Adversarial Multiclass Learning under Weak Supervision with Performance Guarantees

Alessio Mazzetto [*1] **Cyrus Cousins** [*1] **Dylan Sam** [1] **Stephen H. Bach** [1] **Eli Upfal** [1]

Abstract

We develop a rigorous approach for using a set of arbitrarily correlated weak supervision sources in order to solve a multiclass classification task when only a very small set of labeled data is available. Our learning algorithm provably converges to a model that has minimum empirical risk with respect to an adversarial choice over feasible labelings for a set of unlabeled data, where the feasibility of a labeling is computed through constraints defined by rigorously estimated statistics of the weak supervision sources. We show theoretical guarantees for this approach that depend on the information provided by the weak supervision sources. Notably, this method does not require the weak supervision sources to have the same labeling space as the multiclass classification task. We demonstrate the effectiveness of our approach with experiments on various image classification tasks.

1. Introduction

In the last decade, deep neural networks have been applied to accurately solve a wide range of classification tasks in different domains, but the supervised learning of these models requires a considerable amount of labeled data. An alternative strategy is to learn from *weak supervision*, i.e., sources of labels that are *noisy* or *heuristic*. Examples include handwritten rules (Ratner et al., 2017; Wu et al., 2018; Safranchik et al., 2020) and classifiers trained for related tasks (Varma et al., 2017; Bach et al., 2019; Chen et al., 2019). Even if these sources of information are noisy, results show that they can lead to high-quality models, particularly when the outputs from many weak sources are combined.

A key technical challenge in such work is how to combine multiple sources of weak supervision, since they might conflict with one another. We assume access to only a small amount of ground-truth labeled data. Much prior work on aggregating noisy labels (Dawid & Skene, 1979; Zhang et al., 2016; Gao & Zhou, 2013; Karger et al., 2014; Ghosh et al., 2011; Dalvi et al., 2013; Ratner et al., 2016; 2019) assumes that the sources make *independent errors*, which is a very strong assumption. Some recent work (Bach et al., 2017; Varma et al., 2019) attempts to learn more sophisticated distributions, but still relies on *parametric assumptions* that make conditional independence assumptions. Such independence assumptions in models of weak supervision sources are hard to verify and limiting in practice. Furthermore, many useful weak supervision sources, particularly ones learned from related datasets, can be arbitrarily correlated, as there may be systematic differences between the target classification task and the mildly related tasks used to learn them. For example, if all the labelers are fine-tuned from the same pretrained model, they are likely to inherit some of the same biases.

Recent work has addressed the problem of combining weak labelers without distributional assumptions by taking an *adversarial approach*. For binary classification, Balsubramani & Freund (2015) formulate the problem as minimax optimization, where the goal is to find the labels of an unlabeled dataset that minimize the error with respect to the worst-case assignment to the unknown ground-truth labels, while satisfying statistical constraints on the individual error of the weak labelers. This minimax problem can be optimally solved for a large family of loss functions (Balsubramani & Freund, 2016). The *adversarial label learning* (ALL) framework (Arachie & Huang, 2019) uses a similar minimax optimization to learn a model that minimizes risk using the worst-case assignment to the unknown ground-truth labels, and was later extended to the multiclass setting (Arachie & Huang, 2021), but it does not optimally solve the minimax optimization problem, and provides no generalization guarantees for the models it learns.

Another recent work, *performance guaranteed majority vote* (PGMV) (Mazzetto et al., 2021), takes an alternative approach for the binary hard classification setting. Instead of working with an adversarial choice of the ground-truth labels, it uses both a small amount of labeled data and a large amount of unlabeled data to empirically estimate properties

[*]Equal contribution, authorship order decided by coin flip.
[1]Department of Computer Science, Brown University. Correspondence to: Alessio Mazzetto <alessio_mazzetto@brown.edu>.

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

of the labelers which are then used to constraint their joint output distribution. However, this approach is inherently limited to hard binary classification, as it exploits the fact that when two labelers disagree, one must be correct.

In this paper, we address the limitations of previous work by providing a framework for multiclass classification with weak supervision, with rigorous computational efficiency and generalization error guarantees. Similar to ALL, we formulate the search for ground truth as a search over the set of feasible labelings that satisfy statistical constraints on the weak supervision sources. However, ALL lacks theoretical guarantees, and we show using techniques from convex optimization that our training algorithm rapidly converges to the optimal solution of the minimax optimization problem. Furthermore, we provide generalization bounds through uniform convergence theory for the learned model, in terms of the information provided by the weak supervision sources (with respect to the target classification), geometrically represented as the diameter of the set of feasible labelings.

Contributions. We introduce a novel method to use the information provided by a set of arbitrarily correlated weak supervision sources to learn a classifier for a given target task. Inspired by previous work, we use a small amount of labeled data to compute statistics of the weak supervision sources, and we formulate an optimization problem to find the prediction model that achieves the lowest empirical risk with respect to an adversarial choice of a labeling of an unlabeled dataset that agrees with those statistics. Our main contributions are as follows.

1. We develop the first method with theoretical guarantees for learning multiclass classifiers from weak supervision sources without any prior assumptions on the joint distribution of their outputs and the true label (§4).

2. We provide theoretical analysis of our method, proving approximation guarantees on the quality of our solution, and time complexity bounds for the training algorithm (§4).

3. We provide generalization bounds for the solution provided by our method using a geometrical quantity that represents the aggregate information provided by the weak supervision sources with respect to the target classification task (§4.2).

4. While the presentation of our method is general, we demonstrate the applicability of our approach through two practical instances of prediction model and loss function: convex combination of the weak supervision sources and multinomial logistic regression (§4.1).

5. We show how to extend our method to use weak supervision sources with different labeling spaces from the target task. This is useful, e.g., when learning with attributes. In many weak supervision tasks, related classifications, such as whether a classifier detects *stripes* on an animal, yields partial information for target tasks like *species* identification (§4.3).

6. We conduct experiments demonstrating the effectiveness of our novel approach for multiclass classification tasks. Our experiments show that our method compares favorably with the recently-published ALL and PGMV algorithms for (binary classification) from weak supervision sources (§5).

2. Related Work

The problem of learning from multiple, possibly conflicting, weak labelers with little to no ground-truth data has received considerable attention recently (Ratner et al., 2016; Bach et al., 2017; Ratner et al., 2017; Varma et al., 2019; Arachie & Huang, 2019; Mazzetto et al., 2021).This setting is distinct from much work on ensemble learning (Zhang & Ma, 2012), such as boosting (Schapire, 1990; Freund, 1995), where abundant labeled examples are used to learn to combine ensemble members. Other ensemble methods, such as bagging (Breiman, 1996), take an unweighted vote of ensemble members, but rely on the assumption that each member is trained on labeled data sampled from the target distribution. Unlike these methods, in weak supervision, the goal is to use other statistical properties of the labelers, such as their agreements and disagreements, to learn to combine them. In this way, the combination of the labelers can be potentially improved without increasing the need for labeled training data.

This work has its roots in crowdsourcing, where the “labelers” are people with varying unknown levels of reliability. Dawid and Skene's (1979) seminal work showed how the accuracy of each labeler can be estimated with expectation maximization by assuming a naive Bayes distribution over the labelers' votes and the latent ground truth. Since then, much work has provided theoretically guaranteed algorithms for learning under these assumptions (Zhang et al., 2016; Gao & Zhou, 2013; Karger et al., 2014; Ghosh et al., 2011; Dalvi et al., 2013). When the labelers are humans working without coordination, the independence assumption is a reasonable one.

Recently, frameworks for weakly supervised machine learning like Snorkel (Ratner et al., 2016; Bach et al., 2017; Ratner et al., 2017) have used and extended these learning techniques to the setting in which the labelers are programmed rules, weak classifiers, or other heuristics. As described in the introduction, learned and programmed labelers can have heavily correlated errors because of common elements in the heuristics they use. This potential problem has motivated attempts to relax the independence assumption. One line of work (Bach et al., 2017; Varma et al., 2019) has tried to learn more sophisticated parametric models of the labelers, but they are still limited by how correct their assumptions are, which are hard to verify in practice. In this work, we therefore focus on methods for learning from weak supervision that do not make such assumptions on the distribution of labeler outputs and ground truth.

3. Preliminaries

We denote scalar and generic items as lowercase letters, vectors as lowercase bold letters, and matrices as bold uppercase letters. The i-th column of a matrix $\mathbf{A}$ is denoted by the corresponding lowercase symbol $\mathbf{a}_i$, i.e., $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_n]$. Due to space constraints, all proofs are deferred to the appendix.

