x_i^T \boldsymbol{A}^{-1} \boldsymbol{X}_{(i)}^T \boldsymbol{y}_{j(i)} < 0\right] \tag{86}$$

$$\leqslant \mathbb{I}\left[x_i^T \boldsymbol{A}^{-1} \boldsymbol{X}_{(i)}^T \boldsymbol{y}_{c_i(i)} - x_i^T \boldsymbol{A}^{-1} \boldsymbol{X}_{(i)}^T \boldsymbol{y}_{j(i)} < 0\right] = \mathcal{E}(\widehat{f}_{(i)}(x_i), y_i). \tag{87}$$

Using (87), we have

$$\mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathrm{LOO}(\widetilde{S}_M)} := \frac{\sum_{(x_i,y_i)\in\widetilde{S}_M} \mathcal{E}(\widehat{f}_{(i)}(x_i), y_i)}{\left|\widetilde{\mathcal{S}}_M\right|}. \tag{88}$$

We now relate RHS in (71) with the population error on mislabeled distribution. Similar as before, to do this, we leverage Condition 1 and Lemma 12. Using (73) and (88), we have

$$\mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) + \sqrt{\frac{1}{\delta}\left(\frac{1}{2m_1} + \frac{3\beta}{m+n}\right)}. \tag{89}$$

We have now derived a parallel to (45). Using the same arguments in the proof of Lemma 8, we have

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant (k-1)\left(1 - \mathcal{E}_{\widetilde{\mathcal{S}}_M}(\widehat{f})\right) + (k-1)\sqrt{\frac{k}{\delta(k-1)}\left(\frac{1}{2m_1} + \frac{3\beta}{m+n}\right)}. \tag{90}$$

Part-2: We now combine the results in Lemma 9 and Lemma 10 to obtain the final inequality in terms of quantities that can be computed from just the randomly labeled and clean data. Similar to the binary case, we obtained a polynomial concentration instead of exponential concentration. Combining (90) with Lemma 9 and Lemma 10, we have with probability at least $1 - \delta$

$$\mathcal{E}_{\mathcal{D}}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{S}}(\widehat{f}) + (k-1)\left(1 - \frac{k}{k-1}\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f})\right)$$
$$+ \left(k + \sqrt{k} + \frac{m}{n\sqrt{k}}\right)\sqrt{\frac{\log(4/\delta)}{2m}} + \sqrt{k(k-1)}\sqrt{\frac{4}{\delta}\left(\frac{1}{m} + \frac{3\beta}{m+n}\right)}. \tag{91}$$

□

B.3. Discussion on Condition 1

The quantity in LHS of Condition 1 measures how much the function learned by the algorithm (in terms of error on unseen point) will change when one point in the training set is removed. We need hypothesis stability condition to control the variance of the empirical leave-one-out error to show concentration of average leave-one-error with the population error.

Additionally, we note that while the dominating term in the RHS of Theorem 4 matches with the dominating term in ERM bound in Theorem 1, there is a polynomial concentration term (dependence on $1/\delta$ instead of $\log(\sqrt{1/\delta})$) in Theorem 4.

Since with hypothesis stability, we just bound the variance, the polynomial concentration is due to the use of Chebyshev's inequality instead of an exponential tail inequality (as in Lemma 1). Recent works have highlighted that a slightly stronger condition than hypothesis stability can be used to obtain an exponential concentration for leave-one-out error (Abou-Moustafa & Szepesvári, 2019), but we leave this for future work for now.

B.4. Formal statement and proof of Proposition 2

Before formally presenting the result, we will introduce some notation. By $\mathcal{L}_S(w)$, we denote the objective in (59) with $\lambda = 0$. Assume Singular Value Decomposition (SVD) of $\boldsymbol{X}$ as $\sqrt{n}\boldsymbol{U}\boldsymbol{S}^{1/2}\boldsymbol{V}^T$. Hence $\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{V}\boldsymbol{S}\boldsymbol{V}^T$. Consider the GD iterates defined in (60). We now derive closed form expression for the t^{th} iterate of gradient descent:

$$w_t = w_{t-1} + \eta \cdot \boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}w_{t-1}) = (\boldsymbol{I} - \eta\boldsymbol{V}\boldsymbol{S}\boldsymbol{V}^T)w_{k-1} + \eta\boldsymbol{X}^T\boldsymbol{y}\,. \tag{92}$$

