Learning from Label Proportions by Learning with Label Noise

Jianxin Zhang, Yutong Wang, and Clayton Scott
Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI 48109
{jianxinz, yutongw, clayscot}@umich.edu

Abstract

Learning from label proportions (LLP) is a weakly supervised classification problem where data points are grouped into bags, and the label proportions within each bag are observed instead of the instance-level labels. The task is to learn a classifier to predict the labels of future individual instances. Prior work on LLP for multi-class data has yet to develop a theoretically grounded algorithm. In this work, we propose an approach to LLP based on a reduction to learning with label noise, using the forward correction (FC) loss of Patrini et al. [30]. We establish an excess risk bound and generalization error analysis for our approach, while also extending the theory of the FC loss which may be of independent interest. Our approach demonstrates improved empirical performance in deep learning scenarios across multiple datasets and architectures, compared to the leading methods.

1 Introduction

In the weakly supervised problem of *learning from label proportions* (LLP), the learner is presented with bags of instances, where each bag is annotated with the proportions of the different classes in the bag. The learner's objective is to produce a classifier that accurately assigns labels to individual instances in the future. LLP arises in various applications including high energy physics [7], election prediction [45], computer vision [4, 20], medical image analysis [2], remote sensing [8], activity recognition [32], and reproductive medicine [12].

To date, most methods for LLP have addressed the setting of binary classification [50, 36, 39, 34, 41, 44, 24, 37, 38], although multiclass methods have also recently been investigated [9, 22, 46]. The dominant approach to LLP in the literature is "label proportion matching": train a classifier to accurately reproduce the observed label proportions on the training data, perhaps with additional regularization. In the multiclass setting, the Kullback-Leibler (KL) divergence between the observed and predicted label proportions is adopted by the leading approaches to assess proportion matching. Unfortunately, while matching the observed label proportions is intuitive and can work well in some settings, it has little theoretical basis [50, 38], especially in the multiclass setting, and there are natural settings where it fails [50, 39].

Recently, Scott and Zhang [39] demonstrated a principled approach to LLP with performance guarantees based on a reduction to learning with label noise (LLN) in the binary setting. Their basic strategy was to pair bags, and view each pair of bags as an LLN problem, where the observed label proportions are related to the "label flipping" or "noise transition" probabilities. Using an existing technique for LLN based on loss correction, which allows the learner to train directly on the noisy data, they formulated an overall objective based on a (weighted) sum of objectives for each pair of bags. They established generalization error analysis and consistency for the method, and also showed that in the context of kernel methods, their approach outperformed the leading kernel methods.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

The objective of the present paper is to develop a theoretically grounded and practical approach to multiclass LLP, drawing inspiration from Scott and Zhang [39]. The primary challenge stems from the fact that Scott and Zhang [39] employed the so-called "backward correction" loss, which solves LLN by scaling the *output* of a loss function of interest according to the noise transition probabilities [28, 30, 35]. While this loss correction was demonstrated to work well for kernel methods in a binary setting, Patrini et al. [30] introduced an alternative loss correction that performs better empirically in deep learning settings (see also [53]). They proposed the "forward correction" loss, which scales the *inputs* to a loss function of interest according to the noise transition probabilities. Patrini et al. [30] find that backward correction "does not seem to work well in the low noise regime," and is "a linear combination of losses" with "coefficients that can be far [apart] by orders of magnitude " which "makes the learning harder".

The present work is thus inspired by Scott and Zhang [39] but uses the forward correction (FC) loss in a multiclass setting. This requires a number of technical modifications to the arguments of Scott and Zhang [39]. Most notably, it now becomes necessary to demonstrate that the FC loss is *calibrated* with respect to the 0-1 loss, a critical property needed for showing consistency. Such analysis is inherently not needed when using the backward correction, where the target excess risk is *proportional to* the surrogate excess risk (from which calibration follows trivially). Furthermore, Scott and Zhang [39] does not require analysis of proper composite losses, which are needed in the FC framework. Finally, the multiclass setting involves new estimation challenges not present in the binary case. These factors mean that our work is not a straightforward extension of Scott and Zhang [39]. Indeed, the authors of a recent report acknowledge that it is "difficult to extend [the method of Scott and Zhang [39]] to multiclass classification" [16].

Additional related work: Much work on LLP has focused on learning specific types of models, including support vector machines [36, 50, 47, 33, 5, 19, 40], probabilistic models [18, 13, 45, 32, 12], random forests [41], neural networks [21, 1, 9, 22, 46], and clustering-based models, [3, 44]. Many of these works develop learning criteria that are specific to the model being learned.

On the theoretical front, Quadrianto et al. [34] and Patrini et al. [30] initiated the learning theoretic study of LLP, introducing Rademacher style bounds for linear methods, but they do not address consistency w.r.t. a classification performance measure. Yu et al. [51] provides support for label proportion matching but only under the assumption that the bags are very pure. Saket [37] studies learnability of linear threshold functions. Recently Saket et al. [38] introduced a condition under which label proportion matching does provably well w.r.t. a squared error loss in the binary setting, and developed an associated algorithm. This method does not scale easily to large datasets, and further requires knowledge of how bags are grouped according to different bag-generating distributions.

A handful of recent papers have studied multiclass LLP in deep learning scenarios. Dulac-Arnold et al. [9] study the KL loss for label proportion matching, and a variant based on optimal transport. Liu et al. [22, 23] examine an approach based on generative adversarial models. Tsai and Lin [46] study the use of a regularizer derived from semi-supervised learning. One challenge common to these approaches is that their implementations employ mini-batches of bags, which becomes computationally prohibitive for large bag sizes when the batch size is still very small, e.g., 2 or 3 bags. In contrast, our approach avoids this issue. Finally, a recent technical report presents a risk analysis for multiclass LLP under the assumption of fixed bag size, which we do not require [16]. Their method is not tractable for large bag sizes in which case they approximate their objective "using the bag-level loss proposed in the existing research."

Contributions and Outline: Our contributions and the paper structure are summarized as follows. In Section 2, we review the FC loss as a solution to LLN. In Section 3, we extend the theory of the FC loss for LLN. In particular, we show that the FC loss is "uniformly calibrated" with respect to the 0-1 loss using the framework of Steinwart [43], establish an excess risk bound, and determine an explicit lower bound on the calibration function in terms of the noise transition matrix. In Section 4, we extend the results of Section 3 to the setting with multiple noise transition matrices, which form the basis of our approach to LLP. In particular, we establish an excess risk bound and generalization error analysis for learning with multiple noise transition matrices, which in turn enables proofs of consistency. In Section 5, we state the probabilistic model for reducing LLP to LLN with multiple different noise transition matrices and present the LLPFC algorithms. Experiments with deep neural networks are presented in Section 6, where we observe that our approach outperforms competing methods by a substantial margin. Proofs appear in the supplemental material.

2 Learning with Label Noise and the Forward Correction Loss

This section sets notation and introduces the FC loss as a solution to learning with label noise. Let $\mathcal{X}$ be the feature space and $\mathcal{Y} = \{1, 2, \ldots, C\}$ be the label space, $C \in \mathbb{N}$. We define the C-simplex as $\Delta^C = \{p \in \mathbb{R}^C : p_i \geq 0, \forall i = 1, 2, \ldots, C, \sum_{i=1}^C p_i = 1\}$ and denote its interior by $\mathring{\Delta}^C$. Let P be a probability measure on the space $\mathcal{X} \times \mathcal{Y}$.

Viewing P as the "clean" probability measure, a noisy probability measure with label-dependent label noise can be constructed from P in terms of a $C \times C$ column-stochastic matrix T, referred to as the *noise transition matrix*. Formally, we define a measure $\bar{P}_T$ on $\mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$ by requiring $\forall$ events $\mathcal{A} \subset \mathcal{X}, \bar{P}_T(\mathcal{A} \times \{i\} \times \{j\}) = P(\mathcal{A} \times \{i\})t_{j,i}$ where $t_{j,i}$ is the element at the j-th row and i-th column of T. Let $(X, Y, \tilde{Y})$ have joint distribution $\bar{P}_T$ where X is the feature vector, Y is the "clean" label, and $\tilde{Y}$ is the "noisy" label. Thus the element of T at row i and column j is $t_{i,j} = \bar{P}_T(\tilde{Y} = i | Y = j)$. In addition, P is the marginal distribution of (X, Y). Define P_T to be the marginal distribution of $(X, \tilde{Y})$. Let $\mathcal{F}$ be the collection of all measurable functions from $\mathcal{X}$ to Δ^C.

The existence of a regular conditional distribution is guaranteed by the Disintegration Theorem ($e.g.$ Theorem 6.4 in Kallenberg [14]) under suitable properties ($e.g.$ when $\mathcal{X}$ is a Radon space). While the existence of regular conditional probability is beyond the scope of this paper, we assume fixed regular conditional distributions for Y and $\tilde{Y}$ given X exist, denoted by $P(\cdot \mid \cdot) : \mathcal{Y} \times \mathcal{X} \to [0, 1]$ and $P_T(\cdot \mid \cdot) : \mathcal{Y} \times \mathcal{X} \to [0, 1]$, respectively. Given $x \in \mathcal{X}$, we define the probability vectors $\eta(x) = [P(1 \mid x), \ldots, P(C \mid x)]^{tr}$ and $\eta_T(x) = [P_T(1 \mid x), \ldots, P_T(C \mid x)]^{tr}$ where we use tr to denote transposition. It directly follows that $\eta_T(x) = T\eta(x)$.

We use $\mathbb{R}_+$ to denote the positive real numbers. The goal of LLN is to learn a classifier that optimizes a performance measure defined $w.r.t.$ P, given access to corrupted training data $(X_i, \tilde{Y}_i) \overset{i.i.d.}{\sim} P_T$. In this work we assume T is known or can be estimated, as is the case when we apply LLN techniques to LLP (see Section 5). A more formal formulation of LLP is given in Section 5.

When attempting to minimize the risk associated to the 0-1 loss and the clean distribution P, it is common to employ a smooth or convex surrogate loss. For LLN problems, the idea of a *loss correction* is to modify the surrogate loss so that when optimized using the *noisy* data, it still achieves the desired goal. Below, we introduce the forward correction loss, before which we need to define inner risk and proper loss. For this purpose we focus on loss functions of the form $L : \Delta^C \times \mathcal{Y} \to \mathbb{R}$.

Definition 1. *Let $L : \Delta^C \times \mathcal{Y} \to \mathbb{R}$ be a loss function. The **inner L-risk** at x with probability measure P is $\mathcal{C}_{L,P,x} : \Delta^C \to \mathbb{R}, \;\; \mathcal{C}_{L,P,x}(q) := \mathbb{E}_{Y \sim P(\cdot|x)} L(q, Y)$. The **minimal inner L-risk** at x with a probability measure P is $\mathcal{C}^*_{L,P,x} := \inf_{q \in \Delta^C} \mathcal{C}_{L,P,x}(q)$.*

Definition 2. *$\ell : \Delta^C \times \mathcal{Y} \to \mathbb{R}$ is a **proper loss** if $\forall$ probability measures P on $\mathcal{X} \times \mathcal{Y}$, $\forall x \in \mathcal{X}, \mathcal{C}^*_{\ell,P,x} = \mathcal{C}_{\ell,P,x}(\eta(x))$, and a proper loss is called **strictly proper** if the minimizer of $\mathcal{C}_{\ell,P,x}$ is unique for all $x \in \mathcal{X}$.*

Commonly used proper losses include the *log loss* $\ell^{log}(q, c) = -\log q_c$, the *square loss* $\ell^{sq}(q, c) = \sum_{c'=1}^C (\mathbb{1}_{c=c'} - q_{c'})^2$, and the 0-1 *loss* $\ell^{01}(q, c) = \mathbb{1}_{c \neq \min\{\arg\max_j q_j\}}$, among which only the log loss and the square loss are strictly proper [49]. Here $\mathbb{1}$ denotes the indicator function. Note that it is common to compose proper losses with inverted link functions, leading to familiar losses like the cross-entropy loss. Such losses are discussed further in Section 4.

We are now ready to introduce the forward correction loss.

Definition 3. *Let ℓ be a strictly proper loss and let T be a noise transition matrix. Define the **forward correction loss** of ℓ as $\ell_T : \Delta^C \times \mathcal{Y} \to \mathbb{R}, \;\; \ell_T(q, c) := \ell(Tq, c)$.*

It follows from the definition that, if T is invertible, then the inner ℓ_T-risk under the distribution P_T has a unique minimizer $\eta(x)$. Next we introduce L-risk and L-Bayes risk associated with a loss L.

Definition 4. *Let $L : \Delta^C \times \mathcal{Y} \to \mathbb{R}$ and P be a probability measure. Define the **L-risk** of f with distribution P to be $\mathcal{R}_{L,P} : \mathcal{F} \to \mathbb{R}, \;\; \mathcal{R}_{L,P}(f) := \mathbb{E}_P [L(f(X), Y)]$ and the **L-Bayes risk** to be $\mathcal{R}^*_{L,P} := \inf_{f \in \mathcal{F}} \mathcal{R}_{L,P}(f)$.*

We call $\mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,P}$ the *excess L-risk* of f under distribution P. Given a proper loss ℓ, Theorem 2 of Patrini et al. [30] establishes Fisher consistency of the FC loss, meaning the minimizer of ℓ-risk

under the clean distribution P is the same as the minimizer of ℓ_T-risk under noisy distribution P_T: $\arg\min_{f\in\mathcal{F}} \mathcal{R}_{L,P}(f) = \arg\min_{f\in\mathcal{F}} \mathcal{R}_{\ell_T,P_T}(f)$. Next, we present a stronger result relating the excess ℓ_T-risk under the noisy distribution P_T to the excess 0-1 risk under the clean distribution P.

3 Calibration Analysis for the Forward Correction Loss

Our objective in this section is to show that when L is the 0-1 loss and ℓ is a continuous strictly proper surrogate loss, there exists a strictly increasing, invertible function θ with $\theta(0) = 0$ such that $\forall f \in \mathcal{F}$ and $\forall$ distributions P, $\theta\big(\mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,P}\big) \leq \mathcal{R}_{\ell_T,P_T}(f) - \mathcal{R}^*_{\ell_T,P_T}$. Given such a bound, it follows that consistency $w.r.t$ the surrogate risk implies consistency $w.r.t.$ the target risk. The results in this section are standalone results for the FC loss that may be of independent interest, and will be extended in the next section in relation to LLP. The following theorem guarantees the existence of such function θ, given that T is invertible.

Theorem 5. *Let ℓ be a continuous strictly proper loss and T be an invertible column-stochastic matrix. Let L be the 0-1 loss. Assume $\mathcal{R}^*_{\ell_T,P_T} < \infty$. Then $\exists\theta : [0,1] \to [0,\infty]$ that is strictly increasing and continuous, satisfying $\theta(0) = 0$, such that $\forall f \in \mathcal{F}, \mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,P} \leq \theta^{-1}\Big(\mathcal{R}_{\ell_T,P_T}(f) - \mathcal{R}^*_{\ell_T,P_T}\Big)$.*

The function θ in Theorem 5 depends on ℓ and T. The following proposition provides a convex lower bound on θ for the commonly used log loss $\ell^{log}(q,c) = -\log q_c$. Let $M \in \mathbb{R}^{C\times C}$ be a matrix and let $\|\cdot\|$ be a norm on $\mathbb{R}^C$. The *subordinate matrix norm* induced by $\|\cdot\|$ is $\|M\| := \sup_{x\in\mathbb{R}^C: x\neq 0} \frac{\|Mx\|}{\|x\|}$. When $\|\cdot\|$ is the 1-norm on $\mathbb{R}^C$, the induced norm is denoted $\|M\|_1$, referred to as the matrix 1-norm, and can be computed as $\|M\|_1 = \max_{1\leq j\leq C} \sum_{i=1}^{C} |M(i,j)|$ [10].

Proposition 6. *Let $T \in \mathbb{R}^{C\times C}$ be an invertible, column-stochastic matrix. Define $\underline{\theta}_T : [0,\infty] \to [0,\infty]$ by $\underline{\theta}_T(\epsilon) = \frac{1}{2}\frac{\epsilon^2}{\|T^{-1}\|_1^2}$. If L is the 0/1 loss, ℓ is the log loss, then for all $f \in \mathcal{F}$ and distributions P, $\mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,P} \leq \underline{\theta}_T^{-1}\Big(\mathcal{R}_{\ell_T,P_T}(f) - \mathcal{R}^*_{\ell_T,P_T}\Big) = \sqrt{2}\|T^{-1}\|_1\sqrt{\mathcal{R}_{\ell_T,P_T}(f) - \mathcal{R}^*_{\ell_T,P_T}}$*

The factor $\|T^{-1}\|_1$ may be viewed as a constant that captures the overall amount of label noise. The more noise, the larger the constant. For example, let I and N be the identity and the all $1/C$'s matrices, respectively. Let $\alpha \in [0,1]$ and $T = (1-\alpha)I + \alpha N$. Thus, $\alpha = 0$ represents the noise-free case and $\alpha = 1$ the noise-only case. It is easy to verify that $T^{-1} = (1-\alpha)^{-1}(I - \alpha N)$ and $\|T^{-1}\|_1 = (1-\alpha)^{-1}(1 + (1-2/C)\alpha)$.

4 Learning with Multiple Noise Transition Matrices

Our algorithms for LLP, formally stated in subsection 5.4, reduce the problem of LLP to LLN by partitioning bags into groups and modeling each group as an LLN problem. Since each group has its own noise transition matrix, this leads to a new problem that we refer to as learning with multiple noise transition matrices (LMNTM). In this section, we show how to extend the calibration analysis of section 3 to this setting. In addition, we offer a generalization error bound that justifies an empirical risk minimization learning procedure based on a weighted sum of FC losses.

4.1 Learning with Multiple Noise Transition Matrices

We first define the LMNTM problem formally. For all $n \in \mathbb{N}$, denote $\mathbb{N}_n = \{1, 2, \ldots, n\}$. Consider a clean distribution P on $\mathcal{X} \times \mathcal{Y}$ and noise transition matrices $T_1, T_2, \ldots, T_N$. For each i we denote the noisy prior as the $\alpha_i \in \mathring{\Delta}^C$ where, $\forall c \in \mathcal{Y}$, $\alpha_i(c) = P_{T_i}(\tilde{Y} = c)$. We assume the α_i's are known for theoretical analysis. In practice, α_i is estimable as discussed below. In LMNTM, we observe data points $S = \big\{X_{i,c,j} : \ i \in \mathbb{N}_N, \ c \in \mathcal{Y}, \ j \in \mathbb{N}_{n_{i,c}}\big\}$ where $X_{i,c,j} \overset{iid}{\sim} P_{T_i}(\cdot \mid c)$, and $n_{i,c} \in \mathbb{N}$ is the number of data points drawn from the class conditional distribution $P_{T_i}(\cdot \mid c)$. Assume all $X_{i,c,j}$'s are mutually independent. We make additional remarks on this setting in Section C.1 in the appendix.

4.2 A Risk for LMNTM

The following result extends Theorem 5 to LMNTM. It establishes that the risk $\widetilde{R}_{\ell,P,\mathcal{T}}$, which can be estimated from LMNTM training data, is a valid surrogate risk. This type of result is not needed for the backward correction approach of Scott and Zhang [39].