In multiclass learning, we have a domain $\mathcal{X}$ and a classifier function h that maps each $x \in \mathcal{X}$ to one of k possible labels (classes). Since we will work later with distributions over the k classes, it is convenient to represent label $i \in 1, \ldots, k$, as a k-dimensional vector $\mathbf{e}_i$, with all components set to 0, except for the i-th component, which is set to 1. Thus, $h : \mathcal{X} \to \mathcal{Y} = \{\mathbf{e}_1, \ldots, \mathbf{e}_k\}$. A classifier (e.g., the softmax layer of a neural network) may output a probability distribution vector $\mathbf{y} \in \mathbb{R}^k_{\geq 0}$ over the k classes, where y_c is the probability that the item belongs to class c, and $\sum_c y_c = 1$. We take $\tilde{\mathcal{Y}} \supset \mathcal{Y}$ to be the set of all possible probability vectors. A loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ quantifies the error of the classifier's output $h(x)$ with respect to the true label $\mathbf{y}$. Let p_{XY} be the probability distribution over $\mathcal{X} \times \mathcal{Y}$. Given a classifier h, its risk is defined as

$$R(h) \doteq \mathop{\mathbb{E}}_{(x,\mathbf{y}) \sim p_{XY}} \left[\ell(h(x), \mathbf{y}) \right] .$$

In standard supervised learning, we are given *labeled samples* from p_{XY}, and we find a classifier with low risk among a set of classifiers $\mathcal{H}$, which is also called a *hypothesis class*. The amount of labeled data required to guarantee that we can find (or train) such a classifier is referred to as the *sample complexity*, which is related to the *size* or *expressivity* of $\mathcal{H}$. For many classification tasks of interest, there could be low availability of labeled data, and this is a critical problem for a wide range of domains, where the most successful hypothesis classes are very expressive (e.g., convolutional neural networks for images).

In this work, we assume access to m_L i.i.d. labeled samples $\tilde{X} = \{\tilde{x}_1, \ldots, \tilde{x}_{m_L}\}$, $\tilde{\mathbf{Y}} = [\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_{m_L}]$ drawn from p_{XY}, where the sample size m_L is insufficient for the direct supervised learning of $\mathcal{H}$. To circumvent the lack of sufficient training data, we assume access to a set of weak labelers (classifiers) $\lambda_1, \ldots, \lambda_n$, also called weak supervision sources. These labelers are *weak* in the sense that they can be inaccurate with respect to the target classification task. For example, the weak labelers could be trained for classification tasks that are only tangentially related to the target classification task: we could train a labeler to detect stripes on *zebras* and *horses*, and then attempt to use it to label images as either *tigers* or *lions*. Moreover, we add no further assumptions on the properties of those classifiers, and their output could be arbitrarily correlated. We also assume access to m unlabeled data points $X = \{x_1, \ldots, x_m\}$ sampled independently from the marginal distribution p_X, and our method uses the weak supervision sources $\lambda_1, \ldots, \lambda_n$ to constrain the space of possible labels that can be given to these unlabeled data points. We use the limited labeled data to compute *statistics* of the weak labelers, and then consider possible labelings of the unlabeled data X that satisfy feasibility constraints derived from these statistics.

As an example, suppose that we use the m_L labeled data points to compute the *empirical risk* statistic of each weak supervision source, i.e., $\hat{\tau}_i = \frac{1}{m_L} \sum_{j=1}^{m_L} \ell(\lambda_i(\tilde{x}_j), \tilde{\mathbf{y}}_j)$, for each $i \in 1, \ldots, n$. In Section 4, we use related statistics in order to prove generalization guarantees. If we were to assign a labeling to the unlabeled data points X, a reasonable approach would be to find a labeling such that the empirical risk of the weak supervision source i computed with respect of those labels is equal to $\hat{\tau}_i$. However, this is a computationally hard problem, as we have to assign a discrete label (from $\mathcal{Y}$) to each item, and each label affects the empirical risk of all the weak supervision sources. Moreover, there is no guarantee that we can find such a labeling for the unlabeled data, and it is unclear which labeling to choose in case there are multiple solutions.

To address the computational issues with discrete label selection, we assign a probability vector from $\tilde{\mathcal{Y}}$ to each unlabeled data point. In other words, for each unlabeled item x_j, we assign a probability vector $\mathbf{y}_j$, where $y_{j,c}$ represents the probability that item x_j belongs to class c. Given a classifier h, we define the loss of the classifier on item $x \in \mathcal{X}$ with respect to the probability vector $\mathbf{y} \in \tilde{\mathcal{Y}}$ as the expected loss. Abusing notation, let $\mathbf{e} \sim \mathbf{y}$ denote that $\mathbf{e} = \mathbf{e}_c \in \mathcal{Y}$ with probability y_c. We then define

$$\tilde{\ell}(h(x), \mathbf{y}) \doteq \mathop{\mathbb{E}}_{\mathbf{e} \sim \mathbf{y}} \left[\ell(h(x), \mathbf{e}) \right] = \sum_{c=1}^{k} y_c \, \ell(h(x), \mathbf{e}_c) . \quad (1)$$

We observe that this definition of loss generalizes the one computed with respect to a discrete labeling, since for each $\mathbf{e} \in \mathcal{Y}$, we have $\ell(h(x), \mathbf{e}) = \tilde{\ell}(h(x), \mathbf{e})$. Also, the loss (1) is linear with respect to the labeling $\mathbf{y}$. Let $\mathbf{Y} \in \mathbb{R}^{k \times m}$ be a matrix that describes a possible labeling of the unlabeled data points; in particular the j-th column of the matrix $\mathbf{Y}$ is $\mathbf{y}_j \in \tilde{\mathcal{Y}}$, and it denotes the probability vector of the labeling of the item x_j. The *empirical risk* of a classifier h on the unlabeled data X with labeling $\mathbf{Y}$ is defined as

$$\hat{R}(h; X, \mathbf{Y}) \doteq \frac{1}{m} \sum_{j=1}^{m} \tilde{\ell}(h(x_j), \mathbf{y}_j) .$$

Finding a labeling $\mathbf{Y}$ for which $\hat{R}(\lambda_i; X, \mathbf{Y}) = \hat{\tau}_i$ for $i \in 1, \ldots, n$ is equivalent to the computationally easy task of solving a linear system with $O(n+m)$ constraints (the n constraints on the empirical risk equality and m constraints on probability vectors summing to 1) and $O(mk)$ variables. However, there still could be multiple solutions to such an underdefined linear system. The core idea of the method presented in Section 4 is to find a model that has the lowest empirical risk with respect to an adversarial choice among a related feasible set of labelings.

4. Learning Algorithm

Let $\mathcal{H} = \{h_\theta : \theta \in \Theta \subseteq \mathbb{R}^d\}$ be the hypothesis class that we will use to find the classifier for the classification task of interest, where each classifier $h_\theta \in \mathcal{H}$ is parametrized by a vector of weights θ.

Let Y^* be the (unknown) true labeling of the unlabeled data X. For each weak supervision source i, we use the labeled data to compute an interval Δ_i such that, with high probability, we have that $\hat{R}(\psi_i(x); X, Y^*) \in \Delta_i$ for $i \in 1, \ldots, n$. This is a crucial property that we will need to show our theoretical bound (Theorem 8), and we construct such intervals in Lemma 1.

Let $\mathcal{Y}$ be the set of all possible labeling matrices Y such that the empirical risk of ψ_i, computed with respect to the labeling Y of the unlabeled data X, belongs to the corresponding interval Δ_i for each weak supervision source. Formally, the set $\mathcal{Y}$ is defined as

$$\mathcal{Y} \doteq \{ Y \in \mathbb{R}^{k \times m} : \begin{array}{lll} y_j & \in \Delta_k & \text{for } j \in 1, \ldots, m \\ \hat{R}(\psi_i; X, Y) & \in \Delta_i & \text{for } i \in 1, \ldots, n \end{array} \} .$$

We will refer to $\mathcal{Y}$ as the set of *feasible labelings*. The next lemma shows how to build the intervals Δ_i to guarantee that, with high probability, the true labeling Y^* is feasible.

Lemma 1 (Weak Labeler Risk Constraints)**.** Suppose that the codomain of the loss function ℓ is contained in the interval $[0, B]$. Let $\hat{\epsilon}_1, \ldots, \hat{\epsilon}_n$ be the empirical risks of $\psi_1, \ldots, \psi_n$ computed with respect to the m_L labeled samples. Fix a value $\delta \in (0,1)$ and take

$$\epsilon \doteq B \sqrt{\frac{(m_L + m) \ln \frac{2n}{\delta}}{2 m_L m}} .$$

If we set $\Delta_i = [\hat{\epsilon}_i - \epsilon, \hat{\epsilon}_i + \epsilon]$, then with probability at least $1 - \delta$ it holds that $Y^* \in \mathcal{Y}$.

We want to find the classifier that achieves the lowest empirical risk among the feasible labelings of the unlabeled datapoints. That is, we choose the classifier $h_{\hat{\theta}} \in \mathcal{H}$, where $\hat{\theta}$ is the solution of the minimax problem

$$\hat{\theta} \doteq \arg\min_{\theta \in \Theta} \max_{Y \in \mathcal{Y}} \hat{R}(h_\theta; X, Y) . \quad (2)$$

The optimization problem above has some nice properties. The set $\mathcal{Y}$ is specified by linear constraints in Y. Moreover, the objective of the minimax (2) problem is also linear in Y. Hence, it is easy to see that for a given $\theta \in \Theta$, it is possible to solve the maximization problem

$$f(\theta) = \max_{Y \in \mathcal{Y}} \hat{R}(h_\theta; X, Y) , \quad (3)$$

through a linear program with $O(mk)$ variables and $O(m+n)$ constraints.