Rotating by $\boldsymbol{V}^T$, we get

$$\widetilde{w}_t = (\boldsymbol{I} - \eta\boldsymbol{S})\widetilde{w}_{k-1} + \eta\widetilde{\boldsymbol{y}}, \tag{93}$$

where $\widetilde{w}_t = \boldsymbol{V}^T w_t$ and $\widetilde{\boldsymbol{y}} = \boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}$. Assuming the initial point $w_0 = 0$ and applying the recursion in (93), we get

$$\widetilde{w}_t = \boldsymbol{S}^{-1}(\boldsymbol{I} - (\boldsymbol{I} - \eta\boldsymbol{S})^k)\widetilde{\boldsymbol{y}}\,, \tag{94}$$

Projecting solution back to the original space, we have

$$w_t = \boldsymbol{V}\boldsymbol{S}^{-1}(\boldsymbol{I} - (\boldsymbol{I} - \eta\boldsymbol{S})^k)\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}\,. \tag{95}$$

Define $f_t(x) := f(x; w_t)$ as the solution at the t^{th} iterate. Let $\widetilde{w}_\lambda = \arg\min_w \mathcal{L}_{\mathcal{S}}(w;\lambda) = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y} = \boldsymbol{V}(\boldsymbol{S} + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}$. and define $\widetilde{f}_\lambda(x) := f(x; \widetilde{w}_\lambda)$ as the regularized solution. Assume κ be the condition number of the population covariance matrix and let $s_{\min}$ be the minimum positive singular value of the empirical covariance matrix. Our proof idea is inspired from recent work on relating gradient flow solution and regularized solution for regression problems (Ali et al., 2018). We will use the following lemma in the proof:

Lemma 13. *For all $x \in [0,1]$ and for all $k \in \mathbb{N}$, we have (a) $\frac{kx}{1+kx} \leqslant 1 - (1-x)^k$ and (b) $1 - (1-x)^k \leqslant 2 \cdot \frac{kx}{kx+1}$.*

Proof. Using $(1-x)^k \leqslant \frac{1}{1+kx}$, we have part (a). For part (b), we numerically maximize $\frac{(1+kx)(1-(1-x)^k)}{kx}$ for all $k \geqslant 1$ and for all $x \in [0,1]$. □

Proposition 3 (Formal statement of Proposition 2)**.** *Let $\lambda = \frac{1}{t\eta}$. For a training point x, we have*

$$\mathbb{E}_{x\sim\mathcal{S}}\left[(f_t(x) - \widetilde{f}_\lambda(x))^2\right] \leqslant c(t,\eta) \cdot \mathbb{E}_{x\sim\mathcal{S}}\left[f_t(x)^2\right]\,,$$

where $c(t,\eta) := \min(0.25, \frac{1}{s_{min}^2 t^2\eta^2})$. Similarly for a test point, we have

$$\mathbb{E}_{x\sim\mathcal{D}_{\mathcal{X}}}\left[(f_t(x) - \widetilde{f}_\lambda(x))^2\right] \leqslant \kappa \cdot c(t,\eta) \cdot \mathbb{E}_{x\sim\mathcal{D}_{\mathcal{X}}}\left[f_t(x)^2\right]\,.$$

Proof. We want to analyze the expected squared difference output of regularized linear regression with regularization constant $\lambda = \frac{1}{\eta t}$ and the gradient descent solution at the t^{th} iterate. We separately expand the algebraic expression for squared difference at a training point and a test point. Then the main step is to show that $\left[\boldsymbol{S}^{-1}(\boldsymbol{I} - (\boldsymbol{I} - \eta\boldsymbol{S})^k) - (\boldsymbol{S} + \lambda\boldsymbol{I})^{-1}\right] \preceq c(\eta,t) \cdot \boldsymbol{S}^{-1}(\boldsymbol{I} - (\boldsymbol{I} - \eta\boldsymbol{S})^k)$.

Part 1 First, we will analyze the squared difference of the output at a training point (for simplicity, we refer to $S \cup \widetilde{S}$ as S),

i.e.,

$$\mathbb{E}_{x\sim\mathcal{S}}\left[\left(f_t(x)-\widetilde{f}_\lambda(x)\right)^2\right] = \|\boldsymbol{X}w_t-\boldsymbol{X}\widetilde{w}_\lambda\|_2^2 \tag{96}$$

$$= \left\|\boldsymbol{X}\boldsymbol{V}\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}-\boldsymbol{X}\boldsymbol{V}(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}\right\|_2^2 \tag{97}$$

$$= \left\|\boldsymbol{X}\boldsymbol{V}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)-(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1}\right)\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}\right\|_2 \tag{98}$$

$$= \boldsymbol{y}^T\boldsymbol{V}\boldsymbol{X}\left(\underbrace{\boldsymbol{S}^{-1}(\boldsymbol{I}-\underbrace{(\boldsymbol{I}-\eta\boldsymbol{S})^t}_{\text{I}})-(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1}}\right)^2\boldsymbol{S}\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}\,. \tag{99}$$