Theorem 7. *Let L be the 0-1 loss and $N \in \mathbb{N}$. Consider a sequence of invertible column-stochastic matrices $\mathcal{T} = \{T_i\}_{i=1}^N$ and a continuous strictly proper loss function ℓ. Let $w = (w_i)_{i=1}^N \in \Delta^N$. Define $\widetilde{R}_{\ell,P,\mathcal{T}} : \mathcal{F} \to \mathbb{R}$ by $\widetilde{R}_{\ell,P,\mathcal{T}}(f) := \sum_{i=1}^N w_i \mathcal{R}_{\ell_{T_i},P_{T_i}}(f)$ and $\widetilde{R}^*_{\ell,P,\mathcal{T}} = \inf_{f\in\mathcal{F}} \widetilde{R}_{\ell,P,\mathcal{T}}(f)$. Assume $\forall i \in \{1,2,\ldots,N\}, \mathcal{R}^*_{\ell_{T_i},P_{T_i}} < \infty$. Then $\exists$ a strictly increasing continuous function θ : $[0,1] \to [0,\infty]$ with $\theta(0) = 0$ s.t. for all P, $\forall f \in \mathcal{F}$, $\theta\big(\mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,P}\big) \leq \widetilde{\mathcal{R}}_{\ell,P,\mathcal{T}}(f) - \widetilde{\mathcal{R}}^*_{\ell,P,\mathcal{T}}$.*

The weights w_i allow the user flexibility, for example, to place different weights on noisier or larger subsets of data. Unlike Scott and Zhang [39], however, because the weights appear in both our excess risk bound and generalization error bound, it is not straightforward to optimize them a priori.

4.3 Generalization Error Bound

The aggregate risk $\widetilde{R}_{\ell,P,\mathcal{T}}$ is desirable because it can naturally be estimated from the given data. We propose the empirical risk

$$\hat{\mathcal{R}}_{w,S}(f) = \sum_{i=1}^N w_i \sum_{c=1}^C \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_{i,c}} \ell_{T_i}(f(X_{i,c,j}), c). \tag{1}$$

It should be noted that $\hat{\mathcal{R}}_{w,S}(f)$ is an unbiased estimate of $\tilde{\mathcal{R}}_{\ell,P,\mathcal{T}}(f)$. Here we establish a generalization error bound for this estimate which builds on Rademacher complexity analysis .

To state the bound, we must first introduce the notion of a *proper composite loss* [49]. This stems from the fact that in practice, a function f outputting values in Δ^C is typically obtained by composing a $\mathbb{R}^C$-valued function (such as a neural network with C output layer nodes), with another function $\mathbb{R}^C \to \Delta^C$ such as the softmax function. Thus, let $\psi : \mathcal{U} \subset \Delta^C \to \mathcal{V}$ be an invertible function where $\mathcal{V}$ is a subset of a normed space, referred to as an *invertible link function*. Consider $\mathcal{G} \subset \psi \circ \mathcal{F} := \{\psi \circ f : f \in \mathcal{F}\}$, and observe that $\forall g \in \mathcal{G}, \psi^{-1} \circ g \in \mathcal{F}$. In practice, ψ is fixed and we seek to learn $g \in \mathcal{G}$ that leads to an $f \in \mathcal{F}$ with a risk close to the Bayes risk. An example of ψ^{-1} is the softmax function so that $\psi : \mathcal{U} \to \mathcal{V}, \psi_i(p) = \log p_i - \frac{1}{C}\sum_{k=1}^C \log p_k, (\psi^{-1})_i(s) = \frac{e^{s_i}}{\sum_{k=1}^C e^{s_k}}$ where $\mathcal{U}$ is the interior of Δ^C and $\mathcal{V} = \{s \in \mathbb{R}^C : \sum_{i=1}^C s_i = 0\}$. This motivates the following definition.

Definition 8. *Given an invertible link function $\psi : \mathcal{U} \subset \Delta^C \to \mathcal{V}$, we define the **proper composite loss** λ_ℓ of a proper loss $\ell : \Delta^C \times \mathcal{Y} \to \mathbb{R}$ to be $\lambda_\ell : \mathcal{V} \times \mathcal{Y} \to \mathbb{R}$, $\lambda_\ell(v,c) = \ell\big(\psi^{-1}(v), c\big)$.*

For example, when ℓ is the log loss and ψ^{-1} is the softmax function, λ_ℓ is the cross-entropy (or multinomial logistic) loss. With this notation, we are now able to state our generalization error bound for LMNTM. We study two popular choices of function classes, the reproducing kernel Hilbert space (RKHS) and the multilayer perceptron (MLP). We use $\mathcal{G}_1$ to denote the Cartesian product of C balls of radius R in the RKHS and $\mathcal{G}_2$ to denote a multilayer perceptron with C outputs.

Definition 9. *Let k be a symmetric positive definite (SPD) kernel, and let $\mathcal{H}$ be the associated reproducing kernel Hilbert space (RKHS). Assume k is bounded by K, meaning $\forall x$, $\|k(\cdot,x)\|_{\mathcal{H}} \leq K$. Let $\mathcal{G}^k_{K,R}$ denote the ball of radius R in $\mathcal{H}$. Define $\mathcal{G}_1 = \mathcal{G}^k_{K,R} \times \mathcal{G}^k_{K,R} \times \cdots \times \mathcal{G}^k_{K,R}$ (C copies).*

We follow Zhang et al. [54] and define real-valued MLPs inductively:

Definition 10. *Define $\mathcal{N}_1 = \big\{x \to \langle x, v\rangle : v \in \mathbb{R}^d, \|v\|_2 \leq \beta\big\}$, and for $m > 2$, inductively define $\mathcal{N}_m = \Big\{x \to \sum_{j=1}^d v_j \mu(f_j(x)) : v \in \mathbb{R}^d, \|v\|_1 \leq \beta, f_j \in \mathcal{N}_{m-1}\Big\}$, where $\beta \in \mathbb{R}_+$ and μ is a 1-Lipschitz activation function. Define an MLP which outputs a vector in $\mathbb{R}^C$ by $\mathcal{G}_2 = \mathcal{N}_m \times \mathcal{N}_m \times \cdots \times \mathcal{N}_m$ (C copies). We additionally assume that the choice of μ satisfies $\forall m \in \mathbb{N}, 0 \in \mu \circ \mathcal{N}_m$.*

Theorem 11. *Let T_1, T_2, …, T_N be invertible column-stochastic matrices. Let ℓ be a proper loss such that $\forall i, c$ the function $\lambda_{\ell_{T_i}}(\cdot, c)$ is Lipschitz continuous $w.r.t.$ the 2-norm. Let S be the set of*

data points as defined in Section 4.1. Assume $\sup_{x\in\mathcal{X},g\in\mathcal{G}_q}\|g(x)\|_2 \leq A_q$ *for some constant* A_q, $\forall q \in \{1,2\}$. *Let* $\hat{\mathcal{R}}_{w,S}$ *be as defined in equation (1).* $\widetilde{\mathcal{R}}(g) := \widetilde{R}_{\ell,P,\mathcal{T}}\big(\psi^{-1}\circ g\big) = \mathbb{E}_S\Big[\hat{\mathcal{R}}_{w,S}(g)\Big]$. *Then for each* $q \in \{1,2\}$, $\forall \delta \in [0,1]$, *with probability at least* $1-\delta$,

$$\sup_{g\in\mathcal{G}_q}\Big|\hat{\mathcal{R}}_{w,S}(g) - \widetilde{\mathcal{R}}(g)\Big| \leq (\max_i(\big|\lambda_{\ell_{T_i}}\big|A_q + \big|\lambda_{\ell_{T_i}}\big|_0)\sqrt{2\log\frac{2}{\delta}} + CB_q\max_i\big|\lambda_{\ell_{T_i}}\big|)\sqrt{\sum_{i=1}^{N}\sum_{c=1}^{C}\frac{w_i^2}{n_{i,c}}}.$$

where B_q *is a constant depending on* $\mathcal{G}_q$, $\big|\lambda_{\ell_{T_i}}\big|_0 = \max_c\big|\lambda_{\ell_{T_i}}(0,c)\big|$, *and* $\big|\lambda_{\ell_{T_i}}\big|$ *is the smallest real number such that it is a Lipschitz constant of* $\lambda_{\ell_{T_i}}(\cdot,c)$ *for all* c.

Theorem 11 is a special case of of Lemma 26 which extends the notion of Rademacher complexity to the LMNTM setting and applies to arbitrary function classes. Lemma 26 is presented in the appendix. Let HM_i denote the harmonic mean of $n_{i,1}, \ldots, n_{i,C}$, *i.e.*, $HM_i = \frac{C}{\sum_{c=1}^{C}\frac{1}{n_{i,c}}}$. The term $\sqrt{\sum_{i=1}^{N}\sum_{c=1}^{C}\frac{w_i^2}{n_{i,c}}}$ could be written as $\sqrt{C\sum_{i=1}^{N}\frac{w_i^2}{HM_i}}$ and is optimized by $w_i = HM_i/\sum_{m=1}^{N}HM_m$, leading to $\sqrt{\sum_{i=1}^{N}\sum_{c=1}^{C}\frac{w_i^2}{n_{i,c}}} = \sqrt{\frac{C}{\sum_{i=1}^{N}HM_i}}$. The term $\sqrt{\frac{C}{\sum_{i=1}^{N}HM_i}}$ vanishes (needed to establish consistency) when N goes to infinity, or when $\exists i\ s.t.\ \forall c$, $n_{i,c}$ goes to infinity. For the special case where all bags have the same size n and all weights w_i are $1/N$, $\sqrt{\sum_{i=1}^{N}\sum_{c=1}^{C}\frac{w_i^2}{n_{i,c}}} = \sqrt{\frac{C}{Nn}}$. Thus, consistency is possible even if bag size remains bounded. Assuming ℓ is the log loss and ψ^{-1} is the softmax function, we next study the constants $\big|\lambda_{\ell_{T_i}}\big|$ and $\big|\lambda_{\ell_{T_i}}\big|_0$.

Proposition 12. *Let* ℓ *be the log loss,* ψ^{-1} *be the softmax function, and* T *be a column-stochastic matrix. Then* $|\lambda_{\ell_T}| \leq \sqrt{2}$.

The constant $|\lambda_{\ell_T}|_0 = \max_c|\lambda_{\ell_T}(0,c)| = \max_c -\log(\frac{1}{C}\sum_{j=1}^{C}t_{c,j})$. The invertibility of T guarantees $\sum_{j=1}^{C}t_{c,j}$ is positive and hence the finiteness of $|\lambda_{\ell_T}|_0$. However, if we have a "bad" T, $\sum_{j=1}^{C}t_{c,j}$ could be arbitrarily close to 0 leading to a large $|\lambda_{\ell_T}|_0$.

Following Theorem 11, if the function class $\mathcal{G}$ has a universal approximation property, such as an RKHS associated to a universal kernel, or an MLP with increasing number of nodes, consistency for LMNTM via (regularized) minimization of $\hat{\mathcal{R}}_{w,S}(g)$ can be shown by leveraging standard techniques, provided $N \to \infty$ (bag size may remain bounded). Then the excess risk bound in Theorem 7 would automatically imply consistency with respect to 0-1 loss.

5 The LLPFC algorithms

In this section, we define a probabilistic model for LLP, show how LLP reduces to LMNTM, and introduce algorithms that we refer to as the LLPFC algorithms.

5.1 Probabilistic Model for LLP

Given a measure P on the space $\mathcal{X}\times\mathcal{Y}$, let $\{P_c : c\in\mathcal{Y}\}$ denote the class-conditional distributions of $\mathcal{X}$, *i.e.*, $\forall$ events $\mathcal{A}\subset\mathcal{X}, P_c(\mathcal{A}) = P(\mathcal{A}\mid Y=c)$. Let $\sigma(c) = P(Y=c), \forall c\in\mathcal{Y}$ and call $\sigma = (\sigma(1),\ldots,\sigma(C))$ the clean prior. Assume $\forall c\in\mathcal{Y}, \sigma(c)\neq 0$. Given $z = (z(1),\ldots,z(C)) \in \Delta^C$, let P_z be the probability measure on $\mathcal{X}\times\mathcal{Y}$ s.t. $\forall$ events $\mathcal{A}\subset\mathcal{X}, \forall i\in\mathcal{Y}, P_z(\mathcal{A}\times\{i\}) = z(i)P_i(\mathcal{A})$. Thus P_z has the same class-conditional distributions as P but a variable prior z.

We first define a model for a single bag. Given $z\in\Delta^C$, we say that bag b is *governed* by $z\in\Delta^C$ if b is a collection of feature vectors $\big\{X_j : j\in\mathbb{N}_{|b|}\big\}$ annotated by label proportion $\hat{z} = (\hat{z}(1),\hat{z}(2),\ldots,\hat{z}(C))$, where $|b|$ denotes the cardinality of the bag, each X_j is paired with an unobserved label $Y_j\ s.t.\ (X_j,Y_j)\overset{iid}{\sim}P_z$, and $\hat{z}(c) = \frac{1}{|b|}\sum_{j=1}^{|b|}\mathbb{1}_{Y_j=c}$. Note $\mathbb{E}_{P_z}[\hat{z}] = z$ and $P_z(Y_j=c) = z(c)$. We think of z as the true label proportion and $\hat{z}$ as the empirical label proportion.

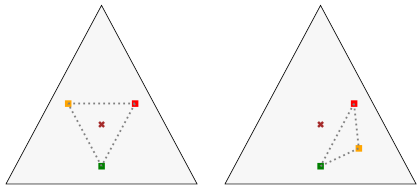

Figure 1: The gray triangle represents the probability simplex Δ^3. The squares represent γ_1, γ_2, and γ_3. The cross is σ. The ternary graph on the left visualizes an example where Assumption 13 holds. The one on the right visualizes an example where Assumption 13 fails.

Using this model for individual bags, we now formally state a model for LLP. Given bags $\{b_k\}$, let each b_k be governed by γ_k. Each b_k is a collection of feature vectors $\left\{X_j^k : j \in \mathbb{N}_{|b_k|}\right\}$ where $(X_j^k, Y_j^k) \overset{i.i.d.}{\sim} P_{\gamma_k}$ and Y_j^k is unknown. Further assume the X_j^k's are independent for all k and j. In practice, γ_k is unknown and we observe $\hat{\gamma}_k$ with $\hat{\gamma}_k(c) = \frac{1}{|b_k|}\sum_{j=1}^{|b_k|} \mathbb{1}_{Y_j^k = c}$ instead. The goal is learn an f that minimizes the risk $\mathcal{R}_{L,P} = \mathbb{E}_{(X,Y)\sim P}[L(f(X), Y)]$ where L is the 0-1 loss, given access to the training data $\{(b_k, \hat{\gamma}_k)\}$.

5.2 The Case of C Bags: Reduction to LLN

To explain our reduction of LLP to LLN, we first consider the case of exactly C bags $b_1, b_2, \ldots, b_C$, governed by respective (unobserved) $\gamma_1, \ldots, \gamma_C \in \Delta^C$, and annotated with label proportions $\hat{\gamma}_1, \ldots \hat{\gamma}_C$. Define $\Gamma \in \mathbb{R}^{C\times C}$ by $\Gamma(i,j) = \gamma_i(j)$, and let Γ^{tr} denote the transpose of Γ. Recall that σ is the class prior associated to P. To model LLP with C bags as an LLN problem, we make the following assumption on Γ and σ:

Assumption 13. *$\exists$ unique $\alpha \in \mathring{\Delta}^C$ s.t. $\Gamma^{tr}\alpha = \sigma$.*

We write $\alpha = (\alpha(1), \ldots, \alpha(C))$. Assumption 13 is equivalent to: $\{\gamma_1, \ldots, \gamma_C\}$ is a linearly independent set and σ is in the interior of the convex hull of $\{\gamma_1, \ldots, \gamma_C\}$. Ternary plots in Figure 5.2 visualize examples where assumption 13 holds and fails when $C = 3$. Intuitively, assumption 13 is more likely to hold when $\{\gamma_i : i \in \mathbb{N}_C\}$ are more "spread out" in Δ^C, in which case it is more likely for σ to reside in the convex hull of $\{\gamma_i : i \in \mathbb{N}_C\}$.

To reduce LLP with C bags to LLN, we simply propose to assign the "noisy label" $\tilde{Y} = i$ to all elements of bag b_i and to construct a noise transition matrix T with $T(i,j) = \gamma_i(j)\alpha(i)/\sigma(j)$. Assumption 13 ensures T is indeed a column-stochastic matrix. Thus, the probability measure $\bar{P}_T$ on $\mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$ satisfies $\alpha(i) = \bar{P}_T(\tilde{Y} = i)$ and $P_{\gamma_i}(\cdot) = \bar{P}_T(\cdot \mid \tilde{Y} = i)$, which further implies $\gamma_i(c) = \bar{P}_T(Y = c \mid \tilde{Y} = i)$. We confirm these facts in Section E in the appendix. Such construction transforms LLP with C bags into LLN with an estimable noise transition matrix T. Each element of a bag can then be viewed as a triplet $(X, Y, \tilde{Y})$, with Y unobserved, such that (X, Y) is drawn from $P_{\gamma_{\tilde{Y}}}$. After assigning the noisy labels, we have a dataset $\bigcup_{c=1}^{C}\left\{\left(X_j^c, c\right) : j \in \mathbb{N}_{|b_c|}\right\}$ along with the noise transition matrix T. This allows us to leverage the forward correction loss ℓ_T to minimize the objective $\mathcal{R}_{\ell_T, P_T}(f) = \mathbb{E}_{P_T}[\ell_T(f(X), \tilde{Y})]$ which can be estimated by the empirical risk $\sum_{c=1}^{C} \frac{\alpha(c)}{|b_c|}\sum_{j=1}^{|b_c|} \ell_T\left(f(X_j^c), c\right)$.

5.3 The General Case: Reduction to LMNTM

More generally, consider LLP with NC bags, $N \in \mathbb{N}$. We propose to randomly partition the bags into N groups, each with C bags indexed from 1 to C. Let $k_{i,c}$ denote the index of the c-th bag in the i-th group. Thus, $b_{k_{i,c}}$ is the c-th bag in the i-th group and it is governed by $\gamma_{k_{i,c}}$. For $i \in \mathbb{N}_N$, define the matrix $\Gamma_i \in \mathbb{R}^{C\times C}$ by $\Gamma_i(c_1, c_2) = \gamma_{k_{i,c_1}}(c_2)$, $\forall c_1, c_2 \in \mathcal{Y}$. We make the following assumption on the Γ_i's and σ:

Assumption 14. *For each $i \in \mathbb{N}_N$, $\exists$ unique $\alpha_i \in \mathring{\Delta}^C$ s.t. $\Gamma_i^{tr}\alpha_i = \sigma$.*

Thus, every group i can be modeled as above as an LLN problem with noise transition matrix T_i where $T_i(c_1, c_2) = \gamma_{k_{i,c_1}}(c_2)\alpha_i(c_1)/\sigma(c_2)$. Data points in the bag assigned with noisy label c in the i-th group can be viewed as drawn $i.i.d.$ from the class conditional distribution $P_{T_i}(\cdot \mid c)$. This

Algorithm 1 LLPFC-ideal

Input: $\{(b_k, \gamma_k)\}_{k=1}^{NC}$ and $w \in \Delta^N$ where $b_k = \{X_j^k : j \in \mathbb{N}_{|b_k|}\}$.
Randomly partition the bags into N groups $\{G_i\}_{i=1}^N$ $s.t.$ $G_i = \{(b_{k_{i,c}}, \gamma_{k_{i,c}}) : c \in \mathcal{Y}\}$
for $i = 1 : N$ **do**
  $\Gamma_i \leftarrow [\gamma_{k_{i,1}}, \gamma_{k_{i,2}}, \ldots, \gamma_{k_{i,C}}]^{tr}$
  $\alpha_i \leftarrow \Gamma_i^{-tr}\sigma$
  for $c_1 = 1 : C, c_2 = 1 : C$ **do**
    $T_i(c_1, c_2) \leftarrow \gamma_{k_{i,c_1}}(c_2)\alpha_i(c_1)/\sigma(c_2)$
  end for
end for
Train f with the empirical objective (1)

Algorithm 2 LLPFC-uniform

Input: $\{(b_k, \hat{\gamma}_k)\}_{k=1}^{NC}$ and $w \in \Delta^N$ where $b_k = \{X_j^k : j \in \mathbb{N}_{|b_k|}\}$.
Partition the bags as step 2 in Algorithm 1.
for $i = 1 : N$ **do**
  $\hat{\Gamma}_i \leftarrow [\hat{\gamma}_{k_{i,1}}, \hat{\gamma}_{k_{i,2}}, \ldots, \hat{\gamma}_{k_{i,C}}]^{tr}$
  $n_i \leftarrow \sum_{c=1}^{C} |b_{k_{i,c}}|$
  $\hat{\alpha}_i(c) \leftarrow |b_{k_{i,c}}|/n_i$ for each $c = 1 : C$
  $\hat{\sigma}_i \leftarrow \hat{\Gamma}_i^{tr}\hat{\alpha}_i$
  for $c_1 = 1 : C, c_2 = 1 : C$ **do**
    $\hat{T}_i(c_1, c_2) \leftarrow \hat{\gamma}_{k_{i,c_1}}(c_2)\hat{\alpha}_i(c_1)/\hat{\sigma}_i(c_2)$
  end for
end for
Train with $\sum_{i,c} \frac{w_i}{n_i} \sum_j \ell_{\hat{T}_i}(f(X_j^{k_{i,c}}), c)$.

problem now maps directly to LMNTM as described in Section 4.1, and satisfies the associated performance guarantees. In the next subsection, we spell out the associated algorithm.