In order to solve the minimax problem (2), we will introduce a few assumptions on the loss function and the model choice $\mathcal{H}$, which are satisfied by many classic machine learning settings. In particular, we would like the function $f(\theta)$ to be convex, so that we can solve the minimization problem $\min_{\theta \in \Theta} f(\theta)$. Even if $f(\theta)$ is convex, we may not be able to apply a gradient-based optimization method, as (3) involves a *maximization*, hence it is not differentiable everywhere. To solve this issue, we use the *subgradient*, which generalizes the gradient. This will require the loss function to be Lipschitz continuous. A function $g : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ is said to be L-Lipschitz continuous if for any $x, y \in \mathbb{R}^{d_1}$, it holds that $||g(x) - g(y)||_2 \le L ||x - y||_2$.

Definition 2 (Subgradient)**.** Let $A \subseteq \mathbb{R}^b$ be the domain of a function g. A vector $v \in \mathbb{R}^b$ is a subgradient for a function g at $x \in A$ if for any $y \in A$ we have that

$$g(y) - g(x) \ge v^T (y - x) .$$

For each $x \in A$, we define

$$\partial g(x) \doteq \{ v : v \text{ is a subgradient of } g \text{ at } x \} .$$

If a function is *differentiable* at a point, then its subgradient with respect to that point is unique, and equals the gradient. Furthermore, if the function is *convex*, then there exists at least one subgradient for each point of its domain.

The following intermediate result, which immediately follows from the definition of ℓ, will prove useful throughout this discussion.

Lemma 3 (Linear Loss Properties)**.** Let $\ell(h_\theta(x); e)$ be convex and L-Lipschitz continuous with respect to θ for any $(x, e) \in \mathcal{X} \times \mathcal{Y}$. Then, for any probability vector $y \in \mathcal{Y}$, the function $\ell(h_\theta(x); y)$ is also convex and L-Lipschitz continuous with respect to θ.

The next Lemma shows that under some conditions often encountered in our adversarial learning framework, it is possible to compute the subgradient of the function f.

Lemma 4 (Subgradient of Adversarial Learning)**.** Fix a value $\theta^0 \in \text{interior}(\Theta)$, let $Y^0 \doteq \arg\max_{Y \in \mathcal{Y}} \hat{R}(h_{\theta^0}; X, Y)$, and assume that $\ell(h_\theta(x); e)$ is convex with respect to θ for any $x \in \mathcal{X}$ and $e \in \mathcal{Y}$. Then

$$\emptyset \ne \partial \hat{R}(h_{\theta^0}; X, Y^0) \subseteq \partial f(\theta^0) .$$

Algorithm 1 Subgradient Algorithm

Input: Number of iterations T, step size h, $\mathcal{H}$, X, $\tau_1, \ldots, \tau_n$
Output: Approximate solution $\tilde{\theta}$ of (2) (See Theorem 5)
$\tilde{\theta}^{(0)} = \theta^{(0)} \leftarrow$ arbitrary point $\in \Theta$
for $t \in 1, \ldots, T$ **do**
  $Y' \leftarrow \arg\max_{Y \in \mathcal{Y}} \hat{R}(h_{\theta^{(t-1)}}, X, Y)$
  $v \leftarrow$ arbitrary vector from $\partial \hat{R}(h_{\theta^{(t-1)}}, X, Y')$
  $\theta^{(t)} \leftarrow P_\Theta(\theta^{(t-1)} - hv)$ (P_Θ is projection onto Θ)
  $\tilde{\theta}^{(t)} \leftarrow \arg\min\{f(\tilde{\theta}^{(t-1)}), f(\theta^{(t)})\}$
end for
Return $\tilde{\theta}^{(T)}$

A subgradient-based optimization approach (Shor et al., 1985) is similar to gradient descent, however at each iteration we use the subgradient instead of the gradient, and we memorize the best solution found among all the iterations.

The subgradient-based optimization algorithm used to solve the optimization problem (2) is presented in Algorithm 1.

As observed before, Y' as defined in the algorithm can be computed by solving a linear program. The projection step depends on the set of parameters Θ. While this is not a requirement for our approach, if the loss function $\ell(h_\theta(x), y)$; is differentiable with respect to θ, then we can compute the gradient of the empirical risk instead of a subgradient.

Theorem 5 (Subgradient Method Convergence Rates)**.** Suppose that for any $(x, y) \in X \times \mathcal{Y}$, $\ell(h_\theta(x), y)$ is L-Lipschitz continuous and convex with respect to θ. Let step size $h > 0$, and iteration count $T \in \mathbb{N}$, and $\tilde{\theta}$ as returned by Algorithm 1. Then, we have that

$$f(\tilde{\theta}) - f(\hat{\theta}) \leq \frac{\operatorname{diameter}(\Theta)^2 + L^2h^2T}{2hT} ;$$

where $\operatorname{diameter}(\Theta)$ is computed with respect to the ℓ_2-norm, i.e., $\operatorname{diameter}(\Theta)^2 \doteq \max_{\theta_1, \theta_2 \in \Theta} \|\theta_1 - \theta_2\|_2^2$, and $\hat{\theta}$ is defined as in (2). Alternatively, for any $\varepsilon > 0$, then if $h = \frac{\varepsilon}{L^2}$ and $T \geq \frac{L^2 \operatorname{diameter}(\Theta)^2}{\varepsilon^2}$, we have that

$$f(\tilde{\theta}) - f(\hat{\theta}) \leq \varepsilon :$$

Therefore, we can compute a solution within additive error ε of (2) by running $O(\frac{L^2 \operatorname{diameter}(\Theta)^2}{\varepsilon^2})$ iterations of the subgradient algorithm.

4.1. Applications

In order to feature the generality of our framework, we show two examples of different instantiations of the optimization problem (2) for different choices of loss function and prediction models for which we can apply Theorem 5.

Convex combination of the weak supervision sources Let $\Theta = \{\theta = (\theta_1, \ldots, \theta_n) \in \mathbb{R}^n_+ : \sum_{i=1}^n \theta_i = 1\}$. Our prediction model is a convex combination of the output of the weak classifiers $\lambda_1, \ldots, \lambda_n$. In particular, given $\theta \in \Theta$, the classifier h_θ is defined as $h_\theta(x) = \sum_{i=1}^n \theta_i \lambda_i(x)$ for any $x \in X$. It is easy to see that $\operatorname{diameter}(\Theta) \leq \sqrt{2}$. Given an arbitrary vector $v \in \mathbb{R}^n$, the projection step to Θ can be done efficiently by using for example the algorithm of Wang & Carreira-Perpiñán (2013).

Let ℓ be the Brier loss, defined for any $(x, e) \in X \times \mathcal{Y}$ as

$$\ell(h_\theta(x), e) \doteq \sum_{c=1}^{k} (h_\theta(x)_c - e_c)^2$$
$$= \|h_\theta(x)\|_2^2 - 2h_\theta(x)^T e + 1 :$$

It is easy to see that the function $\ell(h_\theta(x), e)$ is convex, differentiable with respect to θ, and has codomain $[0, 2]$.

Lemma 6 (Brier Model Lipschitz Properties)**.** The loss $\ell(h_\theta(x), e)$ of a prediction model h_θ defined as in this subsection is $2\sqrt{n}$-Lipschitz continuous with respect to θ.

Softmax (multinomial logistic legression) Suppose that each item is a vector in $\mathbb{R}^b$, i.e., $X \subseteq \mathbb{R}^b$, and assume that $\|x\|_2 \leq B_X$ for any $x \in X$. Let $\Theta = \{\theta = (w_1^T \ldots w_k^T) \in \mathbb{R}^{bk} : w_c \in \mathbb{R}^b \wedge \|w_c\|_2 \leq B_W$ for $c \in \{1, \ldots, k\}\}$. That is, θ is the concatenation of k vectors with bounded norm. Observe that with this definition of Θ, we have that $\operatorname{diameter}(\Theta) \leq \sqrt{2k}B_W$. Given a vector $\theta = (w_1^T \ldots w_k^T)$, the projection step to Θ is simply $\tilde{\theta} = (\tilde{w}_1^T \ldots \tilde{w}_k^T)$, where $\tilde{w}_c = w_c/\min(B_W/\|w_c\|_2, 1)$ for $c \in \{1, \ldots, k\}$.

Given $\theta = (w_1^T \ldots w_k^T) \in \Theta$ and $x \in X$, we define

$$h_\theta(x) \doteq \left(\frac{\exp(w_1^T x)}{\sum_{c=1}^k \exp(w_c^T x)}, \ldots, \frac{\exp(w_k^T x)}{\sum_{c=1}^k \exp(w_c^T x)} \right)^T :$$

This classifier is a particular instantiation of softmax combined with a linear model. For a vector $v = (v_1, \ldots, v_d)^T$, define $\ln v \doteq (\ln v_1, \ldots, \ln v_d)^T$. Given $(x, e) \in X \times \mathcal{Y}$, we define the cross-entropy loss ℓ of the prediction model h_θ as

$$\ell(h_\theta(x), e) \doteq -e^T \ln(h_\theta(x)) :$$

This combination of prediction model and loss function is also known as multinomial logistic regression. It is easy to see that the loss function is differentiable with respect to θ, and it is a known result that $\ell(h_\theta(x), e)$ is convex with respect to θ for any $(x, e) \in X \times \mathcal{Y}$ (Böhning, 1992). We now characterize the boundedness and Lipschitz properties of the softmax function with respect to the cross-entropy loss.