We now separately consider term 1. Substituting $\lambda=\frac{1}{t\eta}$, we get

$$\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)-(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1} = \boldsymbol{S}^{-1}\left((\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)-(\boldsymbol{I}+\boldsymbol{S}^{-1}\lambda)^{-1}\right) \tag{100}$$

$$= \underbrace{\boldsymbol{S}^{-1}\left((\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)-(\boldsymbol{I}+(\boldsymbol{S}t\eta)^{-1})^{-1}\right)}_{\boldsymbol{A}}\,. \tag{101}$$

We now separately bound the diagonal entries in matrix $\boldsymbol{A}$. With s_i, we denote i^{th} diagonal entry of $\boldsymbol{S}$. Note that since $\eta\leqslant 1/\|S\|_{\text{op}}$, for all i, $\eta s_i\leqslant 1$. Consider i^{th} diagonal term (which is non-zero) of the diagonal matrix $\boldsymbol{A}$, we have

$$\boldsymbol{A}_{ii} = \frac{1}{s_i}\left(1-(1-s_i\eta)^t-\frac{t\eta s_i}{1+t\eta s_i}\right) = \frac{1-(1-s_i\eta)^t}{s_i}\left(\underbrace{1-\frac{t\eta s_i}{(1+t\eta s_i)\underbrace{(1-(1-s_i\eta)^t)}_{\text{II}}}}\right) \tag{102}$$

$$\leqslant \frac{1}{2}\left[\frac{1-(1-s_i\eta)^t}{s_i}\right]. \qquad \text{(Using Lemma 13 (b))}$$

Additionally, we can also show the following upper bound on term 2:

$$1-\frac{t\eta s_i}{(1+t\eta s_i)(1-(1-s_i\eta)^t)} = \frac{(1+t\eta s_i)(1-(1-s_i\eta)^t)-t\eta s_i}{(1+t\eta s_i)(1-(1-s_i\eta)^t)} \tag{103}$$

$$\leqslant \frac{1-(1-s_i\eta)^t-t\eta s_i(1-s_i\eta)^t}{(1+t\eta s_i)(1-(1-s_i\eta)^t)} \tag{104}$$

$$\leqslant \frac{1}{t\eta s_i}\,. \qquad \text{(Using Lemma 13 (a))}$$

Combining both the upper bounds on each diagonal entry $\boldsymbol{A}_{ii}$, we have

$$\boldsymbol{A}\preceq c_1(\eta,t)\cdot\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\,, \tag{105}$$

where $c_1(\eta,t)=\min(0.5,\frac{1}{ts_i\eta})$. Plugging this into (99), we have

$$\mathbb{E}_{x\sim\mathcal{S}}\left[\left(f_t(x)-\widetilde{f}_\lambda(x)\right)^2\right] \leqslant c(\eta,t)\cdot\boldsymbol{y}^T\boldsymbol{V}\boldsymbol{X}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\right)^2\boldsymbol{S}\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y} \tag{106}$$

$$= c(\eta,t)\cdot\boldsymbol{y}^T\boldsymbol{V}\boldsymbol{X}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\right)\boldsymbol{S}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\right)\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y} \tag{107}$$

$$= c(\eta,t)\cdot\|\boldsymbol{X}w_t\|_2^2 \tag{108}$$

$$= c(\eta,t)\cdot\mathbb{E}_{x\sim\mathcal{S}}\left[(f_t(x))^2\right], \tag{109}$$

where $c(\eta,t)=\min(0.25,\frac{1}{t^2s_i^2\eta^2})$.