5.4 Algorithms

As above, assume we have NC bags where $N \in \mathbb{N}$. Let each bag b_k be governed by $\gamma_k \in \Delta^C$ and be annotated by label proportion $\hat{\gamma}_k$. We first present the LLPFC-ideal algorithm in an ideal setting where σ, the γ_k's and the α_i's are known precisely and Assumption 14 holds. We then present the real-world adaptations LLPFC-uniform and LLPFC-approx in practical settings.

The LLPFC-ideal algorithm is presented in Algorithm 1. We follow the idea in section 5.3 to partition the bags into N groups of C bags, and model each group as an LLN problem. In Algorithm 1, we assume γ_k and σ are known and Assumption 14 holds. The theoretical analysis in Section 4 is immediately applicable to the LLPFC-ideal algorithm. We partition the bags by uniformly randomly partitioning the set of indices $\mathbb{N}_{NC}$ into disjoint subsets $\{k_{i,c} : c \in \mathcal{Y}\}$, $i \in \mathbb{N}_N$, where $k_{i,c}$ denotes the index of the c-th bag in the i-th group. We denote the inverse transpose of Γ_i by ${\Gamma_i}^{-tr}$.

In practice, when γ_k is unknown, we replace γ_k with $\hat{\gamma}_k$ as a plug-in method. Hence, we work with $\hat{\sigma} = \frac{\sum_{k=1}^{NC}|b_k|\hat{\gamma}_k}{\sum_{k=1}^{NC}|b_k|}$ and $\hat{\Gamma}_i = [\hat{\gamma}_{k_{i,1}}, \hat{\gamma}_{k_{i,2}}, \ldots, \hat{\gamma}_{k_{i,C}}]^{tr}$ instead of σ and Γ_i in Algorithm 1, respectively. Here $\hat{\sigma}$ is the label proportion of all training data points and we use it as an estimate of the clean prior σ. Likewise, $\alpha_i = \Gamma_i^{-tr}\sigma$ in Algorithm 1 should be replaced with $\hat{\alpha}_i = \hat{\Gamma}_i^{-tr}\hat{\sigma}$ and we would like to use $\hat{\Gamma}_i$, $\hat{\sigma}$, and $\hat{\alpha}_i$ to calculate $\hat{T}_i$ as an estimate of T_i. For this to make sense, we need $\hat{\alpha}_i = \hat{\Gamma}_i^{-tr}\hat{\sigma} \in \mathring{\Delta}^C$, which is equivalent to $\hat{\sigma}$ being in the interior of the convex hull of $\{\hat{\gamma}_{k_{i,c}} : c \in \mathcal{Y}\}$ for all i. However, this may not be the case in practice. Thus, we consider two

Algorithm 3 LLPFC-approx

Input: $\{(b_k, \hat{\gamma}_k)\}_{k=1}^{NC}$ and $w \in \Delta^N$ where $b_k = \{X_j^k : j \in \mathbb{N}_{|b_k|}\}$.
$\hat{\sigma} \leftarrow \frac{\sum_{k=1}^{NC}|b_k|\hat{\gamma}_k}{\sum_{k=1}^{NC}|b_k|}$
Partition the bags as step 2 in Algorithm 1.
for $i = 1 : N$ **do**
  $\hat{\Gamma}_i \leftarrow [\hat{\gamma}_{k_{i,1}}, \hat{\gamma}_{k_{i,2}}, \ldots, \hat{\gamma}_{k_{i,C}}]^{tr}$
  $\hat{\alpha}_i \leftarrow \arg\min_{\alpha \in \Delta^C} ||\hat{\sigma} - \hat{\Gamma}_i^{tr}\alpha||_2^2$
  $\hat{\sigma}_i \leftarrow \hat{\Gamma}_i^{tr}\hat{\alpha}_i$
  for $c_1 = 1 : C, c_2 = 1 : C$ **do**
    $\hat{T}_i(c_1, c_2) \leftarrow \hat{\gamma}_{k_{i,c_1}}(c_2)\hat{\alpha}_i(c_1)/\hat{\sigma}_i(c_2)$
  end for
end for
Train with $\sum_{i,c} \frac{w_i\hat{\alpha}_i(c)}{|b_{k_{i,c}}|} \sum_j \ell_{\hat{T}_i}(f(X_j^{k_{i,c}}), c)$

heuristics to estimate $\hat{T}_i$ as real-world adaptations of the LLPFC-ideal algorithm. The first, called LLPFC-uniform, is presented in Algorithm 2 which sets $\hat{\alpha}_i$ by counting the occurrences of the noisy labels. This is motivated by our model wherein α_i is the noisy class prior for the i-th group. The second, called LLPFC-approx, is presented in Algorithm 3 and sets $\hat{\alpha}_i$ to be the solution of $\arg\min_{\alpha \in \Delta^C} ||\hat{\sigma} - \hat{\Gamma}_i \alpha||_2^2$. It should be noted that in both practical algorithms, we use a different $\hat{\sigma}_i$ as an estimate of σ for each group, to ensure that each $\hat{T}_i$ is a column-stochastic matrix. In experiments where we have $NC + k$ number of bags with $0 < k < C$, we can randomly resample NC number of bags and regroup them in every few epochs. Both real-world adaptations perform reasonably well in experiments.

6 Experiments

[1] We compare against three previous works that have studied LLP applying deep learning to image

Table 1: Test Accuracy for Wide ResNet-16-4

Data set	Method	32	64	128	256	512	1024	2048
CIFAR10	KL	.4255 ± .13	.6817 ± .16	.5346 ± .11	.3749 ± .14	.2938 ± .04	Out of RAM	Out of RAM
	LLPVAT	.4911 ± .15	.5137 ± .22	.4744 ± .12	.4423 ± .16	Out of RAM	Out of RAM	Out of RAM
	LLPFC-uniform	.7926 ± .01	**.7683 ± .02**	.7399 ± .02	.7381 ± .01	.7224 ± .01	.7182 ± .01	.6925 ± .03
	LLPFC-approx	**.7993 ± .00**	.7671 ± .01	**.7528 ± .01**	**.7404 ± .00**	**.7409 ± .02**	**.7205 ± .03**	**.7283 ± .02**
SVHN	KL	.2465 ± .10	.1152 ± .07	.1022 ± .03	.1294 ± .04	.1039 ± .04	Out of RAM	Out of RAM
	LLPVAT	.2675 ± .36	.1398 ± .08	.1004 ± .03	.1294 ± .04	Out of RAM	Out of RAM	Out of RAM
	LLPFC-uniform	**.9012 ± .02**	**.8855 ± .02**	.8760 ± .02	.8736 ± .01	.8681 ± .02	**.8709 ± .02**	.8717 ± .01
	LLPFC-approx	.8903 ± .02	.8844 ± .02	**.8815 ± .03**	**.8808 ± .01**	**.8771 ± .02**	.8701 ± .02	**.8738 ± .01**
EMNIST	KL	.8413 ± .04	.8637 ± .04	.9111 ± .00	.5361 ± .12	.0845 ± .01	.0826 ± .01	Out of RAM
	LLPVAT	.8254 ± .04	.9045 ± .02	**.9136 ± .00**	.5071 ± .09	.0859 ± .01	Out of RAM	Out of RAM
	LLPFC-uniform	**.9165 ± .01**	.9061 ± .01	.9015 ± .01	.8790 ± .03	.8886 ± .02	.8461 ± .05	.8817 ± .01
	LLPFC-approx	.9092 ± .01	**.9074 ± .01**	.9065 ± .00	**.8993 ± .01**	**.9048 ± .00**	**.8969 ± .01**	**.9007 ± .01**

Table 2: Test Accuracy for ResNet18

Data set	Method	32	64	128	256	512	1024	2048
CIFAR10	KL	.7837 ± .01	**.7565 ± .01**	.6918 ± .01	.6106 ± .04	.5696 ± .05	.5197 ± .04	.4576 ± .03
	LLPVAT	**.7907 ± .01**	.7499 ± .01	**.6946 ± .01**	**.6115 ± .03**	.5670 ± .04	.4881 ± .02	.4624 ± .02
	LLPFC-uniform	.6601 ± .01	.6310 ± .01	.5867 ± .01	.5603 ± .01	.5670 ± .01	.5623 ± .01	.5288 ± .03
	LLPFC-approx	.6567 ± .01	.6136 ± .01	.5997 ± .01	.5931 ± .02	**.6062 ± .01**	**.6169 ± .01**	**.5591 ± .04**
SVHN	KL	.1279 ± .06	.0716 ± .01	.3042 ± .30	.1026 ± .04	.2489 ± .24	.3123 ± .29	.2797 ± .12
	LLPVAT	.1279 ± .06	.1939 ± .27	.6962 ± .34	.3673 ± .39	.4003 ± .31	.3999 ± .33	.3736 ± .24
	LLPFC-uniform	.8823 ± .01	.8644 ± .01	.8433 ± .01	.8390 ± .01	.8360 ± .00	.8086 ± .02	.8188 ± .01
	LLPFC-approx	**.8824 ± .01**	**.8672 ± .01**	**.8570 ± .01**	**.8483 ± .01**	**.8492 ± .01**	**.8498 ± .01**	**.8534 ± .01**
EMNIST	KL	**.9319 ± .00**	.9295 ± .00	**.9306 ± .00**	.9269 ± .00	**.9267 ± .00**	**.9239 ± .00**	.9106 ± .01
	LLPVAT	.9308 ± .00	**.9299 ± .00**	.9299 ± .00	**.9281 ± .00**	.9248 ± .00	.9222 ± .00	**.9128 ± .00**
	LLPFC-uniform	.9144 ± .00	.8954 ± .00	.8744 ± .00	.8600 ± .00	.8448 ± .00	.8388 ± .01	.8245 ± .01
	LLPFC-approx	.9146 ± .00	.8998 ± .00	.8874 ± .00	.8764 ± .01	.8670 ± .00	.8736 ± .01	.8660 ± .01

Table 3: Test Accuracy for VGG16

Data set	Method	32	64	128	256	512	1024	2048
CIFAR10	KL	.2513 ± .11	.2130 ± .06	.1794 ± .04	.1160 ± .02	.1117 ± .01	.1221 ± .00	.1049 ± .01
	LLPVAT	.4634 ± .07	.2093 ± .03	.1399 ± .03	.1145 ± .02	.1172 ± .02	.1189 ± .00	Out of RAM
	LLPFC-uniform	**.7602 ± .00**	**.7372 ± .01**	**.7300 ± .01**	**.7226 ± .01**	**.7136 ± .01**	**.7111 ± .01**	**.7033 ± .03**
	LLPFC-approx	.7566 ± .00	.7310 ± .01	.7003 ± .01	.7004 ± .01	.6870 ± .03	.6857 ± .02	.6645 ± .03
SVHN	KL	.1277 ± .06	.0893 ± .04	.1054 ± .05	.1024 ± .05	.1104 ± .04	.0885 ± .01	.1372 ± .03
	LLPVAT	.1117 ± .05	.0736 ± .01	.1051 ± .05	.1023 ± .06	.1125 ± .04	.1061 ± .05	Out of RAM
	LLPFC-uniform	.4177 ± .15	.4708 ± .23	**.5402 ± .21**	**.1734 ± .11**	**.4249 ± .30**	**.5691 ± .27**	**.6869 ± .13**
	LLPFC-approx	**.4299 ± .28**	**.4994 ± .23**	.1091 ± .04	.1188 ± .05	.1903 ± .14	.4097 ± .17	.4429 ± .18
EMNIST	KL	.5952 ±.45	.2348 ±.22	.0974 ±.01	.0842 ±.02	.0702 ±.01	.0692 ±.01	.0597 ±.02
	LLPVAT	.8593 ± .16	.3329 ± .33	.1042 ± .01	.0833 ± .02	.0696 ± .01	.0711 ± .00	Out of RAM
	LLPFC-uniform	**.9311 ± .00**	.9279 ± .00	**.9258 ± .00**	**.9242 ± .00**	**.9239 ± .00**	**.9233 ± .00**	**.9220 ± .00**
	LLPFC-approx	.9310 ± .00	**.9280 ± .00**	.9249 ± .00	.9240 ± .00	.9227 ± .00	.9206 ± .00	.9205 ± .00

Table 4: Test Accuracy for LLPGAN architecture

Data set	Method	32	64	128	256	512	1024	2048
CIFAR10	LLPGAN	.3630 ± .01	.3133 ± .02	.3328 ± .03	.3363 ± .03	.3460 ± .03	.2824 ± .05	.2236 ± .08
	LLPFC-uniform	.6145 ± .01	.5826 ± .01	.5565 ± .03	.5452 ± .01	.5511 ± .02	.5358 ± .01	.5438 ± .03
	LLPFC-approx	**.6169 ± .01**	**.5875 ± .01**	**.5642 ± .02**	**.5687 ± .02**	**.5621 ± .03**	**.5610 ± .01**	**.5567 ± .02**
SVHN	LLPGAN	.2378 ± .24	.7135 ± .06	.7680 ± .04	.6058 ± .29	.4863 ± .22	.1725 ± .06	.1382 ± .04
	LLPFC-uniform	**.8800 ± .00**	**.8581 ± .01**	**.8480 ± .01**	.8393 ± .01	.8347 ± .01	.8258 ± .01	.8327 ± .01
	LLPFC-approx	.8779 ± .01	.8519 ± .01	.7061 ± .33	**.8453 ± .02**	**.8423 ± .01**	**.8386 ± .01**	**.8527 ± .01**

data: Dulac-Arnold et al. [9] study the KL loss described in the introduction, and a novel loss

[1] Code is available at `https://github.com/Z-Jianxin/LLPFC`

based on optimal transport. They find that KL performs just as well as the novel loss. Liu et al. [22] employ the KL loss within a generative adversarial framework (LLPGAN). Tsai and Lin [46] propose augmenting the KL loss with a regularizer from semi-supervised learning and show improved performance (LLPVAT). We compare both LLPFC-uniform and LLPFC-approx against the KL loss, LLPGAN, and LLPVAT to clearly establish which empirical objective is better. Recent papers on multiclass LLP for which code is not available were not included [23, 16].

We generate bags with fixed, equal sizes in $\{32, 64, 128, 256, 512, 1024, 2048\}$. To generate each bag, we first sample a label proportion γ from the uniform distribution on Δ^C. Then we sample data points from a benchmark dataset without replacement using a multinomial distribution with parameter γ. It should be noted that Tsai and Lin [46], Dulac-Arnold et al. [9], and Liu et al. [22] generate bags by shuffling all data points and making every B data points a bag where B is a fixed bag size. Their method is equivalent to sampling data points without replacement using a multinomial distribution with a fixed parameter $\gamma = \left\{\frac{1}{C}, \frac{1}{C}, \ldots, \frac{1}{C}\right\}$. As noted by Scott and Zhang [39], this leads to bags with very similar label proportions which makes the learning task much more challenging.

We repeat each experiment 5 times and report the mean test accuracy and standard deviation. All models are trained on a single Nvidia Tesla v100 GPU with 16GB RAM. In our implementation of LLPFC algorithms, the weight w is set to be $(\frac{1}{N}, \ldots, \frac{1}{N}) \in \Delta^N$ and our choice of the proper composite loss is the cross-entropy loss.

For the comparison against KL and LLPVAT, we perform experiments on three benchmark image datasets: the "letter" split of EMNIST [6], SVHN [29], and CIFAR10 [17]. To show that our approach is robust to the choice of architecture, we experiment with three different networks: Wide ResNet-16-4 [52], ResNet18 [11], and VGG16 [42]. We train these networks with the parameters suggested in the original papers. The test accuracies are reported in Tables 1, 2, and 3. Since convergence in the GAN framework is sensitive to the choice of architecture and hyperparameters, we compare LLPFC against LLPGAN using the architecture proposed in the original paper along with the hyperparameters suggested in their code[2]. It should be noted that for LLPFC we only use the discriminator for classification and did not use the generator to augment data. Since Liu et al. [22] only provide hyperparameters for colored images, we perform experiments on SVHN and CIFAR10 only. The test accuracies are reported in Table 4.

LLPFC-uniform and LLPFC-approx substantially outperform the competitors in a clear majority of settings. The experiment results clearly establish our methods as the state-of-the-art by a substantial margin. All three competitors perform gradient descent with minibatches of bags and the GPU at times runs out of memory when the bag size is large. Our implementation, which also uses stochastic optimization, does not suffer from this phenomenon. Full experimental details are in the appendix.

7 Conclusions and Future Work

We propose a theoretically supported approach to LLP by reducing it to learning with label noise and using the forward correction (FC) loss. An excess risk bound and generalization error analysis are established. Our approach outperforms leading existing methods in deep learning scenarios across multiple datasets and architectures. A limitation of our approach is that the theory makes an assumption that may not be verifiable in practice. Future research directions include optimizing the grouping of bags and adapting LLPFC to other objectives beyond accuracy.

Acknowledgement The authors were supported in part by the National Science Foundation under awards 1838179 and 2008074, and by the Department of Defense, Defense Threat Reduction Agency under award HDTRA1-20-2-0002.

[2]https://github.com/liujiabin008/LLP-GAN

References

[1] Ehsan M. Ardehaly and Aron Culotta. "Co-Training for Demographic Classification Using Deep Learning from Label Proportions". In: *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*. 2017, pp. 1017–1024.

[2] Gerda Bortsova, Florian Dubost, Silas Ørting, Ioannis Katramados, Laurens Hogeweg, Laura Thomsen, Mathilde Wille, and Marleen de Bruijne. "Deep Learning from Label Proportions for Emphysema Quantification". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*. Ed. by Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger. Cham: Springer International Publishing, 2018, pp. 768–776. ISBN: 978-3-030-00934-2.

[3] S. Chen, B. Liu, M. Qian, and C. Zhang. "Kernel K-means Based Framework for Aggregate Outputs Classification". In: *2009 IEEE International Conference on Data Mining Workshops*. 2009, pp. 356–361.

[4] Tao Chen, Felix X. Yu, Jiawei Chen, Yin Cui, Yan-Ying Chen, and Shih-Fu Chang. "Object-Based Visual Sentiment Concept Analysis and Application". In: *Proceedings of the 22nd ACM International Conference on Multimedia*. MM '14. Orlando, Florida, USA: Association for Computing Machinery, 2014, pp. 367–376. ISBN: 9781450330633. DOI: 10.1145/2647868.2654935. URL: https://doi.org/10.1145/2647868.2654935.

[5] Zhensong Chen, Zhiquan Qi, Bo Wang, Limeng Cui, Fan Meng, and Yong Shi. "Learning with label proportions based on nonparallel support vector machines". In: *Knowledge-Based Systems* 119 (2017), pp. 126–141.

[6] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. "EMNIST: an extension of MNIST to handwritten letters". In: *ArXiv* abs/1702.05373 (2017).