Lemma 7 (Properties of Multinomial Logistic Regression)**.** For any $(x, e) \in X \times Y$, and $\theta \in \Theta$, we have

1. $\ell(h_\theta(x), e) \in [0, B_W B_X + \ln k]$; and
2. $\ell(h_\theta(x), e)$ is (kB_X)-Lipschitz continuous with respect to θ.

4.2. Statistical Learning Guarantees

In this subsection, we develop a bound on the true risk of the classifier $h_{\hat{\theta}}$ that is a solution of the optimization problem (2). The bounds are expressed in function of the Rademacher complexity of the function family $L = \{\ell \circ h : h \in H\}$ that describes the loss of each function $h \in H$, the risk minimizer $\theta^* = \arg\min_{\theta \in \Theta} R(h_\theta)$, and the average diameter $D_{\mathcal{Y}}$ of the feasible set of solutions $\mathcal{Y}$, where

$$D_{\mathcal{Y}} \doteq \sup_{Y', Y'' \in \mathcal{Y}} \frac{1}{m} \sum_{j=1}^{m} \left\| y'_j - y''_j \right\|_1 . \tag{4}$$

The quantity $D_{\mathcal{Y}}$ characterizes the information given by the classifiers $\lambda_1, \ldots, \lambda_n$ on the classification task. In particular, a weak supervision source provides useful information on the classification task of interest only if it reduces the size of the feasible set, and it provably improves the performance of our algorithm if it decreases the average diameter $D_{\mathcal{Y}}$.

Given a function family L, we define the empirical Rademacher average (see Mitzenmacher & Upfal, 2017) of the unlabeled items X and a possible labeling Y of those items as

$$\hat{R}_m(L; X; Y) \doteq E_\sigma \left[\sup_{h \in L} \frac{1}{m} \sum_{i=1}^{m} \sigma_i \ell(h(x_i); y_i) \right] ,$$

where $\sigma_1, \ldots, \sigma_m$ are independent random variables from the Rademacher distribution, i.e., $P(\sigma_i = 1) = P(\sigma_i = -1) = \frac{1}{2}$. Intuitively, this quantity measures the capacity of H to overfit, and under mild conditions, it approaches 0 as sample size m tends to infinity, in which case overfitting becomes impossible.

Theorem 8 (Adversarial Risk Bounds)**.** Let $h_{\hat{\theta}}$ be the solution of (2). Let $\theta^* = \arg\min_{\theta \in \Theta} R(h_\theta)$. Suppose that the codomain of the loss function ℓ is contained in the interval $[0, B]$. Let Y^* be the true (unknown) labeling of the unlabeled data X, and assume that $Y^* \in \mathcal{Y}$. Then, with probability $1 - \delta$ it holds that

$$R(h_{\hat{\theta}}) \le R(h_{\theta^*}) + BD_{\mathcal{Y}} + \sup_{Y \in \mathcal{Y}} 4\hat{R}_m(L; X; Y) + O\left(B \sqrt{\frac{\ln \frac{1}{\delta}}{m}} \right) .$$

4.3. Constraining the Feasible Set

Previously, our presentation has implicitly assumed an alignment between the output classes of the weak supervision sources $\lambda_1, \ldots, \lambda_n$ and the target classification task. In fact, as seen in Lemma 1, we compute the intervals Δ_i based on the empirical risk of the weak supervision sources using labeled data of the target classification task. However, for many applications of interest, the weak supervision sources could output to a different codomain, potentially with an unequal number of classes. As an example, suppose that we would like to distinguish between images of $\{$cat, dog, rabbit, bear$\}$. A binary classifier that tells us if the animal represented in an image has a tail or not still provides a useful clue with respect to the target classification task, and we would like to use that information.

In this subsection, we will show how to constrain the feasible set of labelings $\mathcal{Y}$ in a more general setting, where the weak supervision source λ_i is a classifier that maps elements from the domain X to soft labels over k_i classes, i.e., $\lambda_i : X \to Y_{k_i}$, where $Y_{k_i} = \{v \in \mathbb{R}^{k_i}_{\ge 0} : \sum_c v_c = 1\}$.

Consider the weak supervision source λ_i. For each $c \in 1, \ldots, k$ and $\tilde{e} \in 1, \ldots, k_i$, we use the m_L labeled examples $(\tilde{x}_1; \tilde{y}_1), \ldots, (\tilde{x}_{m_L}; \tilde{y}_{m_L})$ to compute the statistic

$$\hat{\mu}_{i;c;\tilde{e}}(\tilde{X}; \tilde{Y}) \doteq \frac{1}{|\tilde{X}|} \sum_{j=1}^{|\tilde{X}|} y_{j;c} [\lambda_i(x_j)]_{\tilde{e}} .$$

It is clear that the function $\hat{\mu}_{i;c;\tilde{e}}(\tilde{X}; \tilde{Y})$ is linear in $\tilde{Y}$. For each weak supervision source λ_i, true class $c \in 1, \ldots, k$, and weak supervision source's output class $\tilde{e} \in 1, \ldots, k_i$, based on the value $\hat{\mu}_{i;c;\tilde{e}}(\tilde{X}; \tilde{Y})$, we compute an interval $\Delta_{i;c;\tilde{e}}$, defined as

$$\Delta_{i;c;\tilde{e}} \doteq [\hat{\mu}_{i;c;\tilde{e}}(\tilde{X}; \tilde{Y}) - \epsilon; \hat{\mu}_{i;c;\tilde{e}}(\tilde{X}; \tilde{Y}) + \epsilon] ,$$

where the value ϵ is specified in Lemma 9.

Given a labeling Y of the unlabeled dataset X, we say that Y is a feasible solution if for each i, c and $\tilde{e}$, it holds that:

$$\hat{\mu}_{i;c;\tilde{e}}(X; Y) \in \Delta_{i;c;\tilde{e}} . \tag{5}$$

That is, the set of all the feasible solutions $\mathcal{Y}$ is defined as

$$\begin{aligned} \mathcal{Y} \doteq \{ Y \in \mathbb{R}^{k \times m} : \; & y_j \in Y && \text{for } j \in 1, \ldots, m \\ & \hat{\mu}_{i;c;\tilde{e}}(X; Y) \in \Delta_{i;c;\tilde{e}} && \forall i, c, \tilde{e} \} . \end{aligned}$$

Notice that the constraints specified in $\mathcal{Y}$ are still linear in Y, therefore we can still compute the value $f(\theta)$ (as in (3)) by solving a linear program, and all the discussion done with empirical-risk based constraints still applies.

In order to be able to give the theoretical bound of Theorem 8, we need to guarantee that the true labeling Y^* of the unlabeled data X is feasible. This is possible by choosing a suitable value ϵ when defining the intervals $\Delta_{i,c,\tilde{c}}$.

Lemma 9 (Generalized Weak Labeler Constraints). For every $i \in \{1, \ldots, n\}$, $c \in \{1, \ldots, k\}$, and $\tilde{c} \in \{1, \ldots, k_i\}$ let $\Delta_{i,c,\tilde{c}}$ be computed as in (5). Let $K = k \sum_{i=1}^{n} k_i$. Fix a value $\delta \in (0, 1)$, if we use the value

$$\epsilon \doteq \sqrt{\frac{(m_L + m) \ln \frac{2nK}{\delta}}{2 m_L m}}$$

to compute those intervals, then with probability at least $1 - \delta$ it holds that $Y^* \in \mathcal{Y}$.

Sharper bounds for interval estimates, both for *risk constraints* (Lemma 1), and *generalized constraints* (Lemma 9) are of course possible. The Hoeffding bound, used to show both results, is known to be loose for *low-variance functions*, and the union bound is loose for *correlated functions*. Informative weak labelers should produce low-variance statistics, and our framework is designed explicitly for correlated labelers. The costly union bound can be circumvented via the Rademacher average, and Cousins & Riondato (2020) show that finite or linear families of statistics, particularly those with low variance, can be uniformly-bounded, even more sharply with the *empirically centralized Rademacher average*.

5. Experiments

We demonstrate the applicability and performance of our method on image multiclass classification tasks derived from the DomainNet (Peng et al., 2019) dataset. We also provide experiments on image binary classification tasks derived from the Animals with Attributes 2 (Xian et al., 2018) dataset in order to compare our methods with additional baselines. The code for the experiments is available online.[1]

DomainNet contains images from 345 different classes in 6 different domains, which we refer to as $\mathcal{P}$ = {clipart, infograph, painting, quickdraw, real, sketch}. Animals with Attributes 2 contains natural images of 50 types of animals. Associated with the dataset is a list of 85 attributes for each animal class, which we use to create weak supervision sources. Animals with Attributes 2 is divided into 40 "seen" classes and 10 "unseen" classes, where the seen classes can be used to train attribute classifiers without leaking information about the unseen classes.