Part 2 With $\mathbf{\Sigma}$, we denote the underlying true covariance matrix. We now consider the squared difference of output at an unseen point:

$$\mathbb{E}_{x\sim\mathcal{D}_\mathcal{X}}\left[\left(f_t(x)-\widetilde{f}_\lambda(x)\right)^2\right] = \mathbb{E}_{x\sim\mathcal{D}_\mathcal{X}}\left[\left\|x^T w_t - x^T\widetilde{w}_\lambda\right\|_2\right] \tag{110}$$

$$= \left\|x^T\boldsymbol{V}\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y} - x^T\boldsymbol{V}(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1}\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}\right\|_2 \tag{111}$$

$$= \left\|x^T\boldsymbol{V}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)-(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1}\right)\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}\right\|_2 \tag{112}$$

$$= \boldsymbol{y}^T\boldsymbol{V}\boldsymbol{X}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)-(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1}\right)\boldsymbol{V}^T\mathbf{\Sigma}\boldsymbol{V} \tag{113}$$

$$\left((\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)-(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1}\right)\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y} \tag{114}$$

$$\leqslant \sigma_{\max}\cdot\boldsymbol{y}^T\boldsymbol{V}\boldsymbol{X}\left(\underbrace{\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)-(\boldsymbol{S}+\lambda\boldsymbol{I})^{-1}}_{\mathrm{I}}\right)^2\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}\,, \tag{115}$$

where $\sigma_{\max}$ is the maximum eigenvalue of the underlying covariance matrix $\mathbf{\Sigma}$. Using the upper bound on term 1 in (105), we have

$$\mathbb{E}_{x\sim\mathcal{D}_\mathcal{X}}\left[\left(f_t(x)-\widetilde{f}_\lambda(x)\right)^2\right] \leqslant \sigma_{\max}\cdot c(\eta,t)\cdot\boldsymbol{y}^T\boldsymbol{V}\boldsymbol{X}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\right)^2\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y} \tag{116}$$

$$= \kappa\cdot c(\eta,t)\cdot\sigma_{\min}\cdot\left\|\boldsymbol{V}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\right)\boldsymbol{V}^T\boldsymbol{X}^T\boldsymbol{y}\right\|_2^2 \tag{117}$$

$$\leqslant \kappa\cdot c(\eta,t)\cdot\left[\boldsymbol{V}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\right)\boldsymbol{V}^T\boldsymbol{X}^T\right]^T\mathbf{\Sigma} \tag{118}$$

$$\left[\boldsymbol{V}\left(\boldsymbol{S}^{-1}(\boldsymbol{I}-(\boldsymbol{I}-\eta\boldsymbol{S})^t)\right)\boldsymbol{V}^T\boldsymbol{X}^T\right]\boldsymbol{y} \tag{119}$$

$$= \kappa\cdot c(\eta,t)\cdot\mathbb{E}_{x\sim\mathcal{D}_\mathcal{X}}\left[\left\|x^T w_t\right\|_2\right]. \tag{120}$$

□

B.5. Extension to deep learning

Under Assumption B.6, we present the formal result parallel to Theorem 3.

Theorem 6. *Consider a multiclass classification problem with k classes. Under Assumption 1, for any $\delta > 0$, with probability at least $1-\delta$, we have*

$$\mathcal{E}_\mathcal{D}(\widehat{f}) \leqslant \mathcal{E}_\mathcal{S}(\widehat{f}) + (k-1)\left(1-\tfrac{k}{k-1}\mathcal{E}_{\widetilde{\mathcal{S}}}(\widehat{f})\right) + c\sqrt{\frac{\log(\frac{4}{\delta})}{2m}}\,, \tag{121}$$

for some constant $c \leqslant ((c+1)k+\sqrt{k}+\frac{m}{n\sqrt{k}})$.

The proof follows exactly as in step (i) to (iii) in Theorem 3.

B.6. Justifying Assumption 1

Motivated by the analysis on linear models, we now discuss alternate (and weaker) conditions that imply Assumption 1. We need hypothesis stability (Condition 1) and the following assumption relating training error and leave-one-error:

Assumption 2. *Let $\widehat{f}$ be a model obtained by training with algorithm $\mathcal{A}$ on a mixture of clean S and randomly labeled data $\widetilde{S}$. Then we assume we have*

$$\mathcal{E}_{\widetilde{S}_M}(\widehat{f}) \leqslant \mathcal{E}_{LOO(\widetilde{S}_M)}\,,$$

for all $(x_i,y_i)\in\widetilde{S}_M$ where $\widehat{f}_{(i)} := f(\mathcal{A}, S\cup\widetilde{S}_{M(i)})$ and $\mathcal{E}_{LOO(\widetilde{S}_M)} := \frac{\sum_{(x_i,y_i)\in\widetilde{S}_M}\mathcal{E}(\widehat{f}_{(i)}(x_i),y_i)}{|\widetilde{S}_M|}$.