[7] Lucio Mwinmaarong Dery, Benjamin Nachman, Francesco Rubbo, and Ariel Schwartzman. "Weakly Supervised Classification For High Energy Physics". In: *Journal of Physics: Conference Series* 1085 (Sept. 2018), p. 042006. DOI: 10.1088/1742-6596/1085/4/042006. URL: https://doi.org/10.1088/1742-6596/1085/4/042006.

[8] Yongke Ding, Yuanxiang Li, and W. Yu. "Learning from label proportions for SAR image classification". In: *EURASIP Journal on Advances in Signal Processing* 2017 (2017), pp. 1–12.

[9] Gabriel Dulac-Arnold, Neil Zeghidour, Marco Cuturi, Lucas Beyer, and Jean-Philippe Vert. "Deep multi-class learning from label proportions". In: *ArXiv* abs/1905.12909 (2019).

[10] Jean Gallier and Jocelyn Quaintance. *Linear Algebra and Optimization with Applications to Machine Learning*. World Scientific, 2020.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[12] J. Hernández-González, I. Inza, Lorena Crisol-Ortíz, M. A. Guembe, M. J. Iñarra, and J. A. Lozano. "Fitting the data from embryo implantation prediction: Learning from label proportions". In: *Statistical Methods in Medical Research* 27.4 (2018), pp. 1056–1066.

[13] J. Hernández-González, I. Inza, and J. A. Lozano. "Learning Bayesian network classifiers from label proportions". In: *Pattern Recognition* 46.12 (2013), pp. 3425–3440.

[14] Olav Kallenberg. *Foundations of Modern Probability*. Springer, 2002.

[15] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1412.6980.

[16] Ryoma Kobayashi, Yusuke Mukuta, and Tatsuya Harada. *Risk Consistent Multi-Class Learning from Label Proportions*. 2022. DOI: 10.48550/ARXIV.2203.12836. URL: https://arxiv.org/abs/2203.12836.

[17] Alex Krizhevsky. "Learning Multiple Layers of Features from Tiny Images". In: *University of Toronto* (May 2012).

[18] Hendrik Kück and Nando de Freitas. "Learning about Individuals from Group Statistics". In: *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*. 2005, pp. 332–339.

[19] K. Lai, F. X. Yu, M. Chen, and S. Chang. "Video Event Detection by Inferring Temporal Instance Labels". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 2251–2258.

[20] Kuan-Ting Lai, Felix Yu, Ming-Syan Chen, and S. Chang. "Video Event Detection by Inferring Temporal Instance Labels". In: June 2014. DOI: 10.1109/CVPR.2014.288.

[21] Fan Li and Graham Taylor. "Alter-CNN: An Approach to Learning from Label Proportions with Application to Ice-Water Classification". In: *Neural Information Processing Systems Workshops (NIPSW) on Learning and privacy with incomplete data and weak supervision*. 2015.

[22] Jiabin Liu, Bo Wang, Zhiquan Qi, YingJie Tian, and Yong Shi. "Learning from Label Proportions with Generative Adversarial Networks". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper/2019/file/4fc848051e4459b8a6afeb210c3664ec-Paper.pdf.

[23] Jiabin Liu, Bo Wang, Xin Shen, Zhiquan Qi, and Yingjie Tian. "Two-stage Training for Learning from Label Proportions". In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. Ed. by Zhi-Hua Zhou. Main Track. International Joint Conferences on Artificial Intelligence Organization, Aug. 2021, pp. 2737–2743. DOI: 10.24963/ijcai.2021/377. URL: https://doi.org/10.24963/ijcai.2021/377.

[24] Nan Lu, Shida Lei, Gang Niu, Issei Sato, and Masashi Sugiyama. "Binary Classification from Multiple Unlabeled Datasets via Surrogate Set Classification". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, July 2021, pp. 7134–7144. URL: https://proceedings.mlr.press/v139/lu21c.html.

[25] Andreas Maurer. "A vector-contraction inequality for Rademacher complexities". In: *International Conference on Algorithmic Learning Theory*. Springer. 2016, pp. 3–17.

[26] Colin McDiarmid. "On the method of bounded differences". In: *Surveys in combinatorics* 141.1 (1989), pp. 148–188.

[27] Ron Meir and Tong Zhang. "Generalization Error Bounds for Bayesian Mixture Algorithms". In: *J. Mach. Learn. Res.* 4.null (Dec. 2003), pp. 839–860. ISSN: 1532-4435.

[28] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. "Learning with Noisy Labels". In: *Advances in Neural Information Processing Systems*. Ed. by C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger. Vol. 26. Curran Associates, Inc., 2013. URL: https://proceedings.neurips.cc/paper/2013/file/3871bd64012152bfb53fdf04b401193f-Paper.pdf.

[29] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. "Reading Digits in Natural Images with Unsupervised Feature Learning". In: *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*. 2011. URL: http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

[30] Giorgio Patrini, A. Rozza, A. Menon, R. Nock, and Lizhen Qu. "Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 2233–2241.

[31] Mark S Pinsker. *Information and information stability of random variables and processes*. Holden-Day, 1964.

[32] R. Poyiadzi, R. Santos-Rodriguez, and N. Twomey. "LABEL PROPAGATION FOR LEARNING WITH LABEL PROPORTIONS". In: *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*. 2018, pp. 1–6.

[33] Zhiquan Qi, Bo Wang, Fan Meng, and Lingfeng Niu. "Learning With Label Proportions via NPSVM". In: *IEEE Transactions on Cybernetics* 47 (2017), pp. 3293–3305.

[34] Novi Quadrianto, Alex J. Smola, Tiberio S. Caetano, and Quoc V. Le. "Estimating Labels from Label Proportions". In: *Proceedings of the 25th International Conference on Machine Learning*. ICML '08. Helsinki, Finland: Association for Computing Machinery, 2008, pp. 776–783. ISBN: 9781605582054. DOI: 10.1145/1390156.1390254. URL: https://doi.org/10.1145/1390156.1390254.

[35] Brendan van Rooyen and Robert C. Williamson. "A Theory of Learning with Corrupted Labels". In: *Journal of Machine Learning Research* 18.228 (2018), pp. 1–50.

[36] Stefan Rueping. "SVM Classifier Estimation from Group Probabilities". In: *Proceedings of the 27th International Conference on International Conference on Machine Learning*. ICML'10. Haifa, Israel: Omnipress, 2010, pp. 911–918. ISBN: 9781605589077.

[37] Rishi Saket. "Learnability of Linear Thresholds from Label Proportions". In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan. 2021. URL: https://openreview.net/forum?id=5BnaKeEwuYk.

[38] Rishi Saket, Aravindan Raghuveer, and Balaraman Ravindran. "On Combining Bags to Better Learn from Label Proportions". In: *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*. Ed. by Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera. Vol. 151. Proceedings of Machine Learning Research. PMLR, Mar. 2022, pp. 5913–5927. URL: https://proceedings.mlr.press/v151/saket22a.html.

[39] Clayton Scott and Jianxin Zhang. "Learning from Label Proportions: A Mutual Contamination Framework". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., 2020, pp. 22256–22267. URL: https://proceedings.neurips.cc/paper/2020/file/fcde14913c766cf307c75059e0e89af5-Paper.pdf.

[40] Yong Shi, Limeng Cui, Zhensong Chen, and Zhiquan Qi. "Learning from label proportions with pinball loss". In: *International Journal of Machine Learning and Cybernetics* 10 (2017), pp. 187–205.

[41] Yong Shi, Jiabin Liu, Zhiquan Qi, and Bo Wang. "Learning from label proportions on high-dimensional data". In: *Neural Networks* 103 (2018), pp. 9–18. ISSN: 0893-6080. DOI: https://doi.org/10.1016/j.neunet.2018.03.004. URL: https://www.sciencedirect.com/science/article/pii/S0893608018300893.

[42] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *International Conference on Learning Representations*. 2015.

[43] Ingo Steinwart. "How to Compare Different Loss Functions and Their Risks". In: *Constructive Approximation* 26 (2007), pp. 225–287.

[44] Marco Stolpe and Katharina Morik. "Learning from Label Proportions by Optimizing Cluster Model Selection". In: *Machine Learning and Knowledge Discovery in Databases*. Ed. by Dimitrios Gunopulos, Thomas Hofmann, Donato Malerba, and Michalis Vazirgiannis. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 349–364. ISBN: 978-3-642-23808-6.

[45] Tao Sun, Dan Sheldon, and Brendan O'Connor. "A Probabilistic Approach for Learning with Label Proportions Applied to the US Presidential Election". In: *2017 IEEE International Conference on Data Mining (ICDM)*. 2017, pp. 445–454. DOI: 10.1109/ICDM.2017.54.

[46] Kuen-Han Tsai and Hsuan-Tien Lin. "Learning from Label Proportions with Consistency Regularization". In: *Proceedings of The 12th Asian Conference on Machine Learning, ACML 2020, 18-20 November 2020, Bangkok, Thailand*. Ed. by Sinno Jialin Pan and Masashi Sugiyama. Vol. 129. Proceedings of Machine Learning Research. PMLR, 2020, pp. 513–528. URL: http://proceedings.mlr.press/v129/tsai20a.html.

[47] B. Wang, Z. Chen, and Z. Qi. "Linear Twin SVM for Learning from Label Proportions". In: *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. Vol. 3. 2015, pp. 56–59.

[48] Nik Weaver. *Lipschitz Algebras*. WORLD SCIENTIFIC, 1999. DOI: 10.1142/4100. eprint: https://www.worldscientific.com/doi/pdf/10.1142/4100. URL: https://www.worldscientific.com/doi/abs/10.1142/4100.

[49] Robert C. Williamson, Elodie Vernet, and Mark D. Reid. "Composite Multiclass Losses". In: *Journal of Machine Learning Research* 17.222 (2016), pp. 1–52. URL: http://jmlr.org/papers/v17/14-294.html.

[50] Felix Yu, Dong Liu, Sanjiv Kumar, Jebara Tony, and Shih-Fu Chang. "$\propto$SVM for Learning with Label Proportions". In: *Proceedings of the 30th International Conference on Machine Learning*. Ed. by Sanjoy Dasgupta and David McAllester. Vol. 28. Proceedings of Machine Learning Research 3. Atlanta, Georgia, USA: PMLR, June 2013, pp. 504–512. URL: http://proceedings.mlr.press/v28/yu13a.html.

[51] Felix X. Yu, Krzysztof Choromanski, Sanjiv Kumar, Tony Jebara, and Shih-Fu Chang. *On Learning from Label Proportions*. Tech. rep. arXiv:1402.5902. 2015.

[52] Sergey Zagoruyko and Nikos Komodakis. "Wide Residual Networks". In: *Proceedings of the British Machine Vision Conference (BMVC)*. Ed. by Edwin R. Hancock Richard C. Wilson and William A. P. Smith. BMVA Press, Sept. 2016, pp. 87.1–87.12. ISBN: 1-901725-59-6. DOI: 10.5244/C.30.87. URL: https://dx.doi.org/10.5244/C.30.87.

[53] Mingyuan Zhang, Jane Lee, and Shivani Agarwal. "Learning from Noisy Labels with No Change to the Training Process". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, July 2021, pp. 12468–12478. URL: https://proceedings.mlr.press/v139/zhang21k.html.

[54] Yuchen Zhang, Jason Lee, Martin Wainwright, and Michael I. Jordan. "On the learnability of fully-connected neural networks". In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. Ed. by Aarti Singh and Jerry Zhu. Vol. 54. Proceedings of Machine Learning Research. Fort Lauderdale, FL, USA: PMLR, Apr. 2017, pp. 83–91. URL: http://proceedings.mlr.press/v54/zhang17a.html.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Limitations are described in Section 7.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] All proofs are included in the appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code is included in the supplemental material. Datasets are public and we provide code to download them. We include a README file with instructions on how to reproduce experimental results.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] A concise description of experiments is in Section 6 in the main paper with full details in section A in the appendix.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] In table 1, 2, 3 and 4, we report the mean and standard deviation for 5 trials with different random seeds.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] We provide the hardware information in Section 6 but not the amount of compute. The computational time varies for different experimental settings. It would be too exhaustive to present them given that we have nearly 300 different settings.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] Yes. Creators of code, data, and models are all cited.
   (b) Did you mention the license of the assets? [N/A] We did not directly run experiments using others' code. However, we implement their algorithms with their code as a reference.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Our experiment code is included in the supplemental materials.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Experiment Details

A.1 Datasets

We perform experiments and compare against the KL loss of Dulac-Arnold et al. [9] and LLPVAT of Tsai and Lin [46] on three benchmark datasets of image classification: the "letter" split of EMNIST [6], SVHN [29], and CIFAR10 [17]. We also compare our methods against LLPGAN of Liu et al. [22] on SVHN and CIFAR10. To generate each bag, we first sample a label proportion γ from the uniform distribution on Δ^C and then sample data points without replacement using a multinomial distribution with parameter γ. The generated bags have fixed and equal sizes in $\{32, 64, 128, 256, 512, 1024, 2048\}$. For SVHN and CIFAR10, $32 \times 1280 = 40960$ data points are sampled for every bag size. For EMNIST, the number of sampled data points is $32 \times 3328 = 106496$.

A.2 Architecture and Hyperparameters

To compare our methods against the KL loss and LLPVAT, we train Wide ResNet-16-4 [52], ResNet18 [11], and VGG16 [42] with the hyperparameters suggested in the original papers. For the comparison against LLPGAN, we use the discriminator architecture proposed in Liu et al. [22] and the hyperparameters suggested in their code[3].It should be noted that KL, LLPVAT, and LLPGAN are all required to backpropagate on minibatches of bags and our method does not have such constraint. For all methods, to avoid overfitting, we apply a standard data augmentation procedure: 4 pixels with value 0 are padded on each side, and a crop of the original size is randomly sampled from the padded image or its horizontal flip.

A.2.1 Wide ResNet-16-4

For all datasets, we use SGD with Nesterov momentum with weight decay set to 0.0005, dampening to 0, and momentum to 0.9. The minibatch size is set to 128 for our method. On CIFAR, the initial learning rate is set to 0.01, which is divided by 5 at 60, 120 and 160 epochs, and the network is trained for total 200 epochs. On SVHN and EMNIST, the initial learning rate is set to 0.01, which is divided by 10 at 80 and 120 epochs, and the network is trained for 160 epochs. The dropout probability is 0.3 for CIFAR and 0.4 for both SVHN and EMNIST.

A.2.2 ResNet18

We use SGD and weight decay is set to 0.0001 and momentum to 0.9. The minibatch size is set to 128 for our method. The model is trained for 500 epochs for all datasets. The learning rate is initialized to be 0.1 and divided by 10 at 250 and 375 epochs.

A.2.3 VGG16

We use SGD and weight decay is set to 0.0005 and momentum to 0.9. The minibatch size is set to 256 for our method. Dropout ratio is set to 0.5 for the first two fully-connected layers. The learning rate was initially set to 0.01. We train the model for 74 epochs in total. In the original paper of VGG16 [42], the learning rate is decreased when validation accuracy stops improving and it is decreased 3 times in total. In our experiment, while we do not assume access to fully labeled validation dataset, we divide the learning rate by 10 at 19, 37, and 56 epochs.

A.2.4 LLPGAN's discriminator

The neural network is trained for 3000 epochs and optimized by Adam [15] with a learning rate 0.0003. The minibatch size is set to 128 for our method. The β_1 and β_2 parameters for Adam are set to be 0.5 and 0.999, respectively.

A.3 KL Loss

Recall $C \in \mathbb{N}$ denotes the number of classes and $\mathcal{X}$ denotes the feature space. Let f be a function that maps $\mathcal{X}$ to Δ^C, and $M \in \mathbb{N}$ be the total number of bags. Let $\{(b_i, \hat{\gamma}_i)\}_{i=1}^M$ be

[3] https://github.com/liujiabin008/LLP-GAN

the bags and empirical label proportions where $b_i = (X_1^i, X_2^i, \ldots, X_{m_i}^i)$ and m_i is the size of bag b_i. The KL loss of Dulac-Arnold et al. [9] seeks to minimize the empirical objective $\bar{\mathcal{L}}_{prop} = -\frac{1}{CM}\sum_{i=1}^{M}\sum_{c=1}^{C}\hat{\gamma}_i(c)\log\Big(\frac{1}{m_i}\sum_{j=1}^{m_i} f_c(X_j^i)\Big)$ over the function f in some space $\mathcal{F}_0$. The $\hat{\gamma}_i(c)$ is the label proportion of c-th entry in bag i and f_c is the c-th entry of the output of f. In practice, when $\mathcal{F}_0$ is softmax composed with neural networks of certain architecture, the objective is optimized by stochastic gradient descent (SGD) with "minibatches of bags". For a minibatch of size B, B bags $b_{i_1}, \ldots, b_{i_B}$ are sampled and SGD backpropagates the gradients of $\mathcal{L}_{prop}(b_{i_1}, \ldots, b_{i_B}) = -\frac{1}{CB}\sum_{k=1}^{B}\sum_{c=1}^{C}\hat{\gamma}_{i_k}(c)\log\Big(\frac{1}{m_{i_k}}\sum_{j=1}^{m_{i_k}} f_c(X_j^{i_k})\Big)$. In our experiments, we follow the code of Tsai and Lin [46] [4] and set $B = 2$ (Dulac-Arnold et al. [9] also use minibatches of bags but do not specify B). While optimizing neural networks with the KL loss on GPU nodes, the gradients of all data points in the minibatch of bags need to be stored in the GPU memory simultaneously. So KL loss can potentially exceed GPU memory when bag size increases. In this situation, we report *Out of RAM* in the tables.

A.4 LLPVAT

Let f and $\mathcal{L}_{prop}$ be defined as in A.3. Let D_{KL} denote the KL divergence. The LLPVAT algorithm of Tsai and Lin [46] computes the perturbed examples $\hat{x} = x + r_{adv}$ where

$$r_{adv} = \underset{r:\|r\|_2 \le \epsilon}{\arg\max}\, D_{KL}(f(x) \mid\mid f(x + r_{adv})).$$

Given a minibatch of B bags $b_{i_1}, \ldots, b_{i_B}$, their consistency loss is defined to be

$$\mathcal{L}_{cons}(b_{i_1}, \ldots, b_{i_B}) = \sum_{k=1}^{B}\frac{1}{|b_{i_k}|}\sum_{x \in b_{i_k}} D_{KL}(f(x) \mid\mid f(x + r_{adv})).$$

For each minibatch in the t-th epoch, the LLPVAT algorithm updates the parameters of neural networks with the gradients of the loss $\mathcal{L}(b_{i_1}, \ldots, b_{i_B}) = \mathcal{L}_{prop}(b_{i_1}, \ldots, b_{i_B}) + w(t)\mathcal{L}_{cons}(b_{i_1}, \ldots, b_{i_B})$ where $w(t)$ is a ramp-up function for increasing the weight of consistency regularization. Following the LLPVAT paper, we set ϵ to 1 for both SVHN and EMNIST and set ϵ to 6 for CIFAR10. We follow the code of Tsai and Lin [46] [5] to implement $w(t)$ and set minibatch size B to be 2. Like the KL loss, LLPVAT can potentially exceed GPU memory. In this situation, we report *Out of RAM* in the tables.