We refer to our algorithms by using the acronyms AMCL-CC and AMCL-LR, where AMCL stands for *A*dversial *M*ulti *C*lass *L*earning. AMCL-CC is an implementation of our method that uses a *C*onvex *C*ombination of the weak supervision sources as the prediction model, whereas AMCL-LR uses (multinomial) *L*ogistic *R*egression (see Section 4.1). For every image, we compute the output of a pretrained ResNet-18 and use it as input for AMCL-LR.

5.1. Setup

From DomainNet, we select $k = 5$ random classes from the 25 classes with the largest number of instances. Then, for each domain $p \in \mathcal{P}$, we learn a multiclass classifier h_p for those k classes in domain p. The classifier h_p is trained by fine-tuning a pretrained ResNet-18 network (He et al., 2016), using 60% of the labeled data for that domain. For each domain p, we consider the classifiers trained in the remaining domains, i.e., $\mathcal{P} \setminus \{p\}$, as weak supervision sources, i.e., the classifiers $\{h_q\}$ for $q \in \mathcal{P} \setminus \{p\}$. We remark that these weak supervision sources never have access to samples from domain p.

From Animals With Attributes, we create binary classification tasks by selecting pairs of unseen classes. Following Mazzetto et al. (2021), we create weak supervision sources by using the seen classes to train classifiers for the attributes that distinguish them. Similarly to Domain Net, these classifiers are learned by fine-tuning a pretrained ResNet-18 network using labeled data from the seen classes. In order to focus on the most challenging tasks (where the weak supervision sources are not highly accurate), we select 4 the class pairs among the unseen classes with the lowest majority vote accuracy.

We remark that all algorithms that require unlabeled data are evaluated in a transductive setting: the unlabeled data used by the algorithms are also used to evaluate the final learned prediction models.

5.2. Baselines and Algorithms

Following the example of Mazzetto et al. (2021), we compare our method with the following five baselines and algorithms.

Best Weak Supervision Source (Best WSS): We report the accuracy of the best weak supervision source.

Majority Vote (MV) : We consider a simple approach to combining the weak supervision sources: we average their output and select the most voted class. This approach requires no learning, but is suboptimal when the errors made by weak supervision sources are not independent, or when the error rates of weak supervision sources are not equal.

Semi-Supervised Dawid-Skene Estimator (DS): We also consider a semi-supervised extension to the standard crowdsourcing algorithm (Dawid & Skene, 1979) that finds the optimal aggregation of the outputs of independent weak supervision sources. The Dawid-Skene estimator is also the default aggregation method for the Snorkel system (Ratner

[1] https://github.com/BatsResearch/mazzetto-icml21-code

Figure 1. Experimental results on Animals with Attributes for the binary classification tasks of dolphin v. blue whale and seal v. walrus as we vary the amount of labeled data. Each method uses 560 unlabeled data for dolphin v. blue whale and 602 unlabeled data for seal v. walrus.

Figure 2. Experimental results on Domain Net for the clipart and quickdraw domains as we vary the amount of labeled data. Each method uses 500 unlabeled data. Results are listed for the 5 classes of sea turtle, vase, whale, bird, violin.

et al., 2017). Here, we use a semi-supervised version of this algorithm, for a fair comparison with our work. We simply optimize the marginal likelihood of the weak supervision sources' outputs using the unlabeled data, and the joint likelihood when the label is observed.

Adversarial Label Learning (ALL) : This algorithm (Arachie & Huang, 2019) learns a prediction model that has the highest expected accuracy with respect to an adversarial labeling of an unlabeled dataset, where this labeling must satisfy error constraints on the weak supervision sources. This approach shares similarities with our method; however, it fails to provide theoretical guarantees on the learning of the prediction model. For a fair comparison to our method, we use logistic regression as the prediction model, and use the same features as AMCL-LR.

Performance-Guaranteed Majority Vote (PGMV): This method finds a subset of weak supervision sources whose majority vote achieves high accuracy with respect to the worst-case distribution of the output of the weak supervision sources. Again, this worst-case distribution is constrained by using statistics computed on the weak supervision sources (individual error rates and pairwise differences).

Due to the limitations of PGMV and ALL, we can run those algorithms only for binary classification tasks.

5.3. Results

Animals With Attributes (binary classification) : In Figure 1, we report the results on the Animals With Attributes dataset for two binary classification tasks.

In the binary setting, our methods match or outperform the state-of-the-art methods PGMV and ALL over all labeled-sample sizes. We note that even though AMCL-LR and ALL use the same inputs and train the same prediction model, our method achieves overall higher accuracies, in addition to providing theoretical guarantees on the generalization error of the prediction model.

Domain Net (multiclass classification) : In Figure 2, we report the accuracies of the different algorithms on the Domain Net dataset for the clipart and quickdraw domains. As previously discussed, ALL and PGMV cannot be used in this setting, as they are restricted to binary classification.

In the multiclass setting, our methods again match or outperform the baselines over all quantities of labeled data. We note that in the quickdraw domain, the weak supervision sources are overall very inaccurate, and it is dif cult to recover useful information from them. However, unlike the baseline algorithms DS and MV, AMCL-CC can still recover and improve upon the best weak supervision source.

Again, as noted by the Best WSS column, the weak supervision sources are quite inaccurate in this dataset. Therefore, we do not report the results for the AMCL-LR algorithm, as the weak supervision sources do not constrain the feasible set of solutions suf ciently well for our method to accurately learn a (relatively) complex model like a (multinomial) logistic regressor.

Due to space constraints, additional plots and experimental details for both datasets are reported in the appendix.

6. Conclusion

We develop the rst general framework with theoretical guarantees that can use information provided by arbitrarily-correlated weak supervision sources in order to learn a prediction model for a multiclass classi cation task. In many practical settings, our training method provably converges to the model that achieves the smallest risk with respect to an adversarial feasible labeling of an unlabeled dataset, and we provide generalization guarantees on the quality of the learned model based on a measure of the information provided by the weak supervision sources. Surprisingly, our theoretical guarantees for this adversarial learning setting stem from standard methods in convex optimization and uniform convergence theory. Finally, we provide experiments that illustrate the practical applicability of our approach and its advantages over existing methods.

Acknowledgments

The authors would like to thank Michael Littman and James Tompkin for helpful discussions. This material is based on research sponsored by Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory (AFRL) under agreement number FA8750-19-2-1006, and by the National Science Foundation (NSF) under award IIS-1813444. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the of cial policies or endorsements, either expressed or implied, of Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory (AFRL) or the U.S. Government. We gratefully acknowledge support from Google and Cisco. Disclosure: Stephen Bach is an advisor to Snorkel AI, a company that provides software and services for weakly supervised machine learning.

References

Arachie, C. and Huang, B. Adversarial label learning. In AAAI Conference on Arti cial Intelligence (AAAI), 2019.

Arachie, C. and Huang, B. A general framework for adversarial label learning. The Journal of Machine Learning Research, 22:1–33, 2021.

Bach, S. H., He, B., Ratner, A., and Ré, C. Learning the structure of generative models without labeled data. In International Conference on Machine Learning (ICML), 2017.

Bach, S. H., Rodriguez, D., Liu, Y., Luo, C., Shao, H., Xia, C., Sen, S., Ratner, A., Hancock, B., Alborzi, H., Kuchhal, R., Ré, C., and Malkin, R. Snorkel DryBell: A case study in deploying weak supervision at industrial scale. 2019.

Balsubramani, A. and Freund, Y. Optimally combining classi ers using unlabeled data. In Conference on Learning Theory (COLT), pp. 211–225, 2015.

Balsubramani, A. and Freund, Y. Optimal binary classi er aggregation for general losses. In Neural Information Processing Systems (NeurIPS), 2016.

Bertsekas, D. Convex Optimization Algorithms. Athena Scienti c, 2015.

Böhning, D. Multinomial logistic regression algorithm. Annals of the institute of Statistical Mathematics, 44(1): 197–200, 1992.

Breiman, L. Bagging predictors. Machine Learning, 24(2): 123–140, 1996.

Chen, V. S., Varma, P., Krishna, R., Bernstein, M., Ré, C., and Fei-Fei, L. Scene graph prediction with limited labels. In IEEE/CVF International Conference on Computer Vision (ICCV) 2019.

Cousins, C. and Riondato, M. Sharp uniform convergence bounds through empirical centralization. Advances in Neural Information Processing Systems, 33, 2020.

Dalvi, N., Dasgupta, A., Kumar, R., and Rastogi, V. Aggregating crowdsourced binary ratings. WWW '13, pp. 285–294, 2013.

Dawid, A. P. and Skene, A. M. Maximum likelihood estimation of observer error-rates using the EM algorithm. Journal of the Royal Statistical Society, 28(1):20–28, 1979.

Freund, Y. Boosting a weak learning algorithm by majority. *Information and Computation*, 121(2):256–285, 1995.