Intuitively, this assumption states that the error on a (mislabeled) datum (x,y) included in the training set is less than the error on that datum (x,y) obtained by a model trained on the training set $S-\{(x,y)\}$. We proved this for linear models

trained with GD in the proof of Theorem 5. Condition 1 with $\beta = \mathcal{O}(1)$ and Assumption 2 together with Lemma 12 implies Assumption 1 with a polynomial residual term (instead of logarithmic in $1/\delta$):

$$\mathcal{E}_{\mathcal{S}_M}(\widehat{f}) \leqslant \mathcal{E}_{\mathcal{D}'}(\widehat{f}) + \sqrt{\frac{1}{\delta}\left(\frac{1}{m} + \frac{3\beta}{m+n}\right)}. \tag{122}$$

C. Additional experiments and details

C.1. Datasets

Toy Dataset Assume fixed constants μ and σ. For a given label y, we simulate features x in our toy classification setup as follows:

$$x := \texttt{concat}\,[x_1, x_2] \quad \text{where} \quad x_1 \sim \mathcal{N}(y \cdot \mu, \sigma^2 I_{d\times d}) \;\text{ and }\; x_1 \sim \mathcal{N}(0, \sigma^2 I_{d\times d})\,.$$

In experiements throughout the paper, we fix dimention $d = 100$, $\mu = 1.0$, and $\sigma = \sqrt{d}$. Intuitively, x_1 carries the information about the underlying label and x_2 is additional noise independent of the underlying label.

CV datasets We use MNIST (LeCun et al., 1998) and CIFAR10 (Krizhevsky & Hinton, 2009). We produce a binary variant from the multiclass classification problem by mapping classes $\{0, 1, 2, 3, 4\}$ to label 1 and $\{5, 6, 7, 8, 9\}$ to label -1. For CIFAR dataset, we also use the standard data augementation of random crop and horizontal flip. PyTorch code is as follows:

```
(transforms.RandomCrop(32, padding=4),
     transforms.RandomHorizontalFlip())
```

NLP dataset We use IMDb Sentiment analysis (Maas et al., 2011) corpus.

C.2. Architecture Details

All experiments were run on NVIDIA GeForce RTX 2080 Ti GPUs. We used PyTorch (Paszke et al., 2019) and Keras with Tensorflow (Abadi et al., 2016) backend for experiments.

Linear model For the toy dataset, we simulate a linear model with scalar output and the same number of parameters as the number of dimensions.

Wide nets To simulate the NTK regime, we experiment with $2-$layered wide nets. The PyTorch code for 2-layer wide MLP is as follows:

```
nn.Sequential(
    nn.Flatten(),
    nn.Linear(input_dims, 200000, bias=True),
    nn.ReLU(),
    nn.Linear(200000, 1, bias=True)
    )
```

We experiment both (i) with the second layer fixed at random initialization; (ii) and updating both layers' weights.

Deep nets for CV tasks We consider a 4-layered MLP. The PyTorch code for 4-layer MLP is as follows:

```
nn.Sequential(nn.Flatten(),
    nn.Linear(input_dim, 5000, bias=True),
    nn.ReLU(),
    nn.Linear(5000, 5000, bias=True),
    nn.ReLU(),
    nn.Linear(5000, 5000, bias=True),
    nn.ReLU(),
    nn.Linear(1024, num_label, bias=True)
    )
```

For MNIST, we use 1000 nodes instead of 5000 nodes in the hidden layer. We also experiment with convolutional nets. In particular, we use ResNet18 (He et al., 2016). Implementation adapted from: https://github.com/kuangliu/pytorch-cifar.git.

Deep nets for NLP We use a simple LSTM model with embeddings intialized with ELMo embeddings (Peters et al., 2018). Code adapted from: https://github.com/kamujun/elmo_experiments/blob/master/elmo_experiment/notebooks/elmo_text_classification_on_imdb.ipynb

We also evaluate our bounds with a BERT model. In particular, we fine-tune an off-the-shelf uncased BERT model (Devlin et al., 2018). Code adapted from Hugging Face Transformers (Wolf et al., 2020): `https://huggingface.co/transformers/v3.1.0/custom_datasets.html`.