A.5 LLPGAN

Let bags b_i and $\bar{\mathcal{L}}_{prop}$ be defined as in subsection A.3. The LLPGAN model of Liu et al. [22] consists of a generator g and a discriminator f. The discriminator f is a convolutional neural network and we denote its convolutional layers as f_{conv}. The generator g maps a random noise to the image space and the discriminator maps an image to Δ^{C+1} where the fake images output by the generator are supposed to be classified as the $(C+1)$-th class. Let n be the total number of feature vectors in all bags, *i.e.*, $n = \sum_{i=1}^{M} m_i$ and $\left\{Z_i^j, i \in \mathbb{N}_M, j \in \mathbb{N}_{m_i}\right\}$ be random noise vectors sampled from a fixed distribution. The discriminator loss is defined as

$$\bar{\mathcal{L}}_D = -\sum_{i=1}^{M}\sum_{j=1}^{m_i}\frac{1}{m_i}\log\left(\sum_{c \le C} f_c(X_j^i)\right) - \frac{1}{n}\sum_{i=1}^{M}\sum_{j=1}^{m_i}\log\big(f_{C+1}(g(Z_j^i))\big)$$
$$- \frac{1}{MC}\sum_{i=1}^{M}\sum_{c=1}^{C}\sum_{j=1}^{m_i}\frac{\hat{\gamma}_i(c)}{m_i}\log\big(f_c(X_j^i)\big),$$

where the last term $-\frac{1}{MC}\sum_{i=1}^{M}\sum_{c=1}^{C}\sum_{j=1}^{m_i}\frac{\hat{\gamma}_i(c)}{m_i}\log\big(f_c(X_j^i)\big)$ is proposed as an upper bound of $\bar{\mathcal{L}}_{prop}$. The generator loss is defined as

$$\bar{\mathcal{L}}_G = \frac{1}{n}\sum_{i=1}^{M}\sum_{j=1}^{m_i}\big\|f_{conv}(X_j^i) - f_{conv}(g(Z_j^i))\big\|_2^2.$$

[4] https://github.com/kevinorjohn/LLP-VAT

[5] See footnote 4

Given a minibatch of B bags $b_{i_1}, \ldots, b_{i_B}$, the minibatch version of the discriminator loss is

$$\begin{aligned}\mathcal{L}_D(b_{i_1}, \ldots, b_{i_B}) = &-\sum_{k=1}^{B}\sum_{j=1}^{m_{i_k}} \frac{1}{m_{i_k}} \log\left(\sum_{c\le C} f_c(X_j^{i_k})\right) \\ &- \frac{1}{\sum_{k=1}^{B} m_{i_k}} \sum_{i=1}^{B}\sum_{j=1}^{m_{i_k}} \log\big(f_{C+1}(g(Z_j^{i_k}))\big) \\ &- \frac{1}{BC}\sum_{k=1}^{B}\sum_{c=1}^{C}\sum_{j=1}^{m_{i_k}} \frac{\hat{\gamma}_i(c)}{m_{i_k}} \log\big(f_c(X_j^{i_k})\big).\end{aligned}$$

The minibatch version of generator loss is

$$\mathcal{L}_G(b_{i_1}, \ldots, b_{i_B}) = \frac{1}{\sum_{k=1}^{B} m_{i_k}} \sum_{i=1}^{B}\sum_{j=1}^{m_{i_k}} \left\| f_{conv}(X_j^{i_k}) - f_{conv}(g(Z_j^{i_k}))\right\|_2^2.$$

So the training process of LLPGAN can be described as follows: in each epoch, for a minibatch $(b_{i_1}, \ldots, b_{i_B})$,

1. Sample random noise $Z_j^{i_k}$ for $k \in \mathbb{N}_B$ and $j \in \mathbb{N}_{m_{i_k}}$.
2. Fix g and perform gradient descent on parameters of f in $\mathcal{L}_D$.
3. Fix f and perform gradient descent on parameters of g in $\mathcal{L}_G$.

The code [6] of the original paper implements f as a neural network which outputs a vector in $\mathbb{R}^C$ followed by a $(C+1)$-way softmax with the $(C+1)$-th input fixed to be 0. We follow this practice in our implementation of LLPGAN. We also follow the code of original paper and set B to be 1. While LLPGAN could potentially exceed GPU memory while bag size increases as well, this did not happen in our experiments.

A.6 Implementation Details of LLPFC

The performance of LLPFC algorithms benefit from re-partitioning of bags periodically. We randomly repartition the bags into groups every 20 epochs for WideResNet-16-4, ResNet-18, and LLPGAN discriminator and every 5 epochs for VGG16.

A.7 Experiments in Binary Setting

We carry out an extra set of experiments with kernel methods on binary classification tasks, comparing against InvCal [36], alter-$\propto$SVM [50], and LMMCM [39]. We run our experiments on the exact same datasets used by Scott and Zhang [39] and directly compare against the results presented in their paper. We implement LLPFC-uniform and LLPFC-approx with rbf kernel models and logistic loss by modifying the code provided by Scott and Zhang [39] at `https://github.com/Z-Jianxin/Learning-from-Label-Proportions-A-Mutual-Contamination-Framework`. We run experiments in the same settings of Scott and Zhang [39]. The model is solved by L-BFGS. We compute the kernel parameter by $\frac{1}{d*Var(X)}$ where d is the number of features and $Var(X)$ is the variance of the data matrix. The regularization parameter $\lambda \in \{1, 10^{-1}, 10^{-2}, \ldots, 10^{-5}\}$ is chosen by 5-fold cross validation, using the empirical risk provided in Algorithm 2 and Algorithm 3, respectively. We evaluate the area under the ROC curve (AUC) and report the results in table 5. We bold the largest mean AUC for that experimental setting. Each of LLPFC-uniform, LLPFC-approx, and LMMCM achieves the highest AUC among all the methods in 5 settings. LLPFC-uniform also beats the three competitors from Scott and Zhang [39] in 10 out 16 settings.

[6] See footnote 3

Table 5: AUC. Column header indicates bag size.

Data set, LP dist	Method	8	32	128	512
Adult, $\left[0, \frac{1}{2}\right]$	InvCal	0.8720 ± 0.0035	0.8672 ± 0.0067	0.8537 ± 0.0101	0.7256 ± 0.0159
	alter-$\propto$SVM	0.8586 ± 0.0185	0.7394 ± 0.0686	0.7260 ± 0.0953	0.6876 ± 0.1219
	LMMCM	0.8728 ± 0.0019	$\mathbf{0.8693 \pm 0.0047}$	$\mathbf{0.8669 \pm 0.0041}$	$\mathbf{0.8674 \pm 0.0040}$
	LLPFC-uniform	$\mathbf{0.8751 \pm 0.0022}$	0.8627 ± 0.0034	0.8616 ± 0.0057	0.8594 ± 0.0047
	LLPFC-approx	0.8676 ± 0.0042	0.8540 ± 0.0052	0.8509 ± 0.0094	0.8478 ± 0.0096
Adult, $\left[\frac{1}{2}, 1\right]$	InvCal	0.8680 ± 0.0021	0.8598 ± 0.0073	0.8284 ± 0.0093	0.7480 ± 0.0500
	alter-$\propto$SVM	0.8587 ± 0.0097	0.7429 ± 0.1473	0.8204 ± 0.0318	0.7602 ± 0.1215
	LMMCM	0.8584 ± 0.0164	0.8644 ± 0.0052	0.8601 ± 0.0045	0.8500 ± 0.0186
	LLPFC-uniform	0.8693 ± 0.0036	$\mathbf{0.8666 \pm 0.0047}$	$\mathbf{0.8636 \pm 0.0040}$	$\mathbf{0.8587 \pm 0.0136}$
	LLPFC-approx	$\mathbf{0.8723 \pm 0.0014}$	0.8630 ± 0.0069	0.8560 ± 0.0103	0.8538 ± 0.0193
MAGIC, $\left[0, \frac{1}{2}\right]$	InvCal	$\mathbf{0.8918 \pm 0.0076}$	0.8574 ± 0.0079	0.8295 ± 0.0139	0.8133 ± 0.0109
	alter-$\propto$SVM	0.8701 ± 0.0026	0.7704 ± 0.0818	0.7753 ± 0.0207	0.6851 ± 0.1580
	LMMCM	0.8909 ± 0.0077	$\mathbf{0.8799 \pm 0.0113}$	$\mathbf{0.8753 \pm 0.0157}$	0.8734 ± 0.0092
	LLPFC-uniform	0.8575 ± 0.0644	0.8751 ± 0.0158	0.8715 ± 0.0066	$\mathbf{0.8761 \pm 0.0157}$
	LLPFC-approx	0.8829 ± 0.0135	0.8590 ± 0.0256	0.8721 ± 0.0054	0.8711 ± 0.0155
MAGIC, $\left[\frac{1}{2}, 1\right]$	InvCal	0.8936 ± 0.0066	0.8612 ± 0.0056	0.8180 ± 0.0092	0.8215 ± 0.0136
	alter-$\propto$SVM	0.8689 ± 0.0135	0.8219 ± 0.0218	0.8179 ± 0.0487	0.7949 ± 0.0478
	LMMCM	0.8911 ± 0.0083	0.8790 ± 0.0091	0.8684 ± 0.0046	0.8567 ± 0.0292
	LLPFC-uniform	0.8985 ± 0.0054	0.8851 ± 0.0113	0.8844 ± 0.0101	0.8765 ± 0.0113
	LLPFC-approx	$\mathbf{0.9011 \pm 0.0034}$	$\mathbf{0.8990 \pm 0.0122}$	$\mathbf{0.8882 \pm 0.0088}$	$\mathbf{0.8800 \pm 0.0114}$

B Proofs of Results from Section 3

B.1 Proof of Theorem 5

To prove Theorem 5, we employ the calibration framework of Steinwart [43]. The first lemma of this section establishes an instance of what Steinwart [43] refers to as *uniform calibration*, but in the LLN setting.

Lemma 15. *Let ℓ be a continuous strictly proper loss and T be an invertible column-stochastic matrix. Let L be the* 0-1 *loss. Then* $\forall \epsilon \in \mathbb{R}_+, \exists \delta \in \mathbb{R}_+, s.t. \forall x \in \mathcal{X}, \forall q \in \Delta^C$,

$$\mathcal{C}_{\ell_T, P_T, x}(q) < \mathcal{C}^*_{\ell_T, P_T, x} + \delta \implies \mathcal{C}_{L,P,x}(q) < \mathcal{C}^*_{L,P,x} + \epsilon.$$

Proof of Lemma 15. Write

$$\mathcal{C}_{1,x}(q) := \mathcal{C}_{L,P,x}(q) - \mathcal{C}^*_{L,P,x}$$
$$\mathcal{C}_{2,x}(q) := \mathcal{C}_{\ell_T, P_T, x}(q) - \mathcal{C}^*_{\ell_T, P_T, x}.$$

Let $\|\cdot\|$ be an arbitrary norm on $\mathbb{R}^C$, k and $j \in \{1, \ldots, C\}$ be such that $q_k = \max_i q_i$, and $\eta_j(x) = \max_i \eta_i(x)$. Then we have

$$\begin{aligned}
\mathcal{C}_{1,x}(q) &= (1 - \eta_k(x)) - (1 - \eta_j(x)) \\
&= \eta_j(x) - \eta_k(x) \\
&= (\eta_j(x) - q_k) + (q_k - \eta_k(x)) \\
&= \|\eta(x)\|_\infty - \|q\|_\infty + (q_k - \eta_k(x)) \\
&\leq \|\eta(x) - q\|_\infty + \|\eta(x) - q\|_1 \\
&\leq C_{\|\cdot\|} \|\eta(x) - q\|,
\end{aligned}$$

where $C_{\|\cdot\|} > 0$ is a constant depending on the norm $\|\cdot\|$ and the last inequality is implied by the equivalence of norms in finite dimensional space.
Hence,

$$\|\eta(x) - q\| < \frac{\epsilon}{C_{\|\cdot\|}} \implies \mathcal{C}_{1,x}(q) < \epsilon.$$

So it suffices to prove

$$\forall \delta_0 \in \mathbb{R}_+, \exists \delta \in \mathbb{R}_+ s.t. \forall x \in X, \forall q \in \Delta^C, \mathcal{C}_{2,x}(q) < \delta \implies \|q - \eta(x)\| < \delta_0.$$

To this end, assume its negation,

$$\exists \delta_0 \in \mathbb{R}_+, \forall \delta \in \mathbb{R}_+ s.t. \exists x \in X, \exists q \in \Delta^C, \mathcal{C}_{2,x}(q) < \delta \text{ and } \|q - \eta(x)\| \geq \delta_0.$$

Let $\delta_n = \frac{1}{n}$ for each $n \in \mathbb{N}$. We can obtain a sequence $\{(q_i, \eta(x_i))\}_{i=1}^{\infty} \subset K := \{(q,\eta) \in \Delta^C \times \Delta^C : \|q-\eta\| \geq \delta_0\}$ such that $\mathcal{C}_{2,x_i}(q_i) < \delta_i$ for all $i \in \mathbb{N}$. As K is compact, we can extract a convergent subsequence $\{(q_{i_k}, \eta(x_{i_k}))\}_{k=1}^{\infty}$. Let

$$\lim_k (q_{i_k}, \eta(x_{i_k})) = (q^*, \eta^*) \in K.$$

Write

$$\mathcal{C}_2(q,\eta) := \sum_{c=1}^{C} (T\eta)_c (\ell(Tq,c) - \ell(T\eta,c)),$$

so $\mathcal{C}_2$ is continuous and $\mathcal{C}_{2,x}(q) = \mathcal{C}_2(q, \eta(x))$. Therefore,

$$0 = \lim_k \mathcal{C}_{2,x_{i_k}}(q_{i_k}) = \lim_k \mathcal{C}_2(q_{i_k}, \eta(x_{i_k})) = \mathcal{C}_2(q^*, \eta^*),$$

which contradicts the strict properness of ℓ since $q^* \neq \eta^*$. □

This lemma establishes what may be viewed as a pointwise notion of consistency: For each fixed $x \in \mathcal{X}$, the target excess 0/1-inner risk (defined $w.r.t.\ P$) can be made arbitrarily small by making the surrogate excess ℓ_T-inner risk (defined $w.r.t.\ P_T$) sufficiently small.

Now let $\epsilon \in [0,\infty]$, and denote $A(\epsilon) :=$

$$\left\{\delta \in [0,\infty] : \forall x \in \mathcal{X}, \forall q \in \Delta^C,\ \mathcal{C}_{\ell_T,P_T,x}(q) < \mathcal{C}^*_{\ell_T,P_T,x} + \delta \implies \mathcal{C}_{L,P,x}(q) < \mathcal{C}^*_{L,P,x} + \epsilon\right\}.$$

Define the function

$$\delta : [0,\infty] \to [0,\infty],\ \ \delta(\epsilon) = \sup_{\delta \in A(\epsilon)} \delta.$$

The following properties immediately follow from the definition and Lemma 15:

- $\delta(0) = 0$ and $\delta(\epsilon) > 0$ if $\epsilon > 0$.
- $\delta(\cdot)$ is monotone non-decreasing.
- $A(\epsilon) = [0, \delta(\epsilon)]$

The function δ is a reasonable candidate for the sought after function θ, but it is not necessarily invertible since it might not be strictly increasing. To address this we introduce the following.

Definition 16. *Let $I \subset \mathbb{R}$ be an interval and let $g : I \to [0,\infty]$ be a function. Then the* Fenchel-Legendre biconjugate *$g^{**} : I \to [0,\infty]$ of g is the largest convex function $h : I \to [0,\infty]$ satisfying $h \leq g$.*

We are now prepared to prove theorem 17 of which theorem 5 is a direct corollary.

Theorem 17. *Let ℓ be a continuous strictly proper loss and T be an invertible column-stochastic matrix. Let L be the 0-1 loss and let $\delta(\cdot)$ be the function defined above. Assume $\mathcal{R}^*_{\ell_T,P_T} < \infty$. Then for all P,*

$$\forall f \in \mathcal{F}, \delta^{**}_{|[0,1]}\left(\mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,P}\right) \leq \mathcal{R}_{\ell_T,P_T}(f) - \mathcal{R}^*_{\ell_T,P_T}$$

where $\delta^{**}_{|[0,1]}$ denotes the Fenchel-Legendre biconjugate of the restriction $\delta_{|[0,1]}$.

Proof of Theorem 17. Write

$$\mathcal{C}_{1,x}(q) := \mathcal{C}_{L,P,x}(q) - \mathcal{C}^*_{L,P,x}$$
$$\mathcal{C}_{2,x}(q) := \mathcal{C}_{\ell_T,P_T,x}(q) - \mathcal{C}^*_{\ell_T,P_T,x}.$$

Then,

$$\forall x \in \mathcal{X}, \forall p,q \in \Delta^C, \mathcal{C}_{2,x}(p) < \delta(\mathcal{C}_{1,x}(q)) \implies \mathcal{C}_{1,x}(p) < \mathcal{C}_{1,x}(q).$$

By letting $p = q$, we have $\forall x \in \mathcal{X}, \forall q \in \Delta^C, \mathcal{C}_{2,x}(q) \geq \delta(\mathcal{C}_{1,x}(q))$. Fix $f \in \mathcal{F}$,

$$\delta^{**}_{|[0,1]}\left(\mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,P}\right) = \delta^{**}_{|[0,1]}\left(\int_{\mathcal{X}} \mathcal{C}_{1,x}(f(x))dP_X(x)\right) \tag{2}$$

$$\leq \int_{\mathcal{X}} \delta^{**}_{|[0,1]}(\mathcal{C}_{1,x}(f(x)))dP_X(x) \tag{3}$$

$$\leq \int_{\mathcal{X}} \mathcal{C}_{2,x}(f(x))dP_X(x) \tag{4}$$

$$= \mathcal{R}_{\ell_T,P_T}(f) - \mathcal{R}^*_{\ell_T,P_T} \tag{5}$$

(2) follows the fact $\mathcal{R}_{L,P}(f) = \int_{\mathcal{X}} \mathcal{C}_{L,P,x}(f(x))dP_X(x)$ and $\mathcal{R}^*_{L,P} = \int_{\mathcal{X}} \mathcal{C}_{L,P,x}(\eta(x))dP_X(x) = \int_{\mathcal{X}} \mathcal{C}^*_{L,P,x}dP_X(x)$. (3) is implied by Jensen's inequality and the convexity of $\delta^{**}_{|[0,1]}$ and (4) by the fact $\delta^{**}_{|[0,1]}(\cdot) \leq \delta(\cdot)$. (5) follows $\mathcal{R}_{\ell_T,P_T}(f) = \int_{\mathcal{X}} \mathcal{C}_{\ell_T,P_T,x}(f(x))dP_X(x)$ and $\mathcal{R}^*_{\ell_T,P_T} = \int_{\mathcal{X}} \mathcal{C}_{\ell_T,P_T,x}(\eta(x))dP_X(x) = \int_{\mathcal{X}} \mathcal{C}^*_{\ell_T,P_T,x}dP_X(x) < \infty$. □

B.2 Proof of Proposition 6

For the reader's convenience, we restate Proposition 6 below:

Proposition 18. *Let $T \in \mathbb{R}^{C\times C}$ be an invertible, column-stochastic matrix. Define $\underline{\theta}_T : [0,\infty] \to [0,\infty]$ by*

$$\underline{\theta}_T(\epsilon) = \frac{1}{2}\frac{\epsilon^2}{\|T^{-1}\|_1^2}.$$

Then for all $x \in X$ and $q \in \Delta^C$, we have

$$\underline{\theta}_T(\mathcal{C}_{L,P,x}(q) - \mathcal{C}^*_{L,P,x}) \leq \mathcal{C}_{\ell_T,P_T,x}(q) - \mathcal{C}^*_{\ell_T,P_T,x}.$$

Below, let ℓ denote the *log loss* $\ell^{log}(q,c) = -\log q_c$ and L denote the $0-1$ loss:

$$L : \Delta^C \times Y \to \{0,1\}, \qquad L(q,c) = \mathbb{1}_{\{c\neq \min\{\arg\max q\}\}}.$$

To proceed with the proof, we first introduce some notations and useful results. For $p, q \in \Delta^C$, define

$$\underline{\ell}(q,p) := \mathbb{E}_{y\sim p}\ell(q,y) = \sum_{i=1}^{C} -p_i \log q_i. \tag{6}$$