Gao, B. and Pavel, L. On the properties of the softmax function with application in game theory and reinforcement learning, 2018.

Gao, C. and Zhou, D. Minimax optimal convergence rates for estimating ground truth from crowdsourced labels. *CoRR*, abs/1207.0016, 2013.

Ghosh, A., Kale, S., and McAfee, P. Who moderates the moderators? Crowdsourcing abuse detection in user-generated content. EC '11, pp. 167–176, 2011.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Karger, D. R., Oh, S., and Shah, D. Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research*, 62(1):1–24, 2014.

Mazzetto, A., Sam, D., Park, A., Upfal, E., and Bach, S. H. Semi-supervised aggregation of dependent weak supervision sources with performance guarantees. In *Artificial Intelligence and Statistics (AISTATS)*, 2021.

Mitzenmacher, M. and Upfal, E. *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press, 2017.

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., and Wang, B. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1406–1415, 2019.

Ratner, A., Bach, S. H., Ehrenberg, H., Fries, J., Wu, S., and Ré, C. Snorkel: Rapid training data creation with weak supervision. *Proceedings of the VLDB Endowment*, 11 (3):269–282, 2017.

Ratner, A. J., De Sa, C. M., Wu, S., Selsam, D., and Ré, C. Data programming: Creating large training sets, quickly. In *Neural Information Processing Systems (NeurIPS)*, 2016.

Ratner, A. J., Hancock, B., Dunnmon, J., Sala, F., Pandey, S., and Ré, C. Training complex models with multi-task weak supervision. In *AAAI*, 2019.

Safranchik, E., Luo, S., and Bach, S. H. Weakly supervised sequence tagging from noisy rules. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

Schapire, R. E. The strength of weak learnability. *Machine Learning*, 5(2):197–227, 1990.

Shor, N. Z., Kiwiel, K. C., and Ruszczyński, A. *Minimization Methods for Non-Differentiable Functions*. Springer-Verlag, Berlin, Heidelberg, 1985. ISBN 0387127631.

Varma, P., He, B., Bajaj, P., Khandwala, N., Banerjee, I., Rubin, D., and Ré, C. Inferring generative model structure with static analysis. In *Neural Information Processing Systems (NeurIPS)*, 2017.

Varma, P., Sala, F., He, A., Ratner, A., and Ré, C. Learning dependency structures for weak supervision models. In *International Conference on Machine Learning (ICML)*, 2019.

Wang, W. and Carreira-Perpiñán, M. A. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *arXiv preprint arXiv:1309.1541*, 2013.

Wu, S., Hsiao, L., Cheng, X., Hancock, B., Rekatsinas, T., Levis, P., and Ré, C. Fonduer: Knowledge base construction from richly formatted data. In *International Conference on Management of Data*, 2018.

Xian, Y., Lampert, C. H., Schiele, B., and Akata, Z. Zero-shot learning—A comprehensive evaluation of the good, the bad and the ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2018.

Zhang, C. and Ma, Y. *Ensemble Machine Learning: Methods and Applications*. Springer, 2012.

Zhang, Y., Chen, X., Zhou, D., and Jordan, M. I. Spectral methods meet EM: A provably optimal algorithm for crowdsourcing. *The Journal of Machine Learning Research*, 17(1):3537–3580, 2016.

A. Deferred proofs

Proof of Lemma 1. For the sake of the proof, assume that we have two labeled set of samples of size m and m_L from p_{XY}, call them respectively S and S_L. The set S represents our unlabeled sample, and the set S_L represents the labeled sample. For any $\delta \in (0,1)$, we would like to find a $\epsilon > 0$ such that with probability $1-\delta$, for all $i \in 1,\dots,n$,

$$\left|\frac{1}{m}\sum_{(x,y)\in S}\ell(\psi_i(x),y) - \frac{1}{m_L}\sum_{(x,y)\in S_L}\ell(\psi_i(x),y)\right| \le \epsilon. \tag{6}$$

The sample S represents the unlabeled data $x_1,\dots,x_m$ we have access to. In fact $\frac{1}{m}\sum_{(x,y)\in S}\ell(\psi_i(x),y) = \hat{R}(\psi_i; X, Y^*)$. The inequality (6) implies that for the true labeling of the unlabeled data $x_1,\dots,x_m$, for any $i \in 1,\dots,n$, it holds that:

$$\hat{R}(\psi_i; X, Y^*) \in [\hat{\tau}_i - \epsilon;\ \hat{\tau}_i + \epsilon]$$

where $\hat{\tau}_i = \frac{1}{m_L}\sum_{(x,y)\in S_L}\ell(\psi_i(x),y)$ is the empirical mean computed from the labeled sample S_L.

By using Hoeffding's inequality, we have that for a fixed i, it holds that

$$\begin{aligned}
&\mathbb{P}_{S,S_L}\left(\left|\frac{1}{m}\sum_{(x,y)\in S}\ell(\psi_i(x),y) - \frac{1}{m_L}\sum_{(x,y)\in S_L}\ell(\psi_i(x),y)\right| > \epsilon\right)\\
&\le 2\exp\left(-\frac{2\epsilon^2}{\sum_{j=1}^{m}\left(\frac{B}{m}\right)^2 + \sum_{j=1}^{m_L}\left(\frac{B}{m_L}\right)^2}\right)\\
&= 2\exp\left(-\frac{2 m_L m \epsilon^2}{B^2(m+m_L)}\right) = \frac{\delta}{n}.
\end{aligned}\tag{7}$$

By taking a union bound and solving (7) with respect to ϵ, the statement follows.

Proof of Lemma 4. By invoking Lemma 3, it is easy to see that the function $\hat{R}(h_\theta; X, Y')$ is a convex combination of convex functions with respect to θ, hence it is also convex in θ. Let $v \in \partial\hat{R}(h_{\theta'}; X, Y')$. If a function is convex, then there exists at least one subgradient for each point of its domain, so v is well defined. Then, we have that for any $\theta'' \in \Theta$, it holds that

$$\hat{R}(h_{\theta''}; Y') - \hat{R}(h_{\theta'}; Y') \ge v^T(\theta'' - \theta').$$

As $f(\theta'') \ge \hat{R}(h_{\theta''}; X, Y')$, we have that

$$f(\theta'') - f(\theta') \ge v^T(\theta'' - \theta');$$

which implies that v is a subgradient of f at θ'.

Proof of Theorem 5. We need to show that $f(\theta)$ is convex and L-Lipschitz continuous with respect to θ to apply the standard convergence result for constant step size subgradient optimization (Bertsekas, 2015), which yields

$$f(\tilde{\theta}) - f(\hat{\theta}) \le \frac{\text{diameter}(\Theta)^2 + L^2h^2T}{2hT}. \tag{8}$$

To show that $f(\theta)$ is convex it is straightforward to see that $\hat{R}(h_\theta; X, Y)$ is convex in θ as it is the convex combination of convex functions in θ. For any $\alpha \in [0,1]$, we have that

$$\begin{aligned}
f(\alpha\theta_1 + (1-\alpha)\theta_2) &= \max_{Y\in\mathcal{Y}} \hat{R}(h_{\alpha\theta_1+(1-\alpha)\theta_2}; X, Y)\\
&\le \max_{Y\in\mathcal{Y}}\left[\alpha\hat{R}(h_{\theta_1}; X, Y) + (1-\alpha)\hat{R}(h_{\theta_2}; X, Y)\right]\\
&\le \alpha\max_{Y\in\mathcal{Y}}\hat{R}(h_{\theta_1}; X, Y) + (1-\alpha)\max_{Y\in\mathcal{Y}}\hat{R}(h_{\theta_2}; X, Y)\\
&= \alpha f(\theta_1) + (1-\alpha) f(\theta_2).
\end{aligned}$$

Also, $f(\theta)$ is L-Lipschitz continuous with respect to θ. In fact, it is straightforward to see that $\hat{R}(h_\theta; X, Y)$ is also L-Lipschitz continuous with respect to θ. For any $\theta_1, \theta_2 \in \Theta$, we have that

$$\begin{aligned}
|f(\theta_1) - f(\theta_2)| &\le \max_{Y\in\mathcal{Y}} |\hat{R}(h_{\theta_1}; X, Y) - \hat{R}(h_{\theta_2}; X, Y)|\\
&\le L\|\theta_1 - \theta_2\|_2.
\end{aligned}$$

The subgradient of $f(\theta)$ in θ is computed by using Lemma 4. The last part of the Theorem immediately follows by substituting h and T in (8) as in the Theorem statement.

Proof of Lemma 6. For any $i \in 1,\dots,n$, we have that

$$\frac{\partial}{\partial\theta_i}\ell(h_\theta(x), e) = 2\left(\psi_i(x)^T h_\theta(x) - \psi_i(x)^T e\right).$$

Therefore, we can bound the norm of the gradient of ℓ as

$$\begin{aligned}
\|\nabla_\theta \ell(h_\theta(x), e)\|_2 &= 2\sqrt{\sum_{i=1}^{n}\left(\psi_i(x)^T(h_\theta(x) - e)\right)^2}\\
&\le 2\sqrt{\sum_{i=1}^{n}(1)^2}\\
&\le 2\sqrt{n}.
\end{aligned}$$

The first inequality is an application of Hölder's Inequality, as $\|\psi_i(x)^T\|_1 = 1$ and $\|h_\theta(x) - e\|_\infty \le 1$. This implies that the function $\ell(h_\theta(x); e)$ is $2\sqrt{n}$-Lipschitz continuous with respect to θ.