C.3. Additonal experiments

Results with SGD on underparameterized linear models

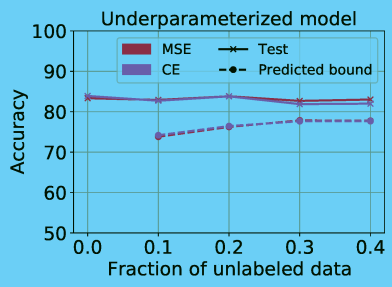
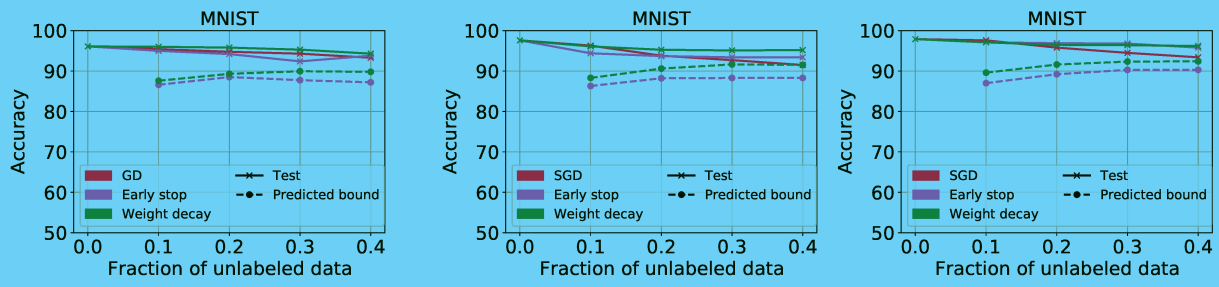


Figure 3. We plot the accuracy and corresponding bound (RHS in (1)) at $\delta = 0.1$ for toy binary classification task. Results aggregated over 3 seeds. Accuracy vs fraction of unlabeled data (w.r.t clean data) in the toy setup with a linear model trained with SGD. Results parallel to Fig. 2(a) with SGD.

Results with wide nets on binary MNIST



(a) GD with MSE loss (b) SGD with CE loss (c) SGD with MSE loss

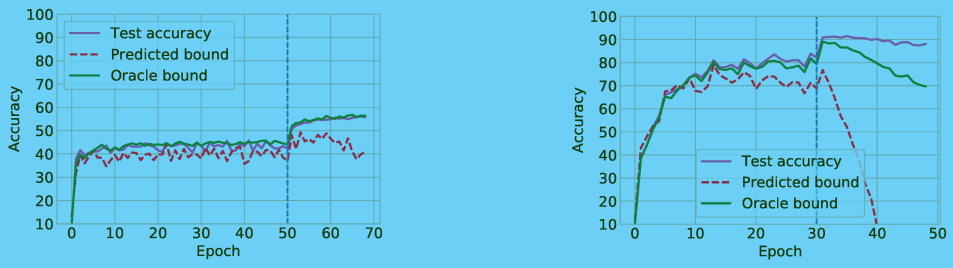
Figure 4. We plot the accuracy and corresponding bound (RHS in (1)) at $\delta = 0.1$ for binary MNIST classification. Results aggregated over 3 seeds. Accuracy vs fraction of unlabeled data for a 2-layer wide network on binary MNIST with both the layers training in (a,b) and only first layer training in (c). Results parallel to Fig. 2(b) .

Results on CIFAR 10 and MNIST We plot epoch wise error curve for results in Table 1(Fig. 5 and Fig. 6). We observe the same trend as in Fig. 1. Additionally, we plot an *oracle bound* obtained by tracking the error on mislabeled data which nevertheless were predicted as true label. To obtain an exact emprical value of the oracle bound, we need underlying true labels for the randomly labeled data. While with just access to extra unlabeled data we cannot calculate oracle bound, we note that the oracle bound is very tight and never violated in practice underscoring an importamt aspect of generalization in multiclass problems. This highlight that even a stronger conjecture may hold in multiclass classification, i.e., error on mislabeled data (where nevertheless true label was predicted) lower bounds the population error on the distribution of mislabeled data and hence, the error on (a specific) mislabeled portion predicts the population accuracy on clean data. On the other hand, the dominating term of in Theorem 3 is loose when compared with the oracle bound. The main reason, we believe is the pessimistic upper bound in (45) in the proof of Lemma 8. We leave an investigation on this gap for future.

Results on CIFAR 100 On CIFAR100, our bound in (5) yields vacous bounds. However, the oracle bound as explained above yields tight guarantees in the initial phase of the learning (i.e., when learning rate is less than 0.1) (Fig. 7).