The above quantity is often referred to as the *cross entropy* of q relative to p. Next, since the log loss is proper [49], we have

$$\inf_{q\in\Delta^C} \underline{\ell}(q,p) = \underline{\ell}(p,p). \tag{7}$$

The *Kullback-Leibler (KL) divergence* between $p, q \in \Delta^c$ is defined as

$$\mathtt{KL}(p\|q) = \underline{\ell}(p,q) - \underline{\ell}(p,p). \tag{8}$$

In the literature, the KL divergence is often presented as $\mathtt{KL}(p\|q) = \sum_{i=1}^{C} p_i \log\left(\frac{p_i}{q_i}\right)$ which is easily shown to be equivalent to (8). We now rewrite the right-hand side of the inequality in Proposition 18 in terms of the KL divergence:

Lemma 19. *Let $p := P(\cdot|x) \in \Delta^C$. Then*

$$\mathcal{C}_{\ell_T,P_T,x}(q) - \mathcal{C}^*_{\ell_T,P_T,x} = \mathtt{KL}(Tp\|Tq). \tag{9}$$

Proof of Lemma 19. By definition, we have $Tp = P_T(\cdot|x)$. Unwinding the definitions, we have

$$\mathcal{C}_{\ell_T,P_T,x}(q) = \mathbb{E}_{y\sim P_T(\cdot|x)}\ell_T(q,y) = \mathbb{E}_{y\sim Tp}\ell(Tq,y) = \underline{\ell}(Tq,Tp).$$

Furthermore,

$$\mathcal{C}^*_{\ell_T,P_T,x} = \inf_{q\in\Delta^C} \mathbb{E}_{y\sim P_T(\cdot|x)}\ell_T(q,y) = \inf_{q\in\Delta^C} \underline{\ell}(Tq,Tp) = \underline{\ell}(Tp,Tp)$$

where the last equality follows from (7). Now, (9) follows immediately from (8). □

Next, we focus on the term $\mathcal{C}_{L,P,x}(q) - \mathcal{C}^*_{L,P,x}$ on the left-hand side in Proposition 18. Analogous to (6), we define

$$\underline{L}(q,p) := \mathbb{E}_{y\sim p}L(q,y) = \sum_{c=1}^{C} p_c \mathbb{1}_{\{c\neq\min\{\arg\max q\}\}} = 1 - p_{\min\{\arg\max q\}}. \tag{10}$$

The $0-1$ loss is also proper and

$$\inf_{q\in\Delta^C} \underline{L}(q,p) = \underline{L}(p,p) = 1 - p_{\min\{\arg\max p\}} = 1 - \max_c p_c. \tag{11}$$

Unwinding the definition, we have

$$\mathcal{C}_{L,P,x}(q) = \mathbb{E}_{y\sim P(\cdot|x)} L(q,y) = \underline{L}(q,p)$$

and

$$\mathcal{C}^*_{L,P,x} = \inf_{q\in\Delta^C} \mathbb{E}_{y\sim P(\cdot|x)} L(q,y) = \inf_{q\in\Delta^C} \underline{L}(q,p) = \underline{L}(p,p).$$

Thus,

$$\mathcal{C}_{L,P,x}(q) - \mathcal{C}^*_{L,P,x} = \underline{L}(q,p) - \underline{L}(p,p). \tag{12}$$

Thus, by (9) and (12), we only need to focus on comparing $\underline{L}(q,p) - \underline{L}(p,p)$ with $\mathtt{KL}(Tp\|Tq)$. This is facilitated by the 1-norm $\|\cdot\|_1$ and the next two results. The first is by Pinsker [31]:

Theorem 20 (Pinsker inequality)**.** *Let $\|\cdot\|_1$ be the 1-norm on $\mathbb{R}^C$. Then for all $p, q \in \Delta^C$, we have*

$$\mathtt{KL}(p\|q) \geq \frac{1}{2}\|p-q\|_1^2.$$

The second one is widely-known in the literature. For the sake of completeness, we provide a proof using our notations:

Lemma 21. *Let $p, q \in \Delta^C$ be arbitrary. Then $\|p-q\|_1 \geq \underline{L}(q,p) - \underline{L}(p,p)$.*

Proof of Lemma 21. Let $i := \min\{\arg\max p\}$ and $j := \min\{\arg\max q\}$. Then by (10) and (11), we have

$$\underline{L}(q,p) - \underline{L}(p,p) = 1 - p_j - (1 - p_i) = p_i - p_j.$$

On the other hand, note that

$$\begin{aligned}
\|p-q\|_1 &= \sum_{i=1}^{C} |p_i - q_i| \\
&\geq |p_i - q_i| + |q_j - p_j| \\
&\geq |p_i - p_j + q_j - q_i| && \because \text{triangle inequality} \\
&= p_i - p_j + q_j - q_i && \because p_i - p_j \geq 0 \text{ and } q_j - q_i \geq 0 \\
&\geq p_i - p_j
\end{aligned}$$

as desired. □

Finally, we need one more result to take into account the presence of the stochastic matrix T when applying Pinsker inequality to lower bound $\mathtt{KL}(Tp\|Tq)$:

Lemma 22. *Let $M \in \mathbb{R}^{C\times C}$ be a matrix and let $\|\cdot\|$ be a norm on $\mathbb{R}^C$. Suppose that M is non-singular. Then*

$$\inf_{x\in\mathbb{R}^C : x\neq 0} \frac{\|Mx\|}{\|x\|} = \frac{1}{\|M^{-1}\|}.$$

Proof of Lemma 22. We begin by rewriting the infimum as the reciprocal of a supremum:

$$\inf_{x\in\mathbb{R}^C : x\neq 0} \frac{\|Mx\|}{\|x\|} = \left(\sup_{x\in\mathbb{R}^C : x\neq 0} \frac{\|x\|}{\|Mx\|} \right)^{-1}.$$

Next, applying the change of variables $x = M^{-1}y$, we have

$$\sup_{x\in\mathbb{R}^C : x\neq 0} \frac{\|x\|}{\|Mx\|} = \sup_{y\in\mathbb{R}^C : y\neq 0} \frac{\|M^{-1}y\|}{\|y\|} = \|M^{-1}\|$$

where the last equality holds by definition. □

Proof of Proposition 6. We are now ready to conclude the proof. Putting it all together, we have

$$\begin{aligned}
&\mathcal{C}_{\ell_T,P_T,x}(q) - \mathcal{C}^*_{\ell_T,P_T,x} \\
&= \mathtt{KL}(Tp\|Tq) \qquad \because \text{Equation (9)} \\
&\geq \frac{1}{2}\|Tp - Tq\|_1^2 \qquad \because \text{Theorem 20, Pinsker inequality} \\
&= \frac{1}{2}\|T(p-q)\|_1^2 \\
&\geq \frac{1}{2}\frac{\|p-q\|_1^2}{\|T^{-1}\|_1^2} \qquad \because \text{Lemma 22} \\
&\geq \frac{1}{2}\frac{(\underline{L}(q,p) - \underline{L}(p,p))^2}{\|T^{-1}\|_1^2}. \qquad \because \text{ Lemma 21} \\
&= \frac{1}{2}\frac{(\mathcal{C}_{L,P,x}(q) - \mathcal{C}^*_{L,P,x})^2}{\|T^{-1}\|_1^2} \qquad \because \text{Equation (12)}
\end{aligned}$$

as desired. This concludes the proof of Proposition 18. □

C Remakrs for Section 4

C.1 Remarks on the Setting of LMNTM

Instead of letting $\left(X, \tilde{Y}\right) \overset{i.i.d.}{\sim} P_{T_i}$, which is a more common assumption, we choose the setting described in Section 4.1 because it fits LLP more naturally. When reducing LLP to LLN, a bag in group i is modeled as a collection of data points sampled from $P_{T_i}(\cdot \mid c)$. If we assume all data points in group i are sampled $i.i.d.$ from P_{T_i}, then we need $(n_{i,1}, n_{i,2}, \ldots, n_{i,C})$, the size of bags in group i, to follow a multinomial distribution, which is too restrictive. Our current setting is more flexible and allows $n_{i,c}$ to be either deterministic or random.

D Proofs for Section 4

D.1 Proof of Theorem 7

Proof of Theorem 7. By Theorem 17, $\forall i \in \mathbb{N}, \exists$ a strictly increasing continuous function θ_i with $\theta_i(0) = 0$ and

$$\theta_i\big(\mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,Pg}\big) \leq \mathcal{R}_{\ell_{T_i},P_{T_i}}(f) - \mathcal{R}^*_{\ell_{T_i},P_{T_i}}.$$

Then,

$$\sum_{i=1}^N w_i\theta_i\big(\mathcal{R}_{L,P}(f) - \mathcal{R}^*_{L,P}\big) \leq \sum_{i=1}^N w_i\Big(\mathcal{R}_{\ell_{T_i},P_{T_i}}(f) - \mathcal{R}^*_{\ell_{T_i},P_{T_i}}\Big) = \widetilde{\mathcal{R}}_{\ell,P,\mathcal{T}}(f) - \widetilde{\mathcal{R}}^*_{\ell,P,\mathcal{T}}$$

The last equality is implied by the fact $\mathcal{R}^*_{\ell_{T_i},P_{T_i}} = \mathcal{R}_{\ell_{T_i},P_{T_i}}(\eta(x)) < \infty$. Let $\theta = \sum_{i=1}^N w_i\theta_i$ which is clearly continuous and satisfies $\theta(0) = 0$. □

D.2 Proof of Theorem 11

Now we introduce a sequence of lemmas to prove the generalization error bound.

Lemma 23. *Let $\mathcal{G} \subset \psi \circ \mathcal{F}$ s.t. $\sup_{x\in X, g\in\mathcal{G}}\|g(x)\|_2 \leq A$ for some constant A. Let $N \in \mathbb{N}$ and $\mathcal{T} = \{T_i\}_{i=1}^N$ be a sequence of invertible column-stochastic matrices. Fix $(w_1, w_2, \ldots, w_N)^{tr} \in \Delta^N$ and $n_{i,c} \in \mathbb{N}$ for each $i \in \mathbb{N}_N$ and $c \in \mathcal{Y}$. Let $S = \left\{X_{i,c,j} : i = \mathbb{N}_N, c \in \mathcal{Y}, j = \mathbb{N}_{n_{i,c}}\right\}$ where each $X_{i,c,j}$ is drawn from the class conditional distribution $P_{T_i}(\cdot \mid c)$ and all $X_{i,c,j}$'s are independent. $\forall i \in \mathbb{N}_N$ and $c \in \mathcal{Y}$, let $\alpha_i \in \mathring{\Delta}^C$ s.t. $\alpha_i(c) = P_{T_i}(\tilde{Y} = c)$. Let ℓ be a proper loss s.t. $\forall i, c$ the function $\lambda_{\ell_{T_i}}(\cdot, c)$ is Lipschitz w.r.t. the 2-norm. Write*

$$\hat{\mathcal{R}}_{w,S}(g) := \sum_{i=1}^N w_i \sum_{c=1}^C \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_i} \lambda_{\ell_{T_i}}(g(X_{i,c,j}), c))$$

and

$$\widetilde{\mathcal{R}}(g) := \widetilde{R}_{\ell,P,\mathcal{T}}\big(\psi^{-1} \circ g\big) = \mathbb{E}\Big[\hat{\mathcal{R}}_{w,S}(g)\Big].$$

Then $\forall \delta \in (0,1]$*, with probability at least* $1-\delta$ *w.r.t. to the draw of* S*,*

$$\sup_{g\in\mathcal{G}}\Big|\hat{\mathcal{R}}_{w,S}(g) - \widetilde{\mathcal{R}}(g)\Big| \leq \sqrt{2\log\frac{2}{\delta}\sum_{i=1}^{N}\sum_{c=1}^{C}\frac{w_i^2\alpha_i^2(c)}{n_{i,c}}\Big(\big|\lambda_{\ell_{T_i}}\big|A + \big|\lambda_{\ell_{T_i}}\big|_0\Big)^2}$$
$$+ 2\mathbb{E}_{S,\epsilon_{i,c,j}}\left[\sup_{g\in\mathcal{G}}\sum_{i=1}^{N} w_i \sum_{c=1}^{C}\frac{\alpha_i(c)}{n_{i,c}}\sum_{j=1}^{n_{i,c}}\epsilon_{i,c,j}\lambda_{\ell_{T_i}}(g(X_{i,c,j}),c)\right],$$

where $\epsilon_{i,c,j}, i = \mathbb{N}_N, c \in \mathcal{Y}, j \in \mathbb{N}_{n_{i,c}}$ *are i.i.d. Rademacher random variables,* $\big|\lambda_{\ell_{T_i}}\big|_0 = \max_c\big|\lambda_{\ell_{T_i}}(0,c)\big|$*, and* $\big|\lambda_{\ell_{T_i}}\big|$ *is the smallest real number such that it is a Lipschitz constant of* $\lambda_{\ell_{T_i}}(\cdot,c)$*,* $\forall i, c$*.*

Proof. Write

$$\xi(S) := \sup_{g\in\mathcal{G}}\Big|\hat{R}_{w,S}(g) - \widetilde{R}(g)\Big|,$$

$$\xi^+(S) := \sup_{g\in\mathcal{G}}\hat{R}_{w,S}(g) - \widetilde{R}(g), \quad \text{and} \quad \xi^-(S) := \sup_{g\in\mathcal{G}} -\Big(\hat{R}_{w,S}(g) - \widetilde{R}(g)\Big).$$

We will show that the same bound on $\xi^+(S)$ and $\xi^-(S)$ holds with probability at least $1-\frac{\delta}{2}$. Combining these bounds gives the desired bound on $\xi(S)$. We first consider $\xi^+(S)$. The analysis for $\xi^-(S)$ is identical. By definition,

$$\xi^+(S) = \sup_{g\in\mathcal{G}}\sum_{i=1}^{N} w_i\left[\sum_{c=1}^{C}\frac{\alpha_i(c)}{n_{i,c}}\sum_{j=1}^{n_i}\lambda_{\ell_{T_i}}(g(X_{i,c,j}),c) - \mathcal{R}_{\ell_{T_i},P_{T_i}}\big(\psi^{-1}\circ g\big)\right].$$

We first use the Bounded Difference Inequality [26] to bound $\xi^+(S) - \mathbb{E}\xi^+(S)$. Substitute $X_{i,c,j}$ with arbitrary $X'_{i,c,j}$ and $\xi^+(S)$ changes by at most $\sup_{g\in\mathcal{G}}\frac{w_i\alpha_i(c)}{n_{i,c}}\big|\lambda_{\ell_{T_i}}(g(X_{i,c,j}),c) - \lambda_{\ell_{T_i}}\big(g(X'_{i,c,j}),c\big)\big|$. Furthermore,

$$\begin{aligned}\big|\lambda_{\ell_{T_i}}(g(X_{i,c,j}),c)\big| &\leq \big|\lambda_{\ell_{T_i}}(g(X_{i,c,j}),c) - \lambda_{\ell_{T_i}}(0,c)\big| + \big|\lambda_{\ell_{T_i}}(0,c)\big| \\ &\leq \big|\lambda_{\ell_{T_i}}\big|\|g(X_{i,c,j})\| + \big|\lambda_{\ell_{T_i}}\big|_0 \\ &\leq \big|\lambda_{\ell_{T_i}}\big|A + \big|\lambda_{\ell_{T_i}}\big|_0.\end{aligned}$$

Hence,

$$\sup_{g\in\mathcal{G}}\frac{w_i\alpha_i(c)}{n_{i,c}}\big|\lambda_{\ell_{T_i}}(g(X_{i,c,j}),c) - \lambda_{\ell_{T_i}}\big(g(X'_{i,c,j}),c\big)\big| \leq 2\frac{w_i\alpha_i(c)}{n_{i,c}}\Big(\big|\lambda_{\ell_{T_i}}\big|A + \big|\lambda_{\ell_{T_i}}\big|_0\Big).$$

By the Bounded Difference Inequality, with probability at least $1-\frac{\delta}{2}$,

$$\xi^+(S) - \mathbb{E}\xi^+(S) \leq \sqrt{2\log\frac{2}{\delta}\sum_{i=1}^{N}\sum_{c=1}^{C}\frac{w_i^2\alpha_i(c)^2}{n_{i,c}}\Big(\big|\lambda_{\ell_{T_i}}\big|A + \big|\lambda_{\ell_{T_i}}\big|_0\Big)^2}.$$

It remains to bound $\mathbb{E}\xi^+(S)$. Let $S' = \left\{X'_{i,c,j}, : i = \mathbb{N}_N, c \in \mathcal{Y}, j = \mathbb{N}_{n_i}\right\}$ where every pair of $X'_{i,c,j}$ and $X_{i,c,j}$ are i.i.d. and all $X'_{i,c,j}$'s are independent. Hence,

$$
\begin{aligned}
\mathbb{E}_S[\xi(S)] &= \mathbb{E}_S\left[\sup_{g\in\mathcal{G}} \sum_{i=1}^{N} w_i \sum_{c=1}^{C} \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_{i,c}} \lambda_{\ell_{T_i}}(g(X_{i,c,j}), c) - \sum_{i=1}^{N} w_i \mathcal{R}_{\ell_{T_i}, P_{T_i}}(\psi^{-1} \circ g)\right] \\
&= \mathbb{E}_S\left[\sup_{g\in\mathcal{G}} \Big(\hat{\mathcal{R}}_{w,S}(g) - \mathbb{E}_{S'}\hat{\mathcal{R}}_{w,S'}(g)\Big)\right] \\
&\leq \mathbb{E}_S\mathbb{E}_{S'}\left[\sup_{g\in\mathcal{G}} \Big(\hat{\mathcal{R}}_{w,S}(g) - \hat{\mathcal{R}}_{w,S'}(g)\Big)\right] \qquad (13) \\
&= \mathbb{E}_{S,S'}\left[\sup_{g\in\mathcal{G}} \sum_{i=1}^{N} w_i \sum_{c=1}^{C} \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_{i,c}} \big(\lambda_{\ell_{T_i}}(g(X_{i,c,j}), c) - \lambda_{\ell_{T_i}}(g(X'_{i,c,j}), c)\big)\right] \\
&= \mathbb{E}_{S,S',\epsilon_{i,c,j}}\left[\sup_{g\in\mathcal{G}} \sum_{i=1}^{N} w_i \sum_{c=1}^{C} \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_{i,c}} \epsilon_{i,c,j}\big(\lambda_{\ell_{T_i}}(g(X_{i,c,j}), c) - \lambda_{\ell_{T_i}}(g(X'_{i,c,j}), c)\big)\right] \qquad (14) \\
&\leq \mathbb{E}_{S,S',\epsilon_{i,c,j}}\left[\sup_{g\in\mathcal{G}} \sum_{i=1}^{N} w_i \sum_{c=1}^{C} \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_{i,c}} \epsilon_{i,c,j}\lambda_{\ell_{T_i}}(g(X_{i,c,j}), c)\right] \\
&\quad + \mathbb{E}_{S,S',\epsilon_{i,c,j}}\left[\sup_{g\in\mathcal{G}} \sum_{i=1}^{N} w_i \sum_{c=1}^{C} \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_{i,c}} \epsilon_{i,c,j}\lambda_{\ell_{T_i}}(g(X'_{i,c,j}), c)\right] \qquad (15) \\
&= 2\mathbb{E}_{S,\epsilon_{i,c,j}}\left[\sup_{g\in\mathcal{G}} \sum_{i=1}^{N} w_i \sum_{c=1}^{C} \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_{i,c}} \epsilon_{i,c,j}\lambda_{\ell_{T_i}}(g(X_{i,c,j}), c)\right].
\end{aligned}
$$

(13) is implied by the convexity of $\sup_{g\in\mathcal{G}}$ and Jensen's inequality. The equality in (14) holds because $X'_{i,c,j}$ and $X_{i,c,j}$ are $i.i.d.$ and $\epsilon_{i,c,j}$ is symmetric. (15) can be justified by the elementary property of supremum and symmetry of $\epsilon_{i,c,j}$. □

We need the next two lemmas to get rid of the $\lambda_{\ell_{T_i}}$'s when the set $\mathcal{V} \subset \mathbb{R}^C$.