Proof of Lemma 7. First, we will prove that $\ell(h_\theta(x); e)$ is bounded. Without loss of generality, suppose that $e_i = 1$. We have that

$$\ell(h_\theta(x); e) = -\ln\left(\frac{\exp(w_i^T x)}{\sum_{c=1}^{k}\exp(w_c^T x)}\right).$$

It is easy to see that $\ell(h_\theta(x); e) \geq 0$. By using the Cauchy-Schwarz inequality, we have that

$$\begin{aligned}\ell(h_\theta(x); e) &= -\ln\left(\frac{\exp(w_i^T x)}{\sum_{c=1}^{k}\exp(w_c^T x)}\right)\\ &\leq -\ln \frac{\exp(-B_w B_x)}{k\exp(B_w B_x)}\\ &\leq 2B_w B_x + \ln k\,.\end{aligned}$$

Now, we prove that $\ell(h_\theta(x); e)$ is Lipschitz continuous with respect to θ. For a fixed $(x, e) \in \mathcal{X} \times \mathcal{Y}$, consider the function $\omega(p): \mathbb{R}^k \to \mathcal{Y}$, defined as

$$\omega(p) \doteq -\sum_{c=1}^{k} e_c \ln(p_c)\,,$$

and let

$$h(\theta) \doteq \left(\frac{\exp(w_1^T x)}{\sum_{c=1}^{k}\exp(w_c^T x)}, \ldots, \frac{\exp(w_k^T x)}{\sum_{c=1}^{k}\exp(w_c^T x)}\right)^T,$$

where $\theta = (w_1 \ldots w_k)^T$, and observe that $\ell(h_\theta(x); e) = \omega \circ h(\theta)$.

It is well known that ℓ is $L_\omega L_h$-Lipschitz continuous with respect to θ, where L_ω and L_h are the Lipschitz constants respectively of ω and h. It is also a known result that $L_\omega \leq 1$ (see for example Proposition 4 of (Gao & Pavel, 2018)).

We now want to compute L_h. We will use the fact that $\max_{\theta \in \Theta} \|J_h(\theta)\|_F \leq L_h$, where J_h denotes the Jacobian matrix of h and $\|\cdot\|_F$ denotes the Frobenius norm.

For ease of notation, let $h(\theta) = p = (p_1, \ldots, p_k)^T$. We have that for any $i \in \{1, \ldots, k\}$, it holds that

$$\begin{aligned}\frac{\partial [h(\theta)]_i}{\partial w_j} &= -p_i p_j x \quad \text{for } j \neq i\,; \text{ and}\\ \frac{\partial [h(\theta)]_i}{\partial w_i} &= (p_i - p_i^2)x\,.\end{aligned}$$

Therefore, we can bound the square of the Frobenius norm of the Jacobian matrix of h with

$$\begin{aligned}\|J_h(\theta)\|_F^2 &= \sum_{i,j}\left\|\frac{\partial [h(\theta)]_i}{\partial w_j}\right\|_2^2\\ &\leq \|x\|_2^2\left(\sum_i [p_i(1-p_i)]^2 + \sum_{i\neq j}[p_i p_j]^2\right)\\ &\leq \|x\|_2^2(k + k^2/2) \leq \|kx\|_2^2\,.\end{aligned}$$

We can conclude that h is kB_x-Lipschitz continuous, and the statement follows.

Proof of Theorem 8. From Chapter 14 of Mitzenmacher & Upfal (2017), we know that

$$R(h_{\hat\theta}) \leq \hat{R}(h_{\hat\theta}; X, Y) + 2\hat{\mathcal{R}}_m(L; X, Y) + O\left(\sqrt{\frac{\ln\frac{1}{\delta}}{m}}\right).$$

By definition of $f(\cdot)$, it holds that $\hat{R}(h_{\hat\theta}; X, Y) \leq f(\hat\theta)$. As $\hat\theta$ is the optimal solution of (2), we have that $f(\hat\theta) \leq f(\theta^*)$. Let $Y' \doteq \arg\max_{Y \in \mathcal{Y}} \hat{R}(h_{\theta^*}; X, Y)$. It holds that

$$\begin{aligned}f(\theta^*) &= \hat{R}(h_{\theta^*}; X, Y')\\ &= \hat{R}(h_{\theta^*}; X, Y') + \hat{R}(h_{\theta^*}; X, Y) - \hat{R}(h_{\theta^*}; X, Y)\\ &= \hat{R}(h_{\theta^*}; X, Y) + |\hat{R}(h_{\theta^*}; X, Y') - \hat{R}(h_{\theta^*}; X, Y)|\,.\end{aligned}$$

By using the fact that ℓ is bounded, and the definition of diameter $D_{\mathcal{Y}}$, we have that

$$\begin{aligned}&|\hat{R}(h_{\theta^*}; X, Y') - \hat{R}(h_{\theta^*}; X, Y)|\\ &= \frac{1}{m}\sum_{j=1}^{m}\sum_{c=1}^{k}\ell(h_{\theta^*}(x_j); e_c)(y'_{jc} - y_{jc})\\ &\leq B\frac{1}{m}\sum_{j=1}^{m}\sum_{c=1}^{k}|y'_{jc} - y_{jc}| \leq BD_{\mathcal{Y}}\,.\end{aligned}$$

To wrap it up, it results that

$$\begin{aligned}R(h_{\hat\theta}) &\leq \hat{R}(h_{\theta^*}; X, Y) + BD_{\mathcal{Y}} + 2\hat{\mathcal{R}}_m(L; X, Y)\\ &\quad + O\left(\sqrt{\frac{\ln\frac{1}{\delta}}{m}}\right)\\ &\leq R(h_{\theta^*}) + BD_{\mathcal{Y}} + 4\hat{\mathcal{R}}_m(L; X, Y)\\ &\quad + O\left(\sqrt{\frac{\ln\frac{1}{\delta}}{m}}\right)\\ &\leq R(h_{\theta^*}) + BD_{\mathcal{Y}}\\ &\quad + \sup_{Y \in \mathcal{Y}} 4\hat{\mathcal{R}}_m(L; X, Y) + O\left(\sqrt{\frac{\ln\frac{1}{\delta}}{m}}\right).\end{aligned}$$

Proof of Lemma 9. The proof is along the same lines of the proof of Lemma 1, but we take a union bound with respect to all the nK intervals $\Delta_{i,c,\hat{c}}$ for $i \in \{1, \ldots, n\}$, $c \in \{1, \ldots, k\}$, and $\hat{c} = 1, \ldots, k_i$. Moreover, as for any $j \in \{1, \ldots, m\}$, we have that $y_{j,c}[\sigma_i(x_j)]_{\hat{c}} \leq 1$, we take $B = 1$ during the proof (as defined in Lemma 1).

B. Additional Experimental Details

We provide further information specifying the experimental setup used to generate our figures.

B.1. Weak Supervision Sources

We first build the weak supervision sources on our two datasets as follows.

Animals with Attributes. Each class is annotated with a binary vector of attributes. For each attribute, we train a binary classifier by finetuning a ResNet-18 using labeled data from the seen classes. When we consider a classification task between two unseen classes, we use as weak supervision sources the classifiers for the attributes which are different between the animals of these two unseen classes. We report the results of the 4 binary classification tasks which have the lowest majority vote accuracy. We chose these particular tasks to demonstrate the abilities of our methods on the tasks that have the least accurate weak supervision sources.

DomainNet. We sample 5 of the 25 classes of DomainNet with the largest number of datapoints. For each domain, we use 60% of the available data for those classes to fine tune a pretrained ResNet-18 network. We perform this procedure on two disjoint samples of test classes to illustrate our results on two distinct multiclass classification tasks.

In our experiments, we use the pretrained ResNet-18 from PyTorch. We finetune this ResNet-18 network following the approach described in (He et al., 2016), using cross-entropy loss.

B.2. Algorithm Hyperparameters

The subgradient method (Algorithm 1) used to train AMCL-CC and AMCL-LR uses the following hyperparameters:

AMCL-CC: We set $= 0.1$, and build the constraints as in Lemma 1. We use $= 0.1$, and define the step size h and the number of iterations T as in Theorem 5, using $L_g = 2\sqrt{n}$ and diameter of equal to $\sqrt{2}$.

AMCL-LR: In this case, the loss function is bounded as in Lemma 7. Since this value could be potentially very large, which in turn it would result in large intervals and number of iterations, we use the value $B = 0.1$ in the experiments. We set to 0.1 and build the constraints as in Lemma 1. We do not bound the set of weights: in the experiments, the norm of the weights of the multinomial logistic regression model has never diverged. We run the subgradient algorithm for $T = 1000$ iterations with step size $h = 0.02$.