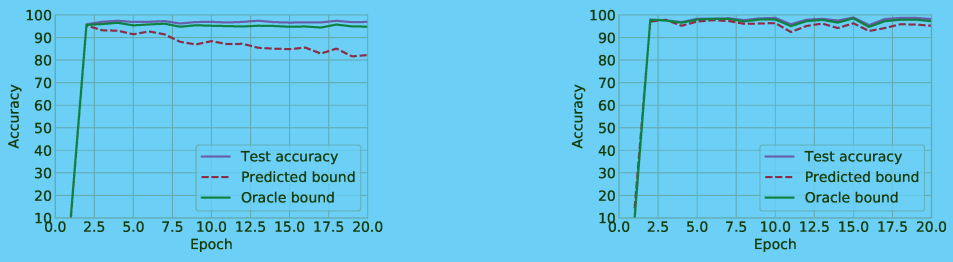
C.4. Hyperparameter Details

Fig. 1 We use clean training dataset of size $40,000$. We fix the amount of unlabeled data at 20% of the clean size, i.e. we include additional $8,000$ points with randomly assigned labels. We use test set of $10,000$ points. For both MLP and ResNet,



(a) MLP (b) ResNet

Figure 5. Per epoch curves for CIFAR10 corresponding results in Table 1. As before, we just plot the dominating term in the RHS of (5) as predicted bound. Additionally, we also plot the predicted lower bound by the error on mislabeled data which nevertheless were predicted as true label. We refer to this as "Oracle bound". See text for more details.



(a) MLP (b) ResNet

Figure 6. Per epoch curves for MNIST corresponding results in Table 1. As before, we just plot the dominating term in the RHS of (5) as predicted bound. Additionally, we also plot the predicted lower bound by the error on mislabeled data which nevertheless were predicted as true label. We refer to this as "Oracle bound". See text for more details.

we use SGD with an initial learning rate of 0.1 and momentum 0.9. We fix the weight decay parameter at 5×10^{-4}. After 100 epochs, we decay the learning rate to 0.01. We use SGD batch size of 100.

Fig. 2 (a) We obtain a toy dataset according to the process described in Sec. C.1. We fix $d = 100$ and create a dataset of $50,000$ points with balanced classes. Moreover, we sample additional covariates with the same procedure to create randomly labeled dataset. For both SGD and GD training, we use a fixed learning rate 0.1.

Fig. 2 (b) Similar to binary CIFAR, we use clean training dataset of size $40,000$ and fix the amount of unlabeled data at 20% of the clean dataset size. To train wide nets, we use a fixed learning of 0.001 with GD and SGD. We decide the weight decay parameter and the early stopping point that maximizes our generalization bound (i.e. without peeking at unseen data). We use SGD batch size of 100.

Fig. 2 (c) With IMDb dataset, we use a clean dataset of size $20,000$ and as before, fix the amount of unlabeled data at 20% of the clean data. To train ELMo model, we use Adam optimizer with a fixed learning rate 0.01 and weight decay 10^{-6} to minimize cross entropy loss. We train with batch size 32 for 3 epochs. To fine-tune BERT model, we use Adam optimizer with learning rate 5×10^{-5} to minimize cross entropy loss. We train with a batch size of 16 for 1 epoch.

Table 1 For multiclass datasets, we train both MLP and ResNet with the same hyperparameters as described before. We sample a clean training dataset of size $40,000$ and fix the amount of unlabeled data at 20% of the clean size. We use SGD with an initial learning rate of 0.1 and momentum 0.9. We fix the weight decay parameter at 5×10^{-4}. After 30 epochs for ResNet and after 50 epochs for MLP, we decay the learning rate to 0.01. We use SGD with batch size 100. For Fig. 7, we use the same hyperparameters as CIFAR10 training, except we now decay learning rate after 100 epochs.

In all experiments, to identify the best possible accuracy on just the clean data, we use the exact same set of hyperparamters except the stopping point. We choose a stopping point that maximizes test performance.

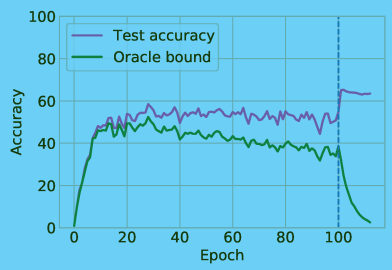


Figure 7. Predicted lower bound by the error on mislabeled data which nevertheless were predicted as true label with ResNet18 on CIFAR100. We refer to this as "Oracle bound". See text for more details. The bound predicted by RATT (RHS in (5)) is vacuous.