Lemma 24. *Let $\mathcal{H}$ be a set of functions from $\mathcal{X}$ to $\mathbb{R}^C$, let ϕ be a function from $\mathcal{H}$ to $\mathbb{R}$, let a be a positive real number, and let $\lambda : \mathbb{R}^C \to \mathbb{R}$ be a Lipschitz function $w.r.t.$ the norm $\|\cdot\|_2$. We denote the Lipschitz constant of λ by $|\lambda|$. Then,*

$$
\mathbb{E}_\epsilon \sup_{f\in\mathcal{H}} \epsilon a\lambda(f(x)) + \phi(f) \leq \mathbb{E}_{\epsilon_1,\ldots,\epsilon_C} \sup_{f\in\mathcal{H}} \sqrt{2}a|\lambda| \sum_{c=1}^{C} \epsilon_c f_c(x) + \phi(f)
$$

where $\epsilon, \epsilon_1, \ldots, \epsilon_C$ are independent Rademacher variables and $f_c(x)$ denotes the c-th entry of $f(x)$.

Proof. By Proposition 1 of Maurer [25],

$$
\forall M \in \mathbb{N}, \forall v \in \mathbb{R}^M, \|v\|_2 \leq \sqrt{2}\mathbb{E}_{\epsilon_m}\left|\sum_{m=1}^{M} v_m\epsilon_m\right|. \qquad (16)
$$

Fix $\delta > 0$, then $\exists f^*, g^* \in \mathcal{F}$,

$$
\begin{aligned}
& 2\left[\mathbb{E}_\epsilon \sup_{f\in\mathcal{H}} \epsilon a\lambda(f(x)) + \phi(f)\right] - \delta \\
&= \sup_{f,g\in\mathcal{H}} [a\lambda(f(x)) + \phi(f) - a\lambda(g(x)) + \phi(g)] - \delta \\
&< a(\lambda(f^*(x)) - \lambda(g^*(x))) + \phi(f^*) + \phi(g^*) && (17) \\
&\leq a|\lambda|\|f^*(x) - g^*(x)\|_2 + \phi(f^*) + \phi(g^*) \\
&\leq \mathbb{E}_{\epsilon_c} \sqrt{2}a|\lambda| \left|\sum_{c=1}^{C} \epsilon_c (f_c^*(x) - g_c^*(x))\right| + \phi(f^*) + \phi(g^*) && (18) \\
&\leq \mathbb{E}_{\epsilon_c} \sup_{f,g\in\mathcal{H}} \left[\sqrt{2}a|\lambda| \left|\sum_{c=1}^{C} \epsilon_c (f_c(x) - g_c(x))\right| + \phi(f) + \phi(g)\right] \\
&= \mathbb{E}_{\epsilon_c} \sup_{f\in\mathcal{H}} \left[\sqrt{2}a|\lambda| \sum_{c=1}^{C} \epsilon_c f_c(x) + \phi(f)\right] + \mathbb{E}_{\epsilon_c} \sup_{g\in\mathcal{H}} \left[-\sqrt{2}a|\lambda| \sum_{c=1}^{C} \epsilon_c g_c(x) + \phi(g)\right] && (19) \\
&= 2\mathbb{E}_{\epsilon_c} \sup_{f\in\mathcal{H}} \left[\sqrt{2}a|\lambda| \sum_{c=1}^{C} \epsilon_c f_c(x) + \phi(f)\right]
\end{aligned}
$$

The existence of f^*, g^* satisfying the inequality in step (17) is guaranteed by the definition of supremum. Step (18) is implied by (16). In (19), we drop the absolute value as we can make $\sum_{c=1}^{C} \epsilon_c (f_c(x) - g_c(x))$ non-negative by exchanging f and g for any realization of $\epsilon_c, c = 1, \ldots, C$. □

Now we move on to the next step.

Lemma 25. *Let $N, C \in \mathbb{N}$. Let $\mathcal{H}$ be a set of functions from $\mathcal{X}$ to $\mathbb{R}^C$. $\forall i = 1, \ldots, N$, let w_i be a positive real numbers, and let $\lambda_i : \mathbb{R}^C \to \mathbb{R}$ a Lipschitz function. Denote the Lipschitz constant of λ_i by $|\lambda_i|$. Then,*

$$
\mathbb{E}_{\epsilon_i} \sup_{f\in\mathcal{H}} \sum_{i=1}^{N} \epsilon_i w_i \lambda_i(f(x_i)) \leq \sqrt{2}\mathbb{E}_{\epsilon_{i,c}} \sup_{f\in\mathcal{H}} \sum_{i=1}^{N} w_i|\lambda_i| \sum_{c=1}^{C} \epsilon_{i,c} f_c(x_i)
$$

where ϵ_i's and $\epsilon_{i,c}$'s are independent Rademacher variables and $f_c(x)$ denotes the c-th entry of $f(x)$.

Proof. Let $m = 0, 1, \ldots, N$. We prove

$$
\mathbb{E}_{\epsilon_i} \sup_{f\in\mathcal{H}} \sum_{i=1}^{N} \epsilon_i w_i \lambda_i(f(x_i)) \leq
\mathbb{E}_{\epsilon_{i,c},\epsilon_i} \left[\sup_{f\in\mathcal{H}} \sqrt{2} \sum_{1\leq i\leq m} w_i|\lambda_i| \sum_{c=1}^{C} \epsilon_{i,c} f(x_i) + \sum_{m<i\leq N} \epsilon_i w_i \lambda_i(f(x_i))\right]
$$

by induction on m.

The base case when $m = 0$ holds with equality. The case when $m = N$ is the desired inequality. Now, suppose the inequality hold for $m - 1$.

$$
\begin{aligned}
&\mathbb{E}_{\epsilon_i} \sup_{f\in\mathcal{H}} \sum_{i=1}^{N} \epsilon_i w_i \lambda_i(f(x_i)) \\
&\leq \mathbb{E}_{\epsilon_{i,c},\epsilon_i}\left[\sup_{f\in\mathcal{H}} \sqrt{2} \sum_{1\leq i<m} w_i|\lambda_i| \sum_{c=1}^{C} \epsilon_{i,c} f_c(x_i) + \sum_{m\leq i\leq N} \epsilon_i w_i \lambda_i(f(x_i))\right] \\
&= \mathbb{E}_{\epsilon_{i,c},\epsilon_i}\left[\sup_{f\in\mathcal{H}} \epsilon_m w_m \lambda_m(f(x_m)) + \phi(f)\right] \\
&= \mathbb{E}_{\{\epsilon_{i,c},\epsilon_i|i\neq m\}}\mathbb{E}_{\epsilon_{m,c}}\left[\sup_{f\in\mathcal{H}} \epsilon_m w_m \lambda_m(f(x_m)) + \phi(f)\right] \\
&\leq \mathbb{E}_{\{\epsilon_{i,c},\epsilon_i|i\neq m\}}\mathbb{E}_{\epsilon_{m,c}}\left[\sup_{f\in\mathcal{H}} \sqrt{2} w_m|\lambda_m| \sum_{c=1}^{C} \epsilon_{m,c} f_c(x_m) + \phi(f)\right] \\
&= \mathbb{E}_{\epsilon_{i,c},\epsilon_i}\left[\sup_{f\in\mathcal{H}} \sqrt{2} \sum_{1\leq i\leq m} w_i|\lambda_i| \sum_{c=1}^{C} \epsilon_{i,c} f_c(x_i) + \sum_{m<i\leq N} \epsilon_i w_i \lambda_i(f(x_i))\right]
\end{aligned}
$$

In the first equality, we let $\phi(f)$ denote the rest of the summation. □

Lemma 26. *Let $\mathcal{G} \subset \psi \circ \mathcal{F}$ s.t. $\sup_{x\in X, g\in\mathcal{G}} \|g(x)\|_2 \leq A$ for some constant A. Let $N \in \mathbb{N}$ and $\mathcal{T} = \{T_i\}_{i=1}^{N}$ be a sequence of invertible column-stochastic matrices. Fix $(w_1, w_2, \ldots, w_N)^{tr} \in \Delta^N$ and $n_{i,c} \in \mathbb{N}$ for each $i \in \mathbb{N}_N$ and $c \in \mathcal{Y}$. Let $S = \left\{X_{i,c,j} : i = \mathbb{N}_N, c \in \mathcal{Y}, j = \mathbb{N}_{n_{i,c}}\right\}$ where each $X_{i,c,j}$ is drawn from the class conditional distribution $P_{T_i}(\cdot \mid c)$ and all $X_{i,c,j}$'s are independent. $\forall i \in \mathbb{N}_N$ and $c \in \mathcal{Y}$, let $\alpha_i \in \mathring{\Delta}^C$ s.t. $\alpha_i(c) = P_{T_i}(\tilde{Y} = c)$. Let ℓ be a proper loss s.t. $\forall i, c$ the function $\lambda_{\ell_{T_i}}(\cdot, c)$ is Lipschitz. Write*

$$
\hat{\mathcal{R}}_{w,S}(g) := \sum_{i=1}^{N} w_i \sum_{c=1}^{C} \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_i} \lambda_{\ell_{T_i}}(g(X_{i,c,j}), c))
$$

and

$$
\widetilde{\mathcal{R}}(g) := \widetilde{R}_{\ell,P,\mathcal{T}}\left(\psi^{-1} \circ g\right) = \mathbb{E}\left[\hat{\mathcal{R}}_{w,S}(g)\right].
$$

Then $\forall \delta \in (0, 1]$, with probability at least $1 - \delta$ w.r.t. to the draw of S,

$$
\begin{aligned}
\sup_{g\in\mathcal{G}}\left|\hat{\mathcal{R}}_{w,S}(g) - \widetilde{\mathcal{R}}(g)\right| \leq &\sqrt{2\log\frac{2}{\delta}\sum_{i=1}^{N}\sum_{c=1}^{C}\frac{w_i^2\alpha_i^2(c)}{n_{i,c}}\left(\left|\lambda_{\ell_{T_i}}\right|A + \left|\lambda_{\ell_{T_i}}\right|_0\right)^2} \\
&+ 2\mathbb{E}_{S,\epsilon_{i,c,j,c'}}\left[\sup_{g\in\mathcal{G}}\sum_{i=1}^{N} w_i\left|\lambda_{\ell_{T_i}}\right| \sum_{c=1}^{C}\frac{\alpha_i(c)}{n_{i,c}}\sum_{j=1}^{n_{i,c}}\sum_{c'=1}^{C}\epsilon_{i,c,j,c'} g_{c'}(X_{i,c,j})\right],
\end{aligned}
$$

where $\epsilon_{i,c,j,c'}, i \in \mathbb{N}_N, c \in \mathcal{Y}, c' \in \mathcal{Y}, j \in \mathbb{N}_{n_{i,c}}$ are i.i.d. Rademacher random variables, $\left|\lambda_{\ell_{T_i}}\right|_0 = \max_c\left|\lambda_{\ell_{T_i}}(0, c)\right|$, and $\left|\lambda_{\ell_{T_i}}\right|$ is the smallest real number such that it is a Lipschitz constant of $\lambda_{\ell_{T_i}}(\cdot, c)$, $\forall i, c$.

Proof of Theorem 26. The theorem is a direct result of Lemmas 23 and 25. □

In Theorem 7, we saw that $\widetilde{\mathcal{R}}(g)$ is a risk for LMNTM satisfying an excess risk bound. Lemma 26 shows that $\hat{\mathcal{R}}_{w,S}(g)$ is an accurate estimate of $\widetilde{\mathcal{R}}(g)$, and therefore justifies its use as an empirical objective for LMNTM.

The second term on the right hand side of the inequality in Lemma 26 depends on the choice of hypothesis class $\mathcal{G}$, and can be viewed as a generalization of Rademacher complexity to LMNTM. To

make this term more concrete, we study two popular choices of function classes, the reproducing kernel Hilbert space (RKHS) and the multilayer perceptron (MLP). We first consider the kernel class.

Proposition 27. *Let k be a symmetric positive definite (SPD) kernel, and let $\mathcal{H}$ be the associated reproducing kernel Hilbert space (RKHS). Assume k bounded by K, meaning $\forall x$, $\|k(\cdot,x)\|_{\mathcal{H}} \leq K$. Let $\mathcal{G}^k_{K,R}$ denote the ball of radius R in $\mathcal{H}$ and $\mathcal{G} = \underbrace{\mathcal{G}^k_{K,R} \times \mathcal{G}^k_{K,R} \times \cdots \times \mathcal{G}^k_{K,R}}_{\text{the Cartesian product of } C\ \mathcal{G}^k_{K,R}\text{'s}}$. Then*

$$\mathbb{E}_{S,\epsilon_{i,c,j,c'}}\left[\sup_{g\in\mathcal{G}}\sum_{i=1}^{N} w_i\big|\lambda_{\ell_{T_i}}\big|\sum_{c=1}^{C}\frac{\alpha_i(c)}{n_{i,c}}\sum_{j=1}^{n_{i,c}}\sum_{c'=1}^{C}\epsilon_{i,c,j,c'}g_{c'}(X_{i,c,j})\right] \leq$$

$$CRK\sqrt{\sum_{i=1}^{N} w_i^2\big|\lambda_{\ell_{T_i}}\big|^2\sum_{c=1}^{C}\frac{\alpha_i^2(c)}{n_{i,c}}},$$

where $\epsilon_{i,c,j,c'}, i\in\mathbb{N}_N, c\in\mathcal{Y}, c'\in\mathcal{Y}, j\in\mathbb{N}_{n_{i,c}}$ are i.i.d. Rademacher random variables. Thus the generalization error bound becomes: $\forall\delta\in[0,1]$, with probability at least $1-\delta$,

$$\sup_{g\in\mathcal{G}}\Big|\hat{\mathcal{R}}_{w,S}(g)-\widetilde{\mathcal{R}}(g)\Big| \leq$$

$$\left(\max_i\Big(\big|\lambda_{\ell_{T_i}}\big|A+\big|\lambda_{\ell_{T_i}}\big|_0\Big)\sqrt{2\log\frac{2}{\delta}}+CRK\max_i\big|\lambda_{\ell_{T_i}}\big|\right)\sqrt{\sum_{i=1}^{N} w_i^2\sum_{c=1}^{C}\frac{\alpha_i^2(c)}{n_{i,c}}}.$$

Proof of Proposition 27. For the reader's convenience, we restate the result:

Proposition 28. *Let k be a symmetric positive definite (SPD) kernel bounded by K and $\mathcal{H}$ be the associated reproducing kernel Hilbert space (RKHS). i.e.$\|k(\cdot,x)\|_{\mathcal{H}} \leq K$. Let $\mathcal{G}^k_{K,R}$ denote the ball of radius R in $\mathcal{H}$ and $\mathcal{G} = \underbrace{\mathcal{G}^k_{K,R} \times \mathcal{G}^k_{K,R} \times \cdots \times \mathcal{G}^k_{K,R}}_{\text{the Cartesian products of } C\ \mathcal{G}^k_{K,R}\text{'s}}$. Then*

$$\mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c\in\mathcal{G}^k_{K,R}}\sum_{i=1}^{M} a_i\sum_{c=1}^{C}\epsilon_{i,c}g_c(x_i)\right] \leq CRK\sqrt{\sum_{i=1}^{M} a_i^2}$$

where $a_i > 0$, and $\epsilon_{i,c}$ are independent Rademacher random variables.

Proof. First, by Cauchy-Schwartz inequality, observe $\forall R>0, g\in\mathcal{G}^k_{K,R}, x\in\mathcal{X}$

$$|g(x)| = |\langle g, k(\cdot,x)\rangle| \leq \|g\|_{\mathcal{H}}\|k(\cdot,x)\|_{\mathcal{H}} \leq RK.$$

Thus,

$$\begin{aligned}
&\mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c\in\mathcal{G}^k_{K,R}}\sum_{i=1}^{M} a_i\sum_{c=1}^{C}\epsilon_{i,c}g_c(x_i)\right] \\
&= \mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c\in\mathcal{G}^k_{K,R}}\sum_{i=1}^{M} a_i\sum_{c=1}^{C}\epsilon_{i,c}\langle g_c, k(\cdot,x_i)\rangle\right] && (20)\\
&= \mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c\in\mathcal{G}^k_{K,R}}\sum_{c=1}^{C}\langle g_c, \sum_{i=1}^{M} a_i\epsilon_{i,c}k(\cdot,x_i)\rangle\right] \\
&= \mathbb{E}_{\epsilon_{i,c}}\left[\sum_{c=1}^{C}\langle R\frac{\sum_{i=1}^{M} a_i\epsilon_{i,c}k(\cdot,x_i)}{\|\sum_{i=1}^{M} a_i\epsilon_{i,c}k(\cdot,x_i)\|}, \sum_{i=1}^{M} a_i\epsilon_{i,c}k(\cdot,x_i)\rangle\right] && (21)\\
&= R\sum_{c=1}^{C}\mathbb{E}_{\epsilon_{i,c}}\sqrt{\left\|\sum_{i=1}^{M} a_i\epsilon_{i,c}k(\cdot,x_i)\right\|^2}
\end{aligned}$$

$$\leq R \sum_{c=1}^{C} \sqrt{\mathbb{E}_{\epsilon_{i,c}} \left\| \sum_{i=1}^{M} a_i \epsilon_{i,c} k(\cdot, x_i) \right\|^2} \tag{22}$$

$$= CR \sqrt{\sum_{i=1}^{M} a_i^2 \left\| k(\cdot, x_i) \right\|^2} \tag{23}$$

$$= CRK \sqrt{\sum_{i=1}^{M} a_i^2}$$

Equality (20) and (21) follow the reproducing property and the equality condition of Cauchy-Schwarz, respectively. (22) is implied by Jensen's inequality and (23) by the independence of Rademacher random variables. □

We now define the Rademacher Complexity-like term $\mathbb{E}_{\epsilon_i} \sup_{g \in \mathcal{G}} \sum_{i=1}^{M} a_i \epsilon_i g(x_i)$ formally and characterize several properties which will be used in the proof of Proposition 31.

Definition 29. *Let $\mathcal{G}$ be a subset of measurable functions from $\mathcal{X}$ to $\mathbb{R}$. Denote the sample path $S = (x_i)_{i=1}^{M}$ and weights by $a = (a_i)_{i=1}^{M}$ where $a_i \geq 0$. Define*

$$\mathit{Rad}_{S,a}(\mathcal{G}) = \mathbb{E}_{\epsilon_i} \sup_{g \in \mathcal{G}} \sum_{i=1}^{M} a_i \epsilon_i g(x_i),$$

where ϵ_i's are i.i.d. Rademacher random variables.

Proposition 30. *$\mathit{Rad}_{S,a}$ has the following properties:*

1. $\mathcal{G} \subset \mathcal{H} \implies \mathit{Rad}_{S,a}(\mathcal{G}) \leq \mathit{Rad}_{S,a}(\mathcal{H})$
2. $\mathit{Rad}_{S,a}(\mathcal{G}_1 + \mathcal{G}_2) = \mathit{Rad}_{S,a}(\mathcal{G}_1) + \mathit{Rad}_{S,a}(\mathcal{G}_2)$, *where* $\mathcal{G}_1 + \mathcal{G}_2 = \{g_1 + g_2 : g_1 \in \mathcal{G}_1, g_2 \in \mathcal{G}_2\}$
3. $\forall c_0 \in \mathbb{R}, \mathit{Rad}_{S,a}(c_0 \mathcal{G}) = |c_0| \mathit{Rad}_{S,a}(\mathcal{G})$, *where* $c_0 \mathcal{G} := \{c_0 g : g \in \mathcal{G}\}$
4. $\mathit{Rad}_{S,a}(\operatorname{conv} \mathcal{G}) = \mathit{Rad}_{S,a}(\mathcal{G})$, *where* $\operatorname{conv} \mathcal{G}$ *denotes the convex hull of* $\mathcal{G}$.
5. *Let $\mu : \mathbb{R} \to \mathbb{R}$ be a Lipschitz function and let $|\mu|$ be its Lipschitz constant. Then,*

$$\mathit{Rad}_{S,a}(\mu \circ \mathcal{G}) \leq |\mu| \mathit{Rad}_{S,a}(\mathcal{G}), \text{ where } \mu \circ \mathcal{G} = \{\mu \circ g : g \in \mathcal{G}\}.$$

Proof. Property 1 and 2 immediately follow the definition. Property 3 is implied by the invariance of ϵ_i under negation. It remains to prove Property 4 and 5.