C. Additional Figures

C.1. Animals with Attributes

We provide the remaining figures for our experiments on the Animals with Attributes dataset. The last two binary classification tasks are bat v. rat and horse v. giraffe.

From Figure 3, we note that our methods show similar results as the figures displayed in the main body of the paper. AMCL-LR matches or outperforms all other methods on both tasks, over all ranges labeled data. AMCL-CC is within a few accuracy points of the other baselines and AMCL-LR on these tasks.

C.2. DomainNet

We provide the remaining figures for our experiments on the DomainNet dataset. We provide histograms when using the other 4 domains as the target task and also provide histograms for results on another of the samples of 5 classes. The first sample of classes as mentioned in the main body of the paper is sea turtle, vase, whale, bird, violin. The second sample is tornado, trombone, submarine, feather, zebra.

From Figures 4–8, we note that in most domains our methods perform better than or match all other approaches, namely in both samples of Clipart, Quickdraw, Painting, and the second sample of Sketch. Our methods achieve slightly lower accuracy than the best performing baseline on the Real domain and on the second sample of the Infograph domain, although they are not beaten by a single baseline in all of these tasks. We believe that the combination of our theoretical guarantees and that our methods achieve similar or sometimes better empirical performance captures the benefits of AMCL.

Figure 3.Experimental results on the Animals with Attributes dataset for the binary classi cation tasks of bat vs. rat and horse vs. giraffe as we vary the amount of labeled data. Each method uses 347 unlabeled data for bat vs. rat and 1424 unlabeled data for horse vs. giraffe.

Figure 4.Experimental results on the second sample of Domain Net for the clipart and quickdraw domains as we vary the amount of labeled data. Each method uses 500 unlabeled data.

Figure 5.Experimental results on both samples of Domain Net for the Infograph domain as we vary the amount of labeled data. Each method uses 500 unlabeled data.

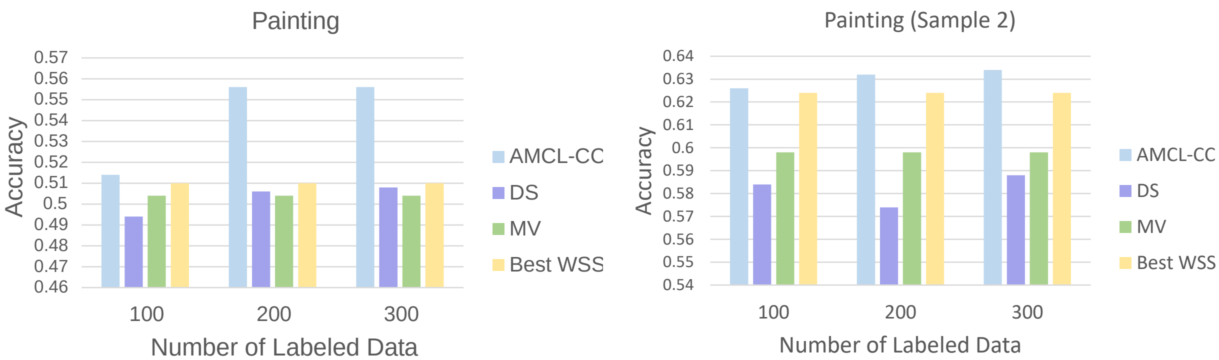

Figure 6. Experimental results on both samples of Domain Net for the Painting domain as we vary the amount of labeled data. Each method uses 500 unlabeled data.

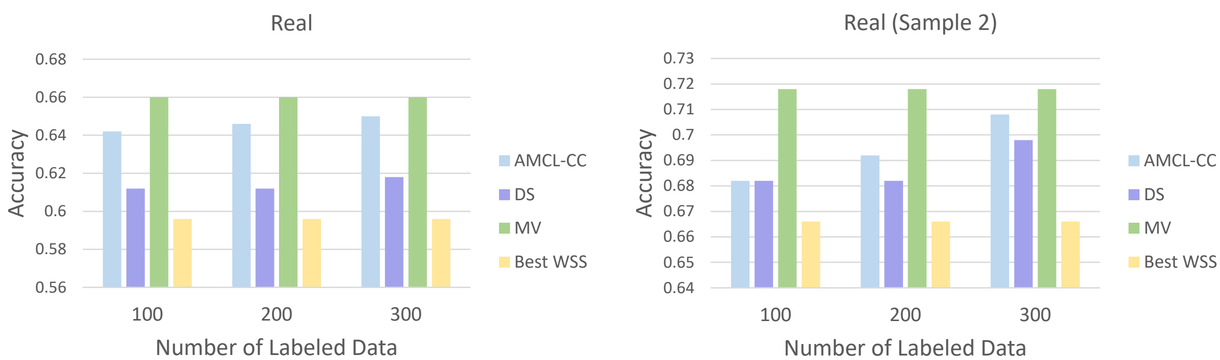

Figure 7. Experimental results on both samples of Domain Net for the Real domain as we vary the amount of labeled data. Each method uses 500 unlabeled data.

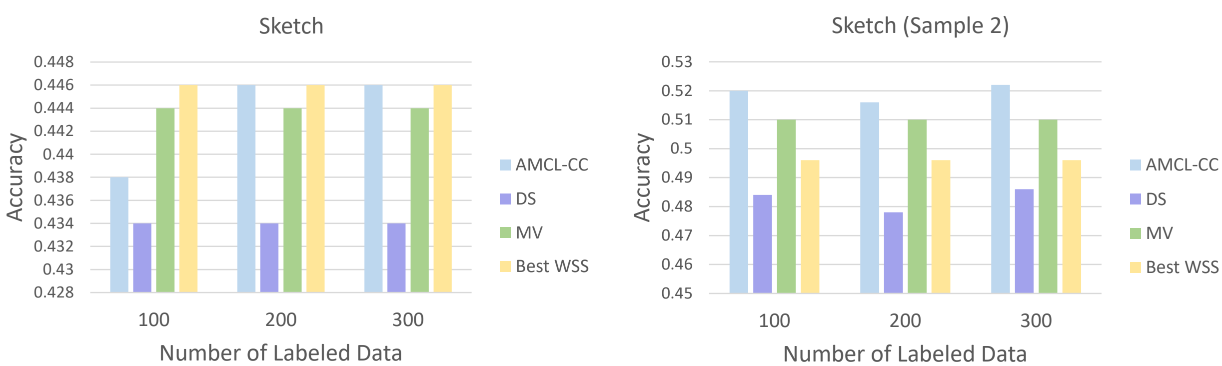

Figure 8. Experimental results on both samples of Domain Net for the Sketch domain as we vary the amount of labeled data. Each method uses 500 unlabeled data.

D. Experiments on Synthetic Data

We run synthetic experiments to show that our method is robust with respect to the addition of correlated weak supervision sources. Similar experiments have been done for ALL by Arachie & Huang (2019).

We consider a multiclass classification task over 5 classes, and 25 weak supervision sources $\,_1, \ldots, \,_{25}$. In this classification task, each item of the domain X has a unique true label. Given an item $x \in X$, for $i \in 1, \ldots, 10$, the weak supervision source $\,_i$ returns the correct label with probability $1/2$, and a random label with probability $1/2$. The output of the weak supervision source $\,_i$ is independent to the output of the weak supervision sources $\,_j$ for $j \in \{1, \ldots, 10\} \setminus \{i\}$. Therefore, the weak supervision source $\,_i$ is correct with probability $\frac{1}{2}(1 + \frac{1}{k})$. For $i = 11, \ldots, 25$, the weak supervision sources $\,_i$ outputs the same result than the weak supervision source $\,_1$. Note that the weak supervision sources $\,_{11}, \ldots, \,_{25}$ do not provide any additional information with respect to the target classification task, as they add redundant constraints to the set of feasible labelings Y. The majority vote of the weak supervision sources $\,_1, \ldots, \,_{25}$ is highly affected by these dependencies, and it is very likely to provide the same answer as $\,_1$, which is only $\frac{1}{2}(1 + \frac{1}{k})$ accurate on average. On the other hand, the majority vote of the weak supervision sources $\,_1, \ldots, \,_{10}$ would improve upon the individual accuracy of the weak supervision sources, as their output is independent.

We use 500 unlabeled examples, run experiments varying the amount of labeled data, and show that our method AMCL-CC is robust against those dependencies. For the sake of these experiments, as we want to use very small amount of labeled data, we set $\; = 0$ when building the constraints for Y as in Lemma 1. The experimental results are reported in Table D. The table shows that AMCL-CC is robust with respect to dependencies among weak supervision sources, whereas majority vote is greatly affected by them. In fact, in this case the majority vote does not improve upon the individual accuracy of the weak supervision sources, which is on average $\frac{1}{2}(1 + \frac{1}{k}) = \frac{3}{5}$.

Table 1. We report the experimental results on the synthetic dataset. We report the accuracy obtained by our method AMCL-CC and the majority vote, when varying the amount of labeled examples (we report the average accuracy over 3 distinct runs).

Labeled Examples	AMCL-CC	Majority Vote
100	0.902	0.595
50	0.828	0.602
25	0.819	0.598