C.5. Summary of experiments

Classification type	Model category	Model	Dataset
Binary	Low dimensional	Linear model	Toy Gaussain dataset
	Overparameterized linear nets	2-layer wide net	Binary MNIST
	Deep nets	MLP	Binary MNIST
			Binary CIFAR
		ResNet	Binary MNIST
			Binary CIFAR
		ELMo-LSTM model	IMDb Sentiment Analysis
		BERT pre-trained model	IMDb Sentiment Analysis
Multiclass	Deep nets	MLP	MNIST
			CIFAR10
		ResNet	MNIST
			CIFAR10
			CIFAR100

D. Proof of Lemma 12

Proof of Lemma 12. Recall, we have a training set $S \cup \widetilde{S}_C$. We defined leave-one-out error on mislabeled points as

$$\mathcal{E}_{\mathrm{LOO}(\widetilde{S}_M)} = \frac{\sum_{(x_i,y_i)\in\widetilde{S}_M} \mathcal{E}(f_{(i)}(x_i), y_i)}{\left|\widetilde{S}_M\right|},$$

where $f_{(i)} := f(\mathcal{A}, (S \cup \widetilde{S})_{(i)})$. Define $S' := S \cup \widetilde{S}$. Assume (x, y) and (x', y') as i.i.d. samples from $\mathcal{D}'$. Using Lemma 25 in Bousquet & Elisseeff (2002), we have

$$\begin{aligned}\mathbb{E}\left[\left(\mathcal{E}_{\mathcal{D}'}(\widehat{f}) - \mathcal{E}_{\mathrm{LOO}(\widetilde{S}_M)}\right)^2\right] \leqslant & \mathbb{E}_{S',(x,y),(x',y')}\left[\mathcal{E}(\widehat{f}(x), y)\mathcal{E}(\widehat{f}(x'), y')\right] - 2\mathbb{E}_{S',(x,y)}\left[\mathcal{E}(\widehat{f}(x), y)\mathcal{E}(f_{(i)}(x_i), y_i)\right] \\ & + \frac{m_1 - 1}{m_1}\mathbb{E}_{S'}\left[\mathcal{E}(f_{(i)}(x_i), y_i)\mathcal{E}(f_{(j)}(x_j), y_j)\right] + \frac{1}{m_1}\mathbb{E}_{S'}\left[\mathcal{E}(f_{(i)}(x_i), y_i)\right]. \end{aligned} \tag{123}$$

We can rewrite the equation above as :

$$\begin{aligned}\mathbb{E}\left[\left(\mathcal{E}_{\mathcal{D}'}(\widehat{f}) - \mathcal{E}_{\mathrm{LOO}(\widetilde{S}_M)}\right)^2\right] \leqslant & \underbrace{\mathbb{E}_{S',(x,y),(x',y')}\left[\mathcal{E}(\widehat{f}(x), y)\mathcal{E}(\widehat{f}(x'), y') - \mathcal{E}(\widehat{f}(x), y)\mathcal{E}(f_{(i)}(x_i), y_i)\right]}_{\text{I}} \\ & + \underbrace{\mathbb{E}_{S'}\left[\mathcal{E}(f_{(i)}(x_i), y_i)\mathcal{E}(f_{(j)}(x_j), y_j) - \mathcal{E}(\widehat{f}(x), y)\mathcal{E}(f_{(i)}(x_i), y_i)\right]}_{\text{II}} \\ & + \underbrace{\frac{1}{m_1}\mathbb{E}_{S'}\left[\mathcal{E}(f_{(i)}(x_i), y_i) - \mathcal{E}(f_{(i)}(x_i), y_i)\mathcal{E}(f_{(j)}(x_j), y_j)\right]}_{\text{III}}. \end{aligned} \tag{124}$$

We will now bound term III. Using Cauchy-Schwarz's inequality, we have

$$\begin{aligned}\mathbb{E}_{S'}\left[\mathcal{E}(f_{(i)}(x_i), y_i) - \mathcal{E}(f_{(i)}(x_i), y_i)\mathcal{E}(f_{(j)}(x_j), y_j)\right]^2 &\leqslant \mathbb{E}_{S'}\left[\mathcal{E}(f_{(i)}(x_i), y_i)\right]^2 \mathbb{E}_{S'}\left[1 - \mathcal{E}(f_{(j)}(x_j), y_j)\right]^2 &(125)\\ &\leqslant \frac{1}{4}. &(126)\end{aligned}$$

Note that since (x_i, y_i), (x_j, y_j), (x, y), and (x', y') are all from same distribution $\mathcal{D}'$, we directly incorporate the bounds on term I and II from the proof of Lemma 9 in Bousquet & Elisseeff (2002). Combining that with (126) and our definition of hypothesis stability in Condition 1, we have the required claim.

□