For Property 4:

$$\begin{aligned}
&\mathrm{Rad}_{S,a}(\operatorname{conv} \mathcal{G}) \\
&= \mathbb{E} \sup_{n \in \mathbb{N}} \sup_{\lambda \in \Delta^n, g_j \in \mathcal{G}} \sum_{i=1}^{M} a_i \epsilon_i \sum_{j=1}^{n} \lambda_j g_j(x_i) \\
&= \mathbb{E} \sup_{n \in \mathbb{N}} \sup_{\lambda \in \Delta^n, g_j \in \mathcal{G}} \sum_{j=1}^{n} \lambda_j \sum_{i=1}^{M} a_i \epsilon_i g_j(x_i) \\
&= \mathbb{E} \sup_{n \in \mathbb{N}} \sup_{\lambda \in \Delta^n, g_j \in \mathcal{G}} \max_{j} \sum_{i=1}^{M} a_i \epsilon_i g_j(x_i) \\
&= \mathbb{E} \sup_{g \in \mathcal{G}} \sum_{i=1}^{M} a_i \epsilon_i g(x_i) \\
&= \mathrm{Rad}_{S,a}(\mathcal{G}).
\end{aligned}$$

For Property 5, we follow the idea of Meir and Zhang [27],

$$
\begin{aligned}
&\mathrm{Rad}_{S,a}(\mu \circ \mathcal{G}) \\
&= \mathbb{E}_{\epsilon_i} \sup_{g \in \mathcal{G}} \sum_{i=1}^{M} a_i \epsilon_i (\mu \circ g)(x_i) \\
&= \mathbb{E}_{\epsilon_i, i=2,3,\ldots,M} \mathbb{E}_{\epsilon_1} \sup_{g \in \mathcal{G}} \sum_{i=1}^{M} a_i \epsilon_i (\mu \circ g)(x_i) \\
&= \frac{1}{2} \mathbb{E}_{\epsilon_i, i=2,3,\ldots,M} \left[\sup_{g \in \mathcal{G}} \left(a_1 (\mu \circ g)(x_1) + \sum_{i=2}^{M} a_i \epsilon_i (\mu \circ g)(x_i) \right) \right. \\
&\left. + \sup_{g' \in \mathcal{G}} \left(-a_1 (\mu \circ g')(x_1) + \sum_{i=2}^{M} a_i \epsilon_i (\mu \circ g')(x_i) \right) \right] \\
&= \frac{1}{2} \mathbb{E}_{\epsilon_i, i=2,3,\ldots,M} \left[\sup_{g, g' \in \mathcal{G}} a_1 (\mu(g(x_1)) - \mu(g'(x_1))) + \sum_{i=2}^{M} a_i \epsilon_i (\mu \circ (g + g'))(x_i) \right] \\
&\leq \frac{1}{2} \mathbb{E}_{\epsilon_i, i=2,3,\ldots,M} \left[\sup_{g, g' \in \mathcal{G}} a_1 |\mu| |g(x_1) - g'(x_1)| + \sum_{i=2}^{M} a_i \epsilon_i (\mu \circ (g + g'))(x_i) \right] \\
&= \frac{1}{2} \mathbb{E}_{\epsilon_i, i=2,3,\ldots,M} \left[\sup_{g, g' \in \mathcal{G}} a_1 |\mu| (g(x_1) - g'(x_1)) + \sum_{i=2}^{M} a_i \epsilon_i (\mu \circ (g + g'))(x_i) \right] \qquad (24) \\
&= \frac{1}{2} \mathbb{E}_{\epsilon_i, i=2,3,\ldots,M} \left[\sup_{g \in \mathcal{G}} \left(a_1 |\mu| g(x_1) + \sum_{i=2}^{M} a_i \epsilon_i (\mu \circ g)(x_i) \right) \right. \\
&\left. + \sup_{g' \in \mathcal{G}} \left(-a_1 |\mu| g'(x_1) + \sum_{i=2}^{M} a_i \epsilon_i (\mu \circ g')(x_i) \right) \right] \\
&= \mathbb{E}_{\epsilon_i} \sup_{g \in \mathcal{G}} \left[a_1 |\mu| g(x_1) \epsilon_1 + \sum_{i=2}^{M} a_i \epsilon_i (\mu \circ g)(x_i) \right].
\end{aligned}
$$

In step (24), we can drop the absolute value since we can always make $(g(x_1) - g'(x_1))$ non-negative by exchanging g and g' while leaving the rest of the equation invariant. Proceeding by the above argument inductively on i, we eventually have

$$
\mathrm{Rad}_{S,a}(\mu \circ \mathcal{G}) \leq \mathbb{E}_{\epsilon_i} \sup_{g \in \mathcal{G}} \sum_{i=1}^{M} a_i |\mu| g(x_i) \epsilon_i = |\mu| \mathrm{Rad}_{S,a}(\mathcal{G})
$$

as desired. □

To simplify the notations, we follow Zhang et al. [54] and define the real-valued MLP inductively:

$$
\mathcal{N}_1 = \left\{ x \to \langle x, v \rangle : v \in \mathbb{R}^d, \|v\|_2 \leq \beta \right\},
$$

$$
\mathcal{N}_m = \left\{ x \to \sum_{j=1}^{d} w_j \mu(f_j(x)) : v \in \mathbb{R}^d, \|v\|_1 \leq \beta, f_j \in \mathcal{N}_{m-1} \right\},
$$

where $\beta \in \mathbb{R}_+$ and μ is a 1-Lipschitz activation function. Define an MLP which outputs a vector in $\mathbb{R}^C$ by $\mathcal{G} = \underbrace{\mathcal{N}_m \times \mathcal{N}_m \times \cdots \times \mathcal{N}_m}_{\text{the Cartesian product of } C\ \mathcal{N}_m\text{'s}}$. To leverage standard techniques for the proof, we additionally assume $\forall m \in \mathbb{N}, 0 \in \mu \circ \mathcal{N}_m$.

Proposition 31. *Let* $\mathcal{G} = \underbrace{\mathcal{N}_m \times \mathcal{N}_m \times \cdots \times \mathcal{N}_m}_{\text{the Cartesian product of } C\ \mathcal{N}_m\text{'s}}$. *Assume* $\forall x \in \mathcal{X}, \|x_i\| \leq \alpha$ *and* $\forall m \in \mathbb{N}, 0 \in \mu \circ \mathcal{N}_m$. *Then,*

$$\mathbb{E}_{S,\epsilon_{i,c,j,c'}}\left[\sup_{g\in\mathcal{G}} \sum_{i=1}^{N} w_i \left|\lambda_{\ell_{T_i}}\right| \sum_{c=1}^{C} \frac{\alpha_i(c)}{n_{i,c}} \sum_{j=1}^{n_{i,c}} \sum_{c'=1}^{C} \epsilon_{i,c,j,c'} g_{c'}(X_{i,c,j})\right] \leq$$

$$C\alpha 2^{m-1}\beta^m \sqrt{\sum_{i=1}^{N} w_i^2 \left|\lambda_{\ell_{T_i}}\right|^2 \sum_{c=1}^{C} \frac{\alpha_i^2(c)}{n_{i,c}}},$$

where $\epsilon_{i,c,j,c'}, i \in \mathbb{N}_N, c \in \mathcal{Y}, c' \in \mathcal{Y}, j \in \mathbb{N}_{n_{i,c}}$ *are i.i.d. Rademacher random variables. Thus, the generalization error bound becomes:* $\forall \delta \in [0,1]$, *with probability at least* $1-\delta$,

$$\sup_{g\in\mathcal{G}} \left|\hat{\mathcal{R}}_{w,S}(g) - \widetilde{\mathcal{R}}(g)\right|$$

$$\leq \left(\max_i \left(\left|\lambda_{\ell_{T_i}}\right| A + \left|\lambda_{\ell_{T_i}}\right|_0\right) \sqrt{2\log\frac{2}{\delta}} + C\alpha 2^{m-1}\beta^m \max_i \left|\lambda_{\ell_{T_i}}\right|\right) \sqrt{\sum_{i=1}^{N} w_i^2 \sum_{c=1}^{C} \frac{\alpha_i^2(c)}{n_{i,c}}}.$$

Proof of Proposition 31. For the reader's convenience, we restate the result:

Proposition 32. *Let* $\mathcal{G} = \underbrace{\mathcal{N}_m \times \mathcal{N}_m \times \cdots \times \mathcal{N}_m}_{\text{the Cartesian products of } C\ \mathcal{N}_m\text{'s}}$. *Assume* $\forall x \in \mathcal{X}, \|x_i\| \leq \alpha$ *and* $\forall k \in \mathbb{N}, 0 \in \mu \circ \mathcal{N}_k$. *Then,*

$$\mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c\in\mathcal{N}_m} \sum_{i=1}^{M} a_i \sum_{c=1}^{C} \epsilon_{i,c} g_c(x_i)\right] \leq C\alpha 2^{m-1}\beta^m \sqrt{\sum_{i=1}^{M} a_i^2}.$$

where $a_i > 0$, *and* $\epsilon_{i,c}$ *are independent Rademacher random variables.*

Recall that the MLP outputs a vector in $\mathbb{R}^C$. The set of MLPs is $\mathcal{G} = \underbrace{\mathcal{N}_m \times \mathcal{N}_m \times \cdots \times \mathcal{N}_m}_{\text{the Cartesian products of } C\ \mathcal{N}_m\text{'s}}$ where the set $\mathcal{N}_m$ is defined inductively as

$$\mathcal{N}_1 = \left\{x \to \langle x, v\rangle : v \in \mathbb{R}^d, \|v\|_2 \leq \beta\right\} \quad \text{for } m = 1, \text{and}$$

$$\mathcal{N}_m = \left\{x \to \sum_{j=1}^{d} w_j \mu(f_j(x)) : v \in \mathbb{R}^d, \|v\|_1 \leq \beta, f_j \in \mathcal{N}_{m-1}\right\} \quad \text{for } m > 1.$$

$\beta \in \mathbb{R}_+$, and μ is a 1-Lipschitz activation function. We now proceed with the proof of Proposition 32.

Proof. We have

$$\begin{aligned}
&\mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c\in\mathcal{N}_m} \sum_{i=1}^{M} a_i \sum_{c=1}^{C} \epsilon_{i,c} g_c(x_i)\right] \\
&= \mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c\in\mathcal{N}_m} \sum_{c=1}^{C} \sum_{i=1}^{M} a_i \epsilon_{i,c} g_c(x_i)\right] \\
&\leq \sum_{c=1}^{C} \mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c\in\mathcal{N}_m} \sum_{i=1}^{M} a_i \epsilon_{i,c} g_c(x_i)\right] \\
&= C\mathbb{E}_{\epsilon_i}\left[\sup_{g\in\mathcal{N}_m} \sum_{i=1}^{M} a_i \epsilon_i g(x_i)\right] \\
&= C\text{Rad}_{S,a}(\mathcal{N}_m)
\end{aligned}$$

where

$$\text{Rad}_{S,a}(\mathcal{N}_m) = \mathbb{E}_{\epsilon_i}\left[\sup_{h_j \in \mathcal{N}_{m-1}, \|v\|_1 \le \beta} \sum_{i=1}^{M} a_i \epsilon_i \left(\sum_{j=1}^{d} v_j(\mu \circ h_j)\right)(x_i)\right].$$

Note $\sum_{j=1}^{d} v_j(\mu \circ h_j) \in \beta \operatorname{conv}(\mu \circ \mathcal{N}_{m-1} - \mu \circ \mathcal{N}_{m-1})$. Here the difference between two sets of functions is $\mathcal{G}_1 - \mathcal{G}_2 = \{g_1 - g_2 : g_1 \in \mathcal{G}_1, g_2 \in \mathcal{G}_2\}$ and $\beta\mathcal{G}_1 = \{\beta g_1 : g_1 \in \mathcal{G}_1\}$ for a real number β. Apply Proposition 30,

$$\begin{aligned}
&\text{Rad}_{S,a}(\mathcal{N}_m) \\
&\le \text{Rad}_{S,a}(\beta \operatorname{conv}(\mu \circ \mathcal{N}_{m-1} - \mu \circ \mathcal{N}_{m-1})) \\
&= \beta \text{Rad}_{S,a}(\operatorname{conv}(\mu \circ \mathcal{N}_{m-1} - \mu \circ \mathcal{N}_{m-1})) \\
&= \beta \text{Rad}_{S,a}((\mu \circ \mathcal{N}_{m-1} - \mu \circ \mathcal{N}_{m-1})) \\
&= \beta(\text{Rad}_{S,a}(\mu \circ \mathcal{N}_{m-1}) + \text{Rad}_{S,a}(-\mu \circ \mathcal{N}_{m-1})) \\
&= 2\beta \text{Rad}_{S,a}(\mu \circ \mathcal{N}_{m-1}) \\
&\le 2|\mu|\beta \text{Rad}_{S,a}(\mathcal{N}_{m-1})
\end{aligned}$$

Proceeding backward inductively on m, we have $\text{Rad}_{S,a}(\mathcal{N}_m) \le 2^{m-1}\beta^{m-1}\text{Rad}_{S,a}(\mathcal{N}_1)$. The set $\mathcal{N}_1$ can be viewed as the ball with radius β centered at 0 in the RKHS associated to linear kernel bounded α, so we can apply Proposition 28. Therefore,

$$\text{Rad}_{S,a}(\mathcal{N}_m) \le 2^{m-1}\beta^{m-1}\text{Rad}_{S,a}(\mathcal{N}_1) \le 2^{m-1}\beta^m \alpha \sqrt{\sum_{i=1}^{M} a_i^2}$$

and

$$\mathbb{E}_{\epsilon_{i,c}}\left[\sup_{g_c \in \mathcal{N}_m} \sum_{i=1}^{M} a_i \sum_{c=1}^{C} \epsilon_{i,c} g_c(x_i)\right] \le C\text{Rad}_{S,a}(\mathcal{N}_m) \le C\alpha 2^{m-1}\beta^m \sqrt{\sum_{i=1}^{M} a_i^2}$$

as desired. □

Proof of Theorem 11. Theorem 11 follows Lemma 26, Proposition 27, Proposition 31, and the fact that $\alpha_i(c) \le 1$. □

D.3 Proof of Proposition 12

Proof of Proposition 12. By Corollary 1.42 of Weaver [48], $\|\|\nabla_s \lambda_{\ell_T}(s,y)\|_2\|_\infty$ is a Lipschitz constant of $\lambda_{\ell_T}(\cdot, y)$, where $y \in \{1, 2 \ldots, C\}$, ∇ denotes the gradient of a function, $\|\nabla_s \lambda_{\ell_T}(s,y)\|_2$ is a function maps s to a real number, and the $\|\cdot\|_\infty$ takes the essential supremum over Δ^C. We use $t_{i,j}$ to denote the element at i-row and j-column of T.

$$\lambda_{\ell_T}(s,y) = -\log\left(\sum_{k=1}^{C} t_{y,k} \frac{e^{s_k}}{\sum_{j=1}^{C} e^{s_j}}\right) = -\log\left(\sum_{k=1}^{C} t_{y,k} e^{s_k}\right) + \log\left(\sum_{j=1}^{C} e^{s_j}\right).$$

$$\begin{aligned}
\frac{\partial \lambda_{\ell_T}(s,y)}{\partial s_i} &= -\frac{t_{y,i} e^{s_i}}{\sum_{j=1}^{C} t_{y,j} e^{s_j}} + \frac{e^{s_i}}{\sum_{j=1}^{C} e^{s_j}} = -\frac{t_{y,i} \frac{e^{s_i}}{\sum_{k=1}^{C} e^{s_k}}}{\sum_{j=1}^{C} t_{y,j} \frac{e^{s_j}}{\sum_{j=k}^{C} e^{s_k}}} + \frac{e^{s_i}}{\sum_{j=1}^{C} e^{s_j}} \\
&= -\frac{t_{y,i} p_i}{\sum_{j=1}^{C} t_{y,j} p_j} + p_i
\end{aligned}$$

In the last equality, we denote $\frac{e^{s_i}}{\sum_{k=1}^{C} e^{s_k}}$ by p_i. Then,

$$\|\nabla_s \lambda_{\ell_T}(s,y)\|_2^2 = \sum_{i=1}^{C}\left(-\frac{t_{y,i}p_i}{\sum_{j=1}^{C} t_{y,j}p_j} + p_i\right)^2 \leq \sum_{i=1}^{C}\left|-\frac{t_{y,i}p_i}{\sum_{j=1}^{C} t_{y,j}p_j} + p_i\right| \tag{25}$$
$$\leq \sum_{i=1}^{C}\left(\frac{t_{y,i}p_i}{\sum_{j=1}^{C} t_{y,j}p_j} + p_i\right) = 2$$

The inequality in step (25) follows the observation that $\left|-\frac{t_{y,i}p_i}{\sum_{j=1}^{C} t_{y,j}p_j} + p_i\right| \leq 1$ □

E Confirmation of Probabilistic Model

In Section 5.2, we state that $\alpha(i) = \bar{P}_T(\tilde{Y} = i)$, $P_{\gamma_i}(\cdot) = \bar{P}_T(\cdot \mid \tilde{Y} = i)$, and $\gamma_i(c) = \bar{P}_T(Y = c \mid \tilde{Y} = i)$ for matrix T with $T(i,j) = \frac{\gamma_i(j)\alpha(i)}{\sigma(j)}$. Here we confirm these facts.

Let T be a stochastic matrix with entries $T(i,j) = \frac{\gamma_i(j)\alpha(i)}{\sigma(j)}$. We construct the joint probability measure $\bar{P}_T$ on $\mathcal{X} \times \mathcal{Y} \times \mathcal{Y}$ as described in Section 2. We can see $\bar{P}_T\Big(\tilde{Y} = i\Big) = \sum_{j=1}^{C} \bar{P}_T\Big(\tilde{Y} = i, Y = j\Big) = \sum_{j=1}^{C} \bar{P}_T(Y = j)T(i,j) = \sum_{j=1}^{C} \sigma(j)\frac{\gamma_i(j)\alpha(i)}{\sigma(j)} = \alpha(i)$ and $\forall$ events $\mathcal{A} \subset \mathcal{X}, \forall i, y \in \mathcal{Y}$ $\bar{P}_T\Big(\tilde{Y} = i\Big) = \alpha_i$ and $\forall \mathcal{A} \in \mathcal{M}_{\mathcal{X}}, \forall i, y \in \mathcal{Y}$

$$\begin{aligned}
&\bar{P}_T\Big(X \in \mathcal{A}, Y = y \mid \tilde{Y} = i\Big) \\
&= \frac{1}{\alpha(i)}\bar{P}_T\Big(X \in \mathcal{A}, Y = y, \tilde{Y} = i\Big) \\
&= \frac{1}{\alpha(i)}P(X \in \mathcal{A}, Y = y)\frac{\gamma_i(y)\alpha(i)}{\sigma(y)} \\
&= P_y(X \in \mathcal{A})\gamma_i(y) \\
&= P_{\gamma_i}(X \in \mathcal{A}, Y = y).
\end{aligned}$$

Hence, $\bar{P}_T\Big(\cdot \mid \tilde{Y} = i\Big) = P_{\gamma_i}(\cdot)$, which implies that $\bar{P}_T\Big(Y = c \mid \tilde{Y} = i\Big) = P_{\gamma_i}(Y = c) = \gamma_i(c)$, and for a data point $\Big(X, Y, \tilde{Y}\Big) \sim \bar{P}_T$ the event $\tilde{Y} = i$ entails that $(X, Y) \sim P_{\gamma_i}